The Unbalancing Effect of Hubs on K–medoids Clustering in High-Dimensional Spaces

Dominik Schnitzer, Arthur Flexer
Austrian Research Institute for Artificial Intelligence
Freyung 6/6, Vienna, Austria
Email: arthur.flexer@ofai.at

***Abstract*—Unbalanced cluster solutions are affected by very different cluster sizes, with some clusters being very large while others contain almost no data. We demonstrate that this phenomenon is connected to 'hubness', a recently discovered general problem of machine learning in high dimensional data spaces. Hub objects have a small distance to an exceptionally large number of data points, and anti-hubs are far from all other data points. In an empirical study of K–medoids clustering we show that hubness gives rise to very unbalanced cluster sizes resulting in impaired internal and external evaluation indices. We compare three methods which reduce hubness in the distance spaces and show that with the balancing of the clusters evaluation indices improve. This is done using artificial and real data sets from diverse domains.**

I. Introduction

In a number of recent publications hubness has been introduced and discussed as a new aspect of the curse of dimensionality [1], [2], [3]. Hub objects are data points which have a small distance to many other data points in high dimensional data spaces which is related to the phenomenon of concentration of distances. This behavior has a negative impact on many machine learning tasks including classification [1], nearest neighbor based recommendation [4] and retrieval [5], outlier detection [1], [6] and also first results on their influence on clustering exist [7]. Since clustering algorithms aim at finding groups of similar objects it is evident that hub objects, being similar to very many objects, have a decisive impact on all forms of clustering. In this paper we investigate the impact of hubs on K–medoids, a partitional clustering algorithm. Our general hypothesis is that hub points are very likely to be selected as cluster centers since per definition they have a small distance to a large number of data points. As a consequence, hub points acting as cluster centers aggregate too large portions of the data in individual clusters. This results in unsatisfactory and unbalanced clusterings as it has already been observed to occur in high dimensions [8].

This paper is the first to connect this unbalancing effect to the phenomenon of hubness in high dimensional data spaces. We demonstrate the effect on artificial as well as on real data and evaluate three unsupervised methods to remove hubs and balance the clusterings: local scaling (LS, [9]), mutual proximity (MP, [2]) and shared nearest neighbors (SNN, [10]).

II. Related Work

The 'hubness' phenomenon is a general problem of machine learning in high-dimensional data spaces and as such yet another aspect of the curse of dimensionality [11]. Hubs are data points which keep appearing unwontedly often as nearest neighbors of a large number of other data points. The hub problem has been linked [1] to the property of concentration [12] which occurs as a natural consequence of high dimensionality. Concentration is the surprising characteristic of all points in a high dimensional space to be at almost the same distance to all other points in that space. It is usually measured as a ratio between spread and magnitude, e.g. the ratio between the standard deviation of all distances to an arbitrary reference point and the mean of these distances. If the standard deviation stays more or less constant with growing dimensionality, while the mean keeps growing, the ratio converges to zero with dimensionality going to infinity. In such a case it is said that the distances concentrate. This has been studied for Euclidean spaces and other ℓ^p norms [13], [12]. It has been argued [1] that in the finite case, some points are expected to be closer to the center 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. It is also important to note that the degree of concentration and hubness is linked to the intrinsic rather than extrinsic dimension of the data space. Whereas the extrinsic dimension is the actual number of dimensions of a data space the intrinsic dimension is the, often much smaller, number of dimensions necessary to represent a feature space without loss of information.

The same authors [1] also argue that hubness impacts both inter– and intra–cluster distances. Hubs are expected to be close to many other data points, including points from other clusters and hence reduce inter-cluster distances. On the other hand anti-hubs are far away from all other data points, including points from clusters to which they belong, thereby increasing intra-cluster distances. This effect has been demonstrated with lower silhouette indices for hubs and anti-hubs compared to random points on eight data sets using spectral clustering [1]. It has also been tried to exploit the hubness phenomenon for clustering high-dimensional data by using hub points as cluster centers in hubness-aware clustering algorithms [7]. Results look promising for data with high levels of noise while results on non-noisy data are rather mixed. Established approaches to achieve high-dimensional clustering are subspace, projected and correlation clustering all searching for solutions in some lower dimensional data space (see [14] for an overview). There are pointers in the literature that K–means clustering "generates some clusters that are empty or

extremely small, specially when the data is in high dimensional (>100) space" [8]. Although a number of methods that counter this unbalancing effect have been published [15], [16], [17], [18], [19], [20], no connection to the concentration of distances or hubness has yet been established.

Two methods (local scaling (LS) and mutual proximity (MP)) which are able to attenuate the negative effects of hubness by repairing asymmetric nearest neighbor relations have been proposed [2]. The asymmetric relations are a direct consequence of the presence of hubs since a hub y is the nearest neighbor of x, but the nearest neighbor of the hub y is another point a ($a \neq x$). This is because hubs are by definition nearest neighbors to very many data points but only a fixed number of data points can be the k-nearest neighbors to a hub. Thus a small distance between two objects should be returned only if their nearest neighbors concur. The positive impact of LS and MP was measured in a decrease of hubness and an accuracy increase in k-nearest neighbor classification experiments. The somewhat related classic shared nearest neighbors method (SNN) has recently been evaluated concerning its ability to 'defeat' the curse of dimensionality [21]. SNN and its impact on hubness have been studied for nearest neighbor classification [22] and compared to LS and MP [23]. It was shown that SNN does reduce hubness but less than LS and MP and that it is not able to improve classification accuracy for all of the data sets used in the study. A different approach to reduce hubness is to center the data either locally or globally [24]. This gives nearest neighbor classification results comparable to LS and MP when applied to text data. It has also been tried [25] to use l^p norms different than the ubiquitous Euclidean l^2 norm for computation of nearest neighbor lists to avoid the negative effects of hubness. This approach also yields lower hubness values and increased nearest neighbor classification for a range of real world data sets. Comparison of using different l^p norms to LS, MP and SNN shows [26] that it is highly problem dependent which of the approaches works best for a given data set.

III. Methods

Before performing this study we briefly introduce all methods and evaluation measures used in this work. We will use C to denote the set of classes $C = \{c_1, c_2, \ldots, c_J\}$, and Ω to denote the set of clusters $\Omega = \{\omega_1, \omega_2, \ldots, \omega_K\}$. The number of items in the data set will be denoted with N, the extrinsic dimensionality of our data spaces by d.

K–medoids: The K–medoids algorithm [27] is a classic partitional clustering algorithm which clusters a data set of N objects into K clusters. In contrast to K–means clustering, the algorithm works by selecting points (medoids) from the data as cluster centers which have a minimal distance to the points in the cluster. A medoid is defined as an object of a cluster ω_k with the minimal average distance to all the objects in the cluster. It is the most centrally located point in the cluster. K–medoids works with an arbitrary matrix of distances between data points and was shown to be robust to noise and outliers. Please note that we initialize K–medoids randomly and use the same initialization when we compare two cluster configurations build on the same data but with different distance measures. All results are usually averaged over multiple runs (30–100) of K–medoids.

Cluster Quality Benchmarks: We use the following indices to measure the strength of hubness as well as the quality of clustering solutions.

Hubness (S^n): To compute hubness we first define $O^n(x)$ as the n-occurrence of point x, that is, the number of times x occurs in the n-nearest neighbor lists of all other objects in the collection. Hubness is then defined as the skewness of the distribution of n-occurrences, O^n:

$$S^n = \frac{\mathrm{E}\big[(O^n - \mu_{O^n})^3\big]}{\sigma^3_{O^n}}. \tag{1}$$

A data set having high hubness produces few hub objects with very high n-occurrence and many anti-hubs with n-occurrence of zero. This makes the distribution of n-occurrences skewed with positive skewness indicating high hubness.

Goodman–Kruskal index: The Goodman–Kruskal index [28] is an internal clustering quality measure that relates the number of *concordant* (Q_c) and *discordant* (Q_d) quadruples found in the data set by comparing their distances (D). A quadruple is concordant if the items x, y are from the same cluster, items u, v are from different clusters and $D_{x,y} < D_{u,v}$, they are discordant if $D_{x,y} > D_{u,v}$. I_{GK} is then defined for a cluster configuration Ω:

$$I_{GK}(\Omega) = \frac{Q_c - Q_d}{Q_c + Q_d}. \tag{2}$$

I_{GK} is bounded to the interval $[-1, 1]$, and the higher I_{GK}, the more concordant and fewer discordant quadruples are present in the data set. Thus a large index value indicates a good clustering (in terms of *pairwise stability*—see [29]).

Purity: To compute purity [30], each cluster ω is assigned to the class which is most frequent in the cluster, and then the accuracy of this assignment is measured by counting the number of correctly assigned data and dividing by N:

$$purity(\Omega, C) = \frac{1}{N} \sum_k \max_j |\omega_k \cap c_j|. \tag{3}$$

Mutual Information: Mutual information [30] is an information theoretic clustering benchmark. It measures the degree of dependency between a cluster configuration Ω and its classes C. We use the mutual information criterion because it successfully captures how related the labeling and clustering are without a bias towards smaller clusters [31]. It is defined for a cluster configuration (Ω) and a class labeling (C) as:

$$I(\Omega, C) = \sum_k \sum_j \frac{|\omega_k \cap c_j|}{N} \log \frac{N|\omega_k \cap c_j|}{|\omega_k||c_j|}. \tag{4}$$

Removing Hubs: We briefly introduce the three methods we use to remove hubs by re-scaling the whole distance matrix and transforming each distance[1].

Local Scaling (LS): Local scaling [9] transforms arbitrary distances to so-called *affinities* (i.e. similarities) according to:

$$LS(D_{x,y}) = \exp\left(-\frac{{D_{x,y}}^2}{\sigma_x\, \sigma_y}\right), \tag{5}$$

[1]Matlab scripts for hubness analysis including Local Scaling and Mutual Proximity are available for download on our web page:
http://ofai.at/research/impml/projects/hubology.html

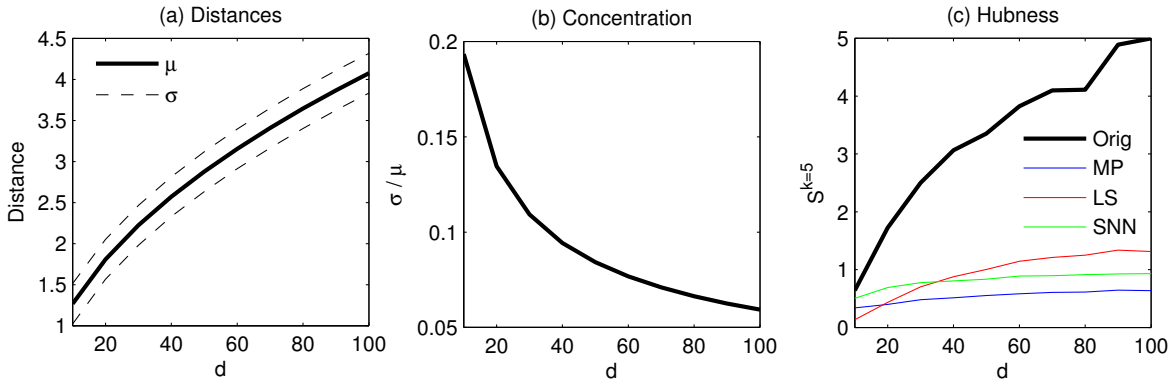

Fig. 1. With increasing dimensionality, the measured ℓ^2 distances concentrate and the hubness effect becomes more pronounced. All three methods, LS (red), MP (blue) and LS (green), seem to be a viable method to reduce hubness.

where σ_x denotes the distance between object x and its k'th nearest neighbor. $LS(D_{x,y})$ tends to make neighborhood relations more symmetric by including local distance statistics of both data points x and y in the scaling. Contrary to [9], who propose to use LS with $k = 7$, we use LS with a $k = 10$, as it returned the best and most stable results.

Mutual Proximity (MP): MP [2] reinterprets 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. This is done by transforming the distance of two objects into a mutual proximity in terms of their distribution of distances. To compute MP, we assume that the distances $D_{x,i=1..N}$ 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 $P(X)$. In this work we assume that the distances $D_{x,i=1..N}$ follow a Gaussian distribution. Previous results [2] have shown that MP is very robust with respect to the choice of $P(X)$. MP is defined as the probability that y is the nearest neighbor of x given $P(X)$ and x is the nearest neighbor of y given $P(Y)$:

$$MP(D_{x,y}) = P(X > D_{x,y} \cap Y > D_{y,x}). \tag{6}$$

Shared Nearest Neighbors (SNN): SNN [10] also uses the neighborhood information to help enforce pairwise stability. In contrast to LS and MP which scale the distance space, SNN is computed as a set intersection of the k-nearest neighbor lists NN of two objects x, y:

$$SNN(x, y) = |NN(x) \cap NN(y)|. \tag{7}$$

This way SNN strictly strengthens symmetric nearest neighbor relations which in turn should also manifest itself in a reduction of hubness. We use SNN with $k = 50$ (see Sect. V for more information).

Computing $1-LS$, $1-MP$ and $1-SNN$ turns the above similarities into distance measures. As to computational costs, both LS and SNN require knowledge of a certain number of nearest neighbors and therefore have to use the full $N \times N$ distance matrix. For MP, the Gaussian distribution parameters can be effectively estimated based on a small fraction of randomly selected data points S. It is therefore only necessary to compute a distance matrix of size $N \times S$, with S being as small as thirty [2].

IV. Experiments with Artificial Data

In our experiments we use uniformly distributed data randomly sampled from a d-dimensional unit cube. We start by generating two dimensional data ($d = 2$) and gradually increase the data dimensionality to $d = 100$, sampling $N = 2\,000$ data points and averaging our measurements of distance concentration and hubness over 100 repetitions. We use the Euclidean distances (ℓ^2 norm). In Fig. 1.a and 1.b we can see that with increasing dimensionality, the distances clearly concentrate. At the same time the measured hubness (Fig. 1.c) increases steadily up to a value of 5, which already indicates a strong skewness of the O^n–occurrences. The figure also shows the impact of using LS (red line), MP (blue) and SNN (green) on the distances (Fig 1.c). The measured hubness declines to very low values.

The Effect of Increasing Dimensionality on Clustering: Clustering uniformly distributed data in K clusters should yield a set of K equally sized clusters Ω. Looking at Fig. 2.a.1, which depicts the result of a K–medoids clustering of $N = 2\,000$ uniformly distributed data points in the two dimensional space, this is exactly what happens. Figure 2.a.2 shows that $K = 15$ clusters of roughly the same size are found (their size is about $N/K = 133.\dot{3}$). However in the high dimensional space ($d = 100$) the same experiment yields very unbalanced cluster sizes (a schematic plot[2] based on the cluster sizes of the obtained cluster configuration is shown in Fig. 2.b.1). From the cluster size histogram (Fig. 2.b.2) we see that the largest cluster suddenly attracts over 400 points, while the smallest cluster only contains three data points. When applying LS, MP or SNN to reduce hubness two things can be seen to happen in the high dimensional case (see Fig. 2.c): (i) the cluster sizes are much more balanced compared to the original clustering and (ii) the large cluster attracting more than 400 points and the smallest cluster with only three points vanished.

[2]Since it is not possible to visualize a one hundred dimensional data space we provide this schematic plot as an illustration.

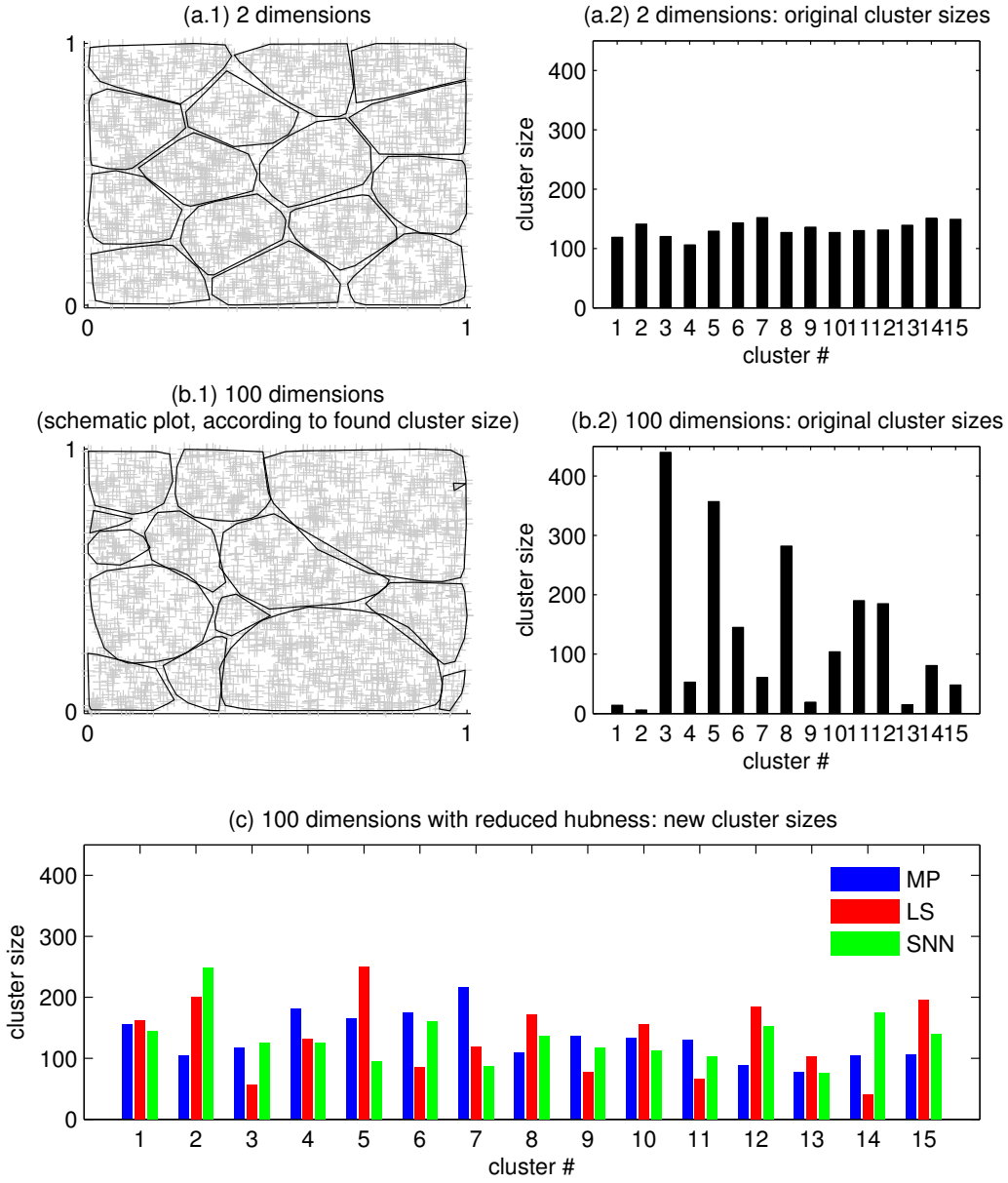

Fig. 2. Visualization of clustering artificial uniformly distributed data: (a) in the two-dimensional, and (b) 100-dimensional case. Effects of reducing hubness shown in (c).

The Relation of Hubs and Cluster Medoids: In our next experiment we investigate to what degree hubs are responsible for the skewed cluster sizes by examining if hub points are more often selected as medoids than other points. We again increase the dimensionality of our artificial data from $d = 2$ to $d = 100$ and check if the medoid of (i) the largest cluster (ω_{max}) and (ii) any cluster (ω_K) is the biggest hub point in the respective data space. Note that assumption (i) is a very strict one, as we check if a single point with highest $O^{n=5}$ value in a dataset of $N = 2000$ points is selected as the medoid of the largest cluster of $K = 15$ clusters. Figure 3 shows the result of the clustering experiment (averaged over 100 runs). The blue line displays the percentage of times the largest hub was in fact the medoid of the largest cluster at a given feature dimension, the red line tracks the percentage the largest hub is chosen as a medoid of any of the clusters. In both cases high dimensionality (and hubness) yields very high numbers for both measurements. At $d = 100$ the largest hub is found as medoid of the largest cluster in over 80% of the cases, while the hub is almost always ($> 95\%$ of the time) chosen as medoid of one of the 15 clusters.

Summary of Experiments with Artificial Data: To summarize our experiments with artificial data, we made the following observations: (i) in high dimensions, and with high

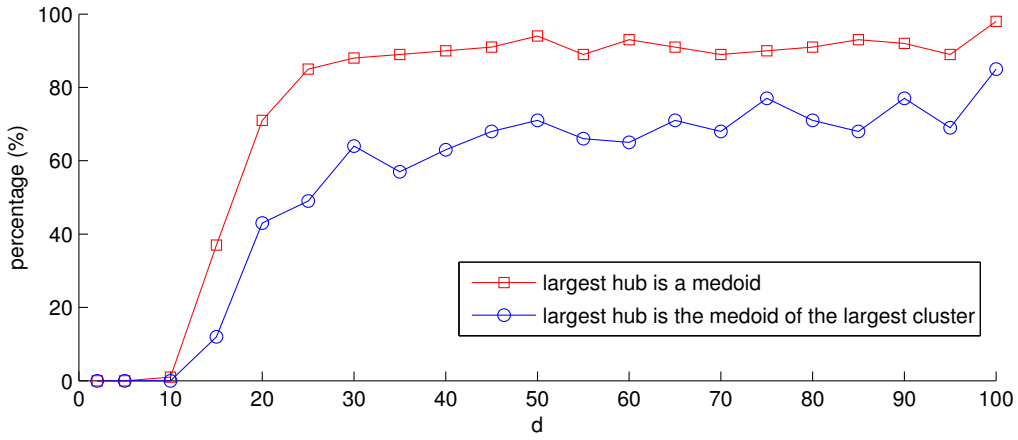

Fig. 3. Percentage of instances where the largest hub object is identical with the medoid of the largest cluster (blue) or any (red) of the clusters (y-axis) vs. dimensionality d (x-axis).

hubness, highly unbalanced clusters (in terms of their size) emerge; (ii) hubs tend to be selected as the medoid of these clusters; (iii) reducing hubness has a balancing effect on the resulting clustering with artificially generated data.

V. Experiments with Real Data

For the experiments with real data we examine six standard machine learning datasets from different domains showing high hubness: (i) *dexter*, *dorothea*, *mini-newsgroups* (UCI, [32]), (ii) *splice*, *dna* (LibSVM, [33]), (iii) *c1ka-twitter* (CP, [34]). Whereas *dexter*, *mini-newsgroups* and *c1ka-twitter* are text-based data sets, *dorothea*, *splice* and *dna* contain data from biological domains.

The selected data sets are characterized in more detail in Tab. I. Each set is described by its number of classes (Cls), its size (N), its extrinsic (d) and intrinsic (d_{mle}) data dimension and the distance measure used (column *Distance*). To measure the intrinsic data dimension we use the maximum likelihood estimator proposed by [35]. We sorted the table according to the hubness ($S^{n=5}$) of each dataset. We also show the reduced hubness when using LS, MP and SNN. Since the performance of SNN varies with the selected k, depending on the size of the collection and individual class sizes, we did a range search for $k = 5, 10, 20, \ldots, 50$ in all six databases. We found that performance plateaued at $k = 50$ for all tested databases.

Medoids and Hubs: In a first experiment with the real data sets, we examine our hypothesis that hubs tend to be selected as medoids. The setup is similar to the one with artificial data which we conducted in Sect. IV, but this time we vary the number of clusters since the different data sets already show varying degrees of intrinsic dimensionality. We track the largest hub in each data set to check if it gets selected as a medoid. The result of this experiment is shown in Fig. 4 where the impact of hubs on the medoid set is clearly visible: the higher the hubness of the data set (cf. $S^{n=5}$ in Tab. I), the more likely the largest hub is selected as a medoid. For the three data sets with the largest measured hubness (*dna*, *c1ka-twitter*, *dorothea*), the largest hub is *always* among the medoids. For data sets *mini-newsgroups* and *splice* the largest hub is one of the medoids in about 80% to 100%. Only with *dexter*, having the lowest hubness value, the percentage decreases below 50%.

Clustering and Hubs in Real Data Sets: The next experiment investigates the real data sets in more depth by concurrently evaluating internal as well as external clustering measures for the original data, LS, MP and SNN. We increase the number of clusters from $K = 2 \ldots 50$. In each iteration we compute internal as well as external benchmarks and compare them to the configurations created when reducing hubs in the data (see Sect. III for details). Figure 5 shows the results, please note the different scales in the plots for the different data sets and measures in these figures.

The first column in Fig. 5 shows the size of the largest cluster (relative to the size of the data set: $|\omega_{max}|/N$) for all six data sets. Due to the unbalancing effect of hubs, the size of the largest cluster stays constantly high at some level in the original distance space (black line). In the extreme case of data set *c1ka-twitter*, a single cluster attracts over 80% of all data points, even when clustering with $K = 50$ clusters where we would expect to see cluster sizes around 2%. For all data sets, the size of the largest cluster is far above the expected value of K/N with results being worst for data sets with highest hubness (*dorothea*, *c1ka-twitter*, *dna*). Results after reducing hubness before clustering (using LS (red line), MP (blue line) or SNN (green line)) are much closer to these expected values and always well below the highly unbalanced results for the original data.

The second column in Fig. 5 shows the Goodman–Kruskal index as an internal measure of cluster quality. Results for the original distance spaces (black line) are very low for all six data sets with all of the three hubness reducing methods decisively improving the results and SNN performing best for 4 out of 6 data sets. Results for the original distance space are again worst for data sets with highest hubness (*dorothea*, *c1ka-twitter*, *dna*).

The third and fourth columns in Fig. 5 show the external benchmark measures purity and mutual information based on class label information. We observe that results for the three hubness reducing methods across all data sets (colored lines) are above results for original distance spaces (black lines) in

TABLE I. SELECTED DATA SETS ORDERED BY ASCENDING HUBNESS ($S^{n=5}$) OF THE DISTANCE SPACE.

Name	Cls	N	d	d_{mle}	Dist.	$S^{n=5}$	$S^{n=5}_{MP}$	$S^{n=5}_{LS}$	$S^{n=5}_{SNN}$
dexter	2	300	20 000	161	cos	**4.22**	0.13	1.41	1.65
splice	2	1 000	60	27	ℓ^2	**4.55**	0.48	1.18	1.34
mini-newsgroups	20	2 000	8 811	188	cos	**5.14**	0.60	0.93	1.13
dorothea	2	800	100 000	201	ℓ^2	**12.91**	1.66	1.23	0.82
c1ka-twitter	17	969	49 820	46	cos	**14.63**	1.79	3.42	1.39
dna	3	2 000	180	33	cos	**16.52**	0.59	1.31	1.28

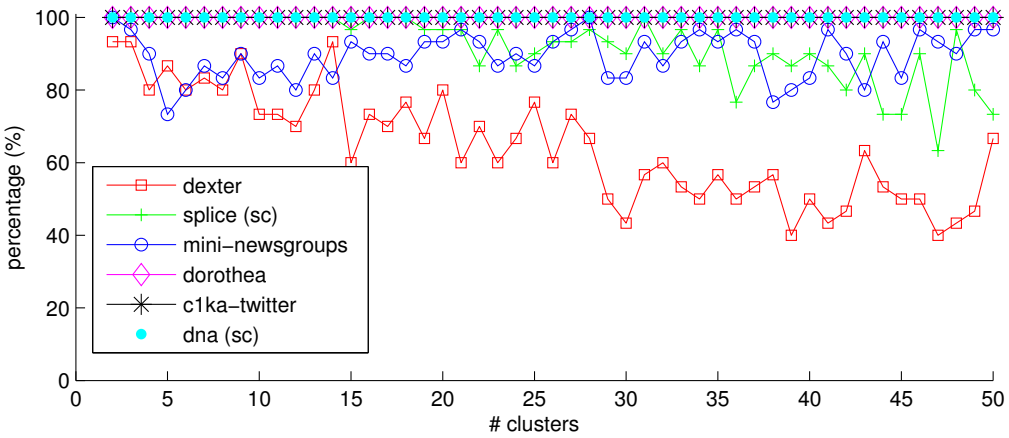

Fig. 4. The percentage of cluster configurations where the largest hub is a cluster medoid versus the number of clusters.

32 out of 36 cases. Exceptions are SNN for *dorothea* and purity, LS and SNN for *dexter* and mutual information, SNN for *dorothea* and mutual information. The improved purity and mutual information measures show that LS, MP and SNN not only decrease hubness and create better cluster solutions but also that these clusters are more meaningful in terms of the corresponding class label information. We also note that at very low number of clusters (K) the negative effects of hubs are not that pronounced and sometimes applying some of the hubness reducing methods yields even worse cluster configurations. If the data is labeled into J classes, one strategy of clustering is to set the number of clusters K equal to J. But very often this class information does not directly correspond to the number and size of clusters to be discovered, since data from one class may form multiple clusters. Most of the six data sets used in this study are labeled into only two or three classes (see Tab. I). This seems to be the reason why external evaluation criteria based on class labels do not always show improved results at low numbers of clusters even for high dimensional data.

Summary of Experiments with Real Data: The results of our experiments examining the impact of hubs on clustering real data sets confirm the effects identified in our experiments with artificial data, but also further indicate that hub reducing methods like LS, MP or SNN are increasing the quality of the clustering measured with internal as well as external measures. All three methods seem to be a viable solution to 'defeat' the curse of dimensionality and unbalanced cluster sizes in K–medoids clustering.

VI. CONCLUSION

This work conducted an analysis of the impact of hubness on medoid-based partitional clustering in high dimensional data sets. In our study we used artificial as well as real data sets to show that: (i) hubs are very likely to be chosen as cluster medoids, (ii) these hub-medoids are responsible for very large unbalanced clusters, (iii) this results in impaired internal and external cluster quality measures. In addition to demonstrating these effects we applied three different methods to reduce hubness before clustering as a preprocessing step. All three hubness reduction methods, Local Scaling (LS), Mutual Proximity (MP) and Shared Nearest Neighbors (SNN), are able to decisively reduce these problems and lead to better clustering results.

Our study used the partitional clustering algorithm of K-medoids but other cluster algorithms aiming at minimizing intra-cluster distances to a centroid are very likely to be effected in an equal manner. Since negative effects of hubs are most visible when clustering with a high number of clusters, reducing hubness could probably be most important when creating fine-grained hierarchical clusters of high dimensional data (e.g. phylogenetic trees of DNA sequences). In future work we will compare our approach to methods that directly constrain cluster sizes to make them balanced as well as to established approaches to cluster high-dimensional data like subspace, projected or correlation clustering. It will also be interesting to investigate to what extent these alternative methods influence and possibly reduce hubness.

The phenomenon of hubness as another aspect of the curse of dimensionality is still a rather new topic within the field of machine learning. In this paper we were able to demonstrate the so far unknown unbalancing effect of hubness on clustering high dimensional data. At the same time we presented the possible remedy of pre-processing the data to reduce hubness before clustering.

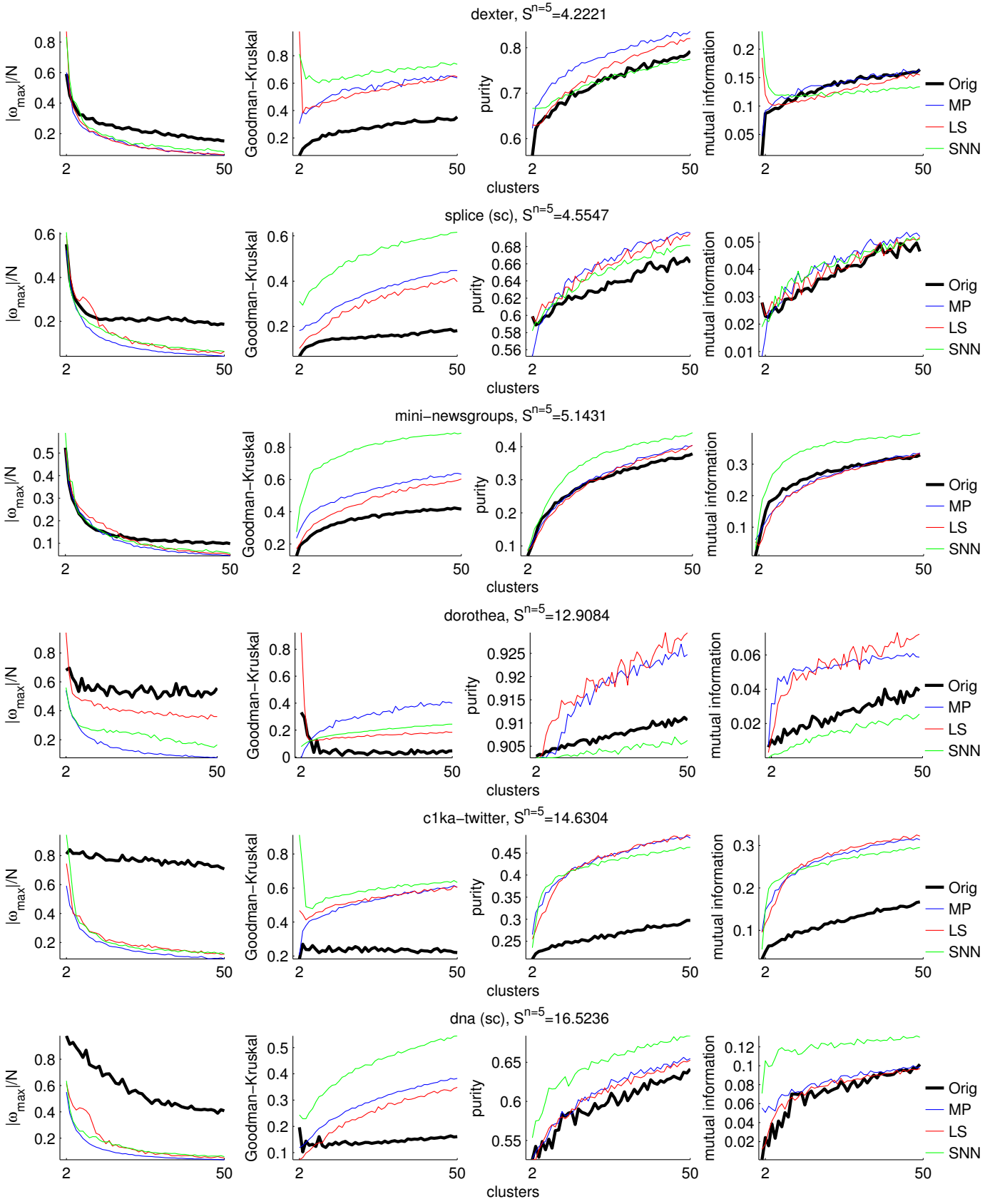

Fig. 5. K–medoids clustering performance (averaged over 30 iterations, y-axis) versus the number of clusters (x-axis). The black line is the clustering computed from the original data; blue (MP), red (LS) and green (SNN).

ACKNOWLEDGMENT

This research is supported by the Austrian Science Fund (FWF, project P27082 and P27703).

REFERENCES

[1] M. Radovanović, A. Nanopoulos, and M. Ivanović, "Hubs in space: Popular nearest neighbors in high-dimensional data," *Journal of Machine Learning Research*, vol. 11, pp. 2487–2531, December 2010.

[2] D. Schnitzer, A. Flexer, M. Schedl, and G. Widmer, "Local and global scaling reduce hubs in space," *Journal of Machine Learning Research*, vol. 13, pp. 2871–2902, October 2012.

[3] I. Suzuki, K. Hara, M. Shimbo, Y. Matsumoto, and M. Saerens, "Investigating the effectiveness of laplacian-based kernels in hub reduction," in *Proceedings of the Twenty-Sixth Conference on Artificial Intelligence (AAAI)*, 2012, pp. 1112–1118.

[4] A. Flexer, D. Schnitzer, and J. Schlüter, "A mirex meta-analysis of hubness in audio music similarity." in *Proceedings of the 13th International Society for Music Information Retrieval Conference (ISMIR)*, 2012, pp. 175–180.

[5] D. Schnitzer, A. Flexer, and N. Tomasev, "A case for hubness removal in high-dimensional multimedia retrieval," in *Advances in Information Retrieval - 36th European Conference on IR Research, ECIR 2014, Proceedings*, 2014, pp. 687–692.

[6] A. Flexer and D. Schnitzer, "Using mutual proximity for novelty detection in audio music similarity," in *6th International Workshop on Machine Learning and Music (MML), In conjunction with the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML/PKDD)*, Prague, Czech Republic, 2013.

[7] N. Tomasev, M. Radovanovic, D. Mladenic, and M. Ivanovic, "The role of hubness in clustering high-dimensional data," *IEEE Transactions on Knowledge and Data Engineering*, vol. 26, no. 3, pp. 739–751, 2014.

[8] A. Banerjee and J. Ghosh, "On scaling up balanced clustering algorithms," in *In Proceedings of the SIAM International Conference on Data Mining*, 2002.

[9] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2005, vol. 17, pp. 1601–1608.

[10] R. Jarvis and E. A. Patrick, "Clustering using a similarity measure based on shared near neighbors," *IEEE Transactions on Computers*, vol. 22, pp. 1025–1034, 1973.

[11] R. Bellman, *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

[12] D. François, V. Wertz, and M. Verleysen, "The concentration of fractional distances," *IEEE Transactions on Knowledge and Data Engineering*, vol. 19, pp. 873–886, 2007.

[13] C. Aggarwal, A. Hinneburg, and D. Keim, "On the surprising behavior of distance metrics in high dimensional space," in *Database Theory - ICDT 2001*, ser. Lecture Notes in Computer Science. Springer Berlin/Heidelberg, 2001, pp. 420–434.

[14] H.-P. Kriegel, P. Kröger, and A. Zimek, "Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering," *ACM Transactions on Knowledge Discovery from Data*, vol. 3, no. 1, pp. 1:1–1:58, Mar. 2009.

[15] P. Bradley, K. Bennett, and A. Demiriz, "Constrained k-means clustering," *Microsoft Research, Redmond*, pp. 1–8, 2000.

[16] S. Zhong and J. Ghosh, "Scalable, balanced model-based clustering." in *SDM*. SIAM, 2003, pp. 71–82.

[17] ——, "Model-based clustering with soft balancing," in *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*. IEEE, 2003, pp. 459–466.

[18] A. Banerjee and J. Ghosh, "Frequency-sensitive competitive learning for scalable balanced clustering on high-dimensional hyperspheres," *Neural Networks, IEEE Transactions on*, vol. 15, no. 3, pp. 702–719, 2004.

[19] X. Li, G. Yu, and D. Wang, "Mmpclust: A skew prevention algorithm for model-based document clustering," in *Database Systems for Advanced Applications*. Springer, 2005, pp. 536–547.

[20] M. I. Malinen and P. Fränti, "Balanced k-means for clustering," in *Structural, Syntactic, and Statistical Pattern Recognition*. Springer, 2014, pp. 32–41.

[21] M. E. Houle, H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek, "Can Shared-Neighbor Distances Defeat the Curse of Dimensionality? ," in *Scientific and Statistical Database Management*, ser. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin / Heidelberg, 2010, vol. 6187, ch. 34, pp. 482–500.

[22] N. Tomašev and D. Mladenić, "Hubness-aware shared neighbor distances for high-dimensional k-nearest neighbor classification," in *Hybrid Artificial Intelligent Systems*. Springer, 2012, pp. 116–127.

[23] A. Flexer and D. Schnitzer, "Can shared nearest neighbors reduce hubness in high-dimensional spaces?" in *Proceedings of 1st International Workshop on High Dimensional Data Mining (HDM), in conjunction with the IEEE International Conference on Data Mining (IEEE ICDM 2013)*, Dec 2013, pp. 460–467.

[24] K. Hara, I. Suzuki, M. Shimbo, K. Kobayashi, K. Fukumizu, and M. Radovanovic, "Localized centering: Reducing hubness in large-sample data," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*. AAAI Press, 2015, pp. 2645–2651.

[25] D. Schnitzer and A. Flexer, "Choosing the metric in high-dimensional spaces based on hub analysis," in *22th European Symposium on Artificial Neural Networks, ESANN 2014, Bruges, Belgium, April 23-25*, 2014.

[26] A. Flexer and D. Schnitzer, "Choosing l^p norms in highdimensional spaces based on hub analysis," *Neurocomputing*, 2015, to appear, available online 18 April 2015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231215004336

[27] L. Rousseeuw and L. Kaufman, "Clustering by means of medoids," *Statistical data analysis based on the L1-norm and related methods*, vol. 405, 1987.

[28] S. Günter and H. Bunke, "Validation indices for graph clustering," *Pattern Recognition Letters*, vol. 24, no. 8, pp. 1107 – 1113, 2003.

[29] K. P. Bennett, U. Fayyad, and D. Geiger, "Density-based indexing for approximate nearest-neighbor queries," in *Proceedings of the 5th International Conference on Knowledge Discovery and Data mining (ACM SIGKDD)*, ser. KDD '99. New York, NY, USA: ACM, 1999, pp. 233–243.

[30] C. D. Manning, P. Raghavan, and H. Schütze, *Introduction to information retrieval*. Cambridge University Press Cambridge, 2008, vol. 1.

[31] A. Strehl and J. Ghosh, "Impact of similarity measures on web-page clustering," in *Workshop on Artificial Intelligence for Web Search (AAAI)*, 2000, pp. 58–64.

[32] A. Frank and A. Asuncion, "UCI machine learning repository," 2010, repository located at: http://archive.ics.uci.edu/ml.

[33] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[34] M. Schedl, "On the Use of Microblogging Posts for Similarity Estimation and Artist Labeling," in *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)*, Utrecht, the Netherlands, August 2010.

[35] E. Levina and P. J. Bickel, "Maximum likelihood estimation of intrinsic dimension," in *Advances in Neural Information Processing Systems 17*. Cambridge, MA: MIT Press, 2005, pp. 777–784.