Hubness: An Interesting Property of Nearest Neighbor Graphs and its Impact on Classification

Krisztian Buza*

BioIntelligence Lab
Genomic Medicine and Rare Disorders
Semmelweis University
Tömő utca 25-29, 1083 Budapest, Hungary
buza@biointelligence.hu

Abstract: The presence of hubs, i.e., a few vertices that appear as neighbors of surprisingly many other vertices, is a recently explored property of nearest neighbor graphs. Several Authors argue that the presence of hubs should be taken into account for various data mining tasks, such as classification, clustering or instance selection. In this paper, we review recent works on hubness-aware instance selection for classification. We refer to applications of the reviewed techniques, such as time series classification or the analysis of biomedical data.

Keywords: nearest neighbor graphs, hubs, data mining, machine learning

1 Introduction

Most prominent data mining tasks include classification and clustering. Classification is the common denominator of various recognition tasks such as signature verification [1], [2], speech and handwriting recognition [3], [4], [5]. For example, in case of handwriting recognition, the user is writing a symbol on the touch screen of a tablet or smartphone while the device is recording the tip's position at consecutive moments of time, e.g. 100 times per second. The positions can be described quantitatively by the horizontal and vertical coordinates of the points where the screen is touched. This results in a sequence of measured numerical values (horizontal and vertical coordinates). Based on this information, the device aims to automatically recognize which character was written by the user.

Similar recognition tasks arise in various other domains. For example, biomedical devices, such as electroencephalograph (EEG) and electrocardiograph (ECG) capture the electrical activity of the brain and heart respectively, and based on the data recorded by these devices, one may try to recognize or predict the events related to diseases such as epileptic seizures or irregular heart beats [6], [7]. Furthermore, one may try to recognize subtypes of diseases based on gene expression data [8]. Further recognition tasks of the same type are related to texts and images [9], [10].

In order to solve the aforementioned recognition and prediction tasks, due to the large amount of underlying data and/or the required recognition speed, human experts usually need to be assisted by automated recognition systems. Many state-of-the-art solutions are based on machine learning: a recognition model, called *classifier*, is constructed based on previously collected data and evidence (such as which sequence of positions corresponds to which handwritten symbol, or which medical signal corresponds to which disease, where are the symptoms of that disease expressed in the data).

Nearest neighbor classifiers* are among the most promising classifiers due to multiple reasons. First, there are theoretical performance guarantees for the accuracy of nearest neighbor models [11], [12]. Second, they have been shown to be competitive, if not superior, to many other, more complex classifiers in

*Discussions with Dr. Alexandros Nanopoulos and Dr. Nenad Tomašev are highly appreciated. This research was performed within the framework of the grant of the Hungarian Scientific Research Fund - OTKA 111710 PD and OTKA 108947 K. This paper was supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences.

*We will describe nearest neighbor classifiers in the subsequent section in more detail.

various applications, see e.g. [13] and the references therein. Third, unlike in case of many other classifiers, nearest neighbors deliver human-understandable explanations for their predictions in form of a set of *similar* instances (i.e., *similar* signals, text, images, gene expression vectors, etc., depending on the current application). Fourth, in order to perform nearest neighbor classification, solely an appropriate distance (or similarity) measure between the instances of the dataset is required, the instances do not need to be represented in a vector space. For example, dynamic time warping [5], Levenshtein distance [14] and Smith-Waterman [15] distance work directly on time series, texts and genetic sequences respectively, without the need for representing the data in a vector space. Moreover, nearest neighbor classifiers are intuitive and simple to implement, which may be relevant aspects in real-world applications. Therefore, we focus on nearest neighbor classification, and its hubness-aware extensions in this talk.

2 Background

The problem of classification can be stated as follows. We are given a set of instances and some groups. The groups are called classes, and they are denoted as $C_1, \ldots, C_m$. Each instance x belongs to one of the classes. Whenever x belongs to class C_i, we say that the class label of x is C_i. We denote the set of all the classes by $\mathcal{C}$, i.e., $\mathcal{C} = \{C_1, \ldots, C_m\}$. Let $\mathcal{D}$ be a dataset of instances x_i and their class labels y_i, i.e., $\mathcal{D} = \{(x_1, y_1) \ldots (x_n, y_n)\}$. We are given a dataset $\mathcal{D}^{train}$, called *training data*. The task of classification is to induce a function $f(x)$, called *classifier*, which is able to assign class labels to instances not contained in $\mathcal{D}^{train}$.

In real-world applications, for some instances we know (from measurements and/or historical data) to which classes they belong, while the class labels of other instances are unknown. Based on the data with known classes, we induce a classifier, and use it to determine the class labels of the rest of the instances.

In experimental settings we usually aim at measuring the performance of a classifier. Therefore, after inducing the classifier using $\mathcal{D}^{train}$, we use a second dataset $\mathcal{D}^{test}$, called *test data*: for the instances of $\mathcal{D}^{test}$, we compare the output of the classifier, i.e., the predicted class labels, with the true class labels, and calculate the accuracy of classification. Therefore, the task of classification can be defined formally as follows: given two datasets $\mathcal{D}^{train}$ and $\mathcal{D}^{test}$, the task of classification is to induce a classifier $f(x)$ that maximizes prediction accuracy for $\mathcal{D}^{test}$. For the induction of $f(x)$, however, solely $\mathcal{D}^{train}$ can be used, but not $\mathcal{D}^{test}$.

Next, we describe the k-nearest neighbor classifier (kNN). Suppose, we are given an instance $x^* \in \mathcal{D}^{test}$ that should be classified. The kNN classifier searches for those k instances of the training dataset that are most similar to x^*. These k most similar instances are called the k nearest neighbors of x^*. The kNN classifier considers the k nearest neighbors, and takes the majority vote of their labels and assigns this label to x^*: e.g. if $k = 3$ and two of the nearest neighbors of x^* belong to class C_1, while one of the nearest neighbors of x belongs to class C_2, then this 3-NN classifier recognizes x^* as an instance belonging to the class C_1. We use $\mathcal{N}_k(x)$ to denote the set of k nearest neighbors of x. $\mathcal{N}_k(x)$ is also called as the k-neighborhood of x.

3 Presence of Hubs

The presence of hubs, i.e., instances that occur surprisingly frequently as neighbors of other instances, has been observed in various natural and artificial networks, such as protein-protein-interaction networks or the internet [16]. In case of nearest neighbor classification, we consider the nearest neighbor graph, in which vertices correspond to instances of the dataset and there is an edge from vertex v_x to vertex v_z if instance z is one of the k nearest neighbors of instance x. An example for such a nearest neighbor graph with $k = 1$ is shown in Figure 1. The presence of hubs in nearest neighbor graphs has been confirmed in various contexts, such as text mining, music retrieval and recommendation, image data and time series [17, 18, 19, 20, 21].

In context of classification, hubness was discussed in [22, 23, 24]. The property of hubness states that for data with high (intrinsic) dimensionality, a few instances tend to become nearest neighbors surprisingly

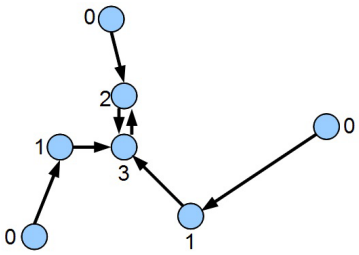

Figure 1: Nearest neighbor graph with $k = 1$. Directed edges point from each instance to its first nearest neighbor. The number next to each instance denotes the in-degree of the vertex, i.e., how many times the corresponding instance appears as nearest neighbor of *other* instances.

frequently, while other instances (almost) never appear as nearest neighbors. Intuitively speaking, very frequent neighbors, or hubs, dominate the neighbor sets and therefore, in the context of similarity-based learning, they represent the centers of influence within the data. For example in Figure 1 the instance that appears as nearest neighbor of three other instances can be considered as a hub. In contrast to hubs, there are instances that rarely occur as nearest neighbors of other instances. Such contributing little to the analytic process. We will refer to them as *orphans* or *anti-hubs*.

In order to express hubness more precisely, for a dataset $\mathcal{D}$ one can define the *k-occurrence* of an instance $x from \mathcal{D}$, denoted by $N_k(x)$, as the number of other instances in $\mathcal{D}$ having x among their k nearest neighbors:

$$N_k(x) = |\{x_i | x \in \mathcal{N}_k(x_i)\}|. \tag{1}$$

With the term *hubness* we refer to the phenomenon that the distribution of $N_k(x)$ becomes significantly skewed to the right. We can measure this skewness, denoted by $\mathcal{S}_{N_k(x)}$, with the standardized third moment of $N_k(x)$:

$$\mathcal{S}_{N_k(x)} = \frac{E[(N_k(x) - \mu_{N_k(x)})^3]}{\sigma^3_{N_k(x)}} \tag{2}$$

where $\mu_{N_k(x)}$ and $\sigma_{N_k(x)}$ are the mean and standard deviation of the distribution of $N_k(x)$. When $\mathcal{S}_{N_k(x)}$ is higher than zero, the corresponding distribution is skewed to the right and starts presenting a long tail.

In the presence of class labels, we distinguish between *good hubness* and *bad hubness*: we say that the instance x' is a *good k-nearest neighbor* of the instance x, if (i) x' is one of the k-nearest neighbors of x, and (ii) both have the same class labels. Similarly: we say that the instance x' is a *bad k-nearest neighbor* of the instance x, if (i) x' is one of the k-nearest neighbors of x, and (ii) they have different class labels. This allows us to define *good (bad) k-occurrence* of an instance x, $GN_k(x)$ (and $BN_k(x)$ respectively), which is the number of other instances that have x as one of their good (bad respectively) k-nearest neighbors. According to empirical results, both distributions $GN_k(x)$ and $BN_k(x)$ are usually skewed, as it is exemplified in Fig. 2, which depicts the distribution of $GN_1(x)$ for some time series datasets from the UCR collection[†]. As shown, the distributions have long tails in which the good hubs occur.

We say that an instance x is a good (or bad) hub, if $GN_k(x)$ (or $BN_k(x)$ respectively) is exceptionally large for x. For nearest neighbor classification, the skewness of good occurrence is of particular importance, because some few instances are responsible for large portion of the overall error: bad hubs tend to misclassify a surprisingly large number of other instances [21]. Therefore, one has to take into account the presence of good and bad hubs. While the kNN classifier is frequently used for time series classification, the k-nearest neighbor approach is also well suited for learning under class imbalance, see e.g. [10] and the references therein, therefore hubness is relevant for the classification of imbalanced data too.

As hubs appear in data with high (intrinsic) dimensionality, hubness is one of the recently explored aspects of the curse of dimensionality [22], [23], [24]. However, dimensionality reduction can not entirely

[†]http://www.cs.ucr.edu/~eamonn/time_series_data/

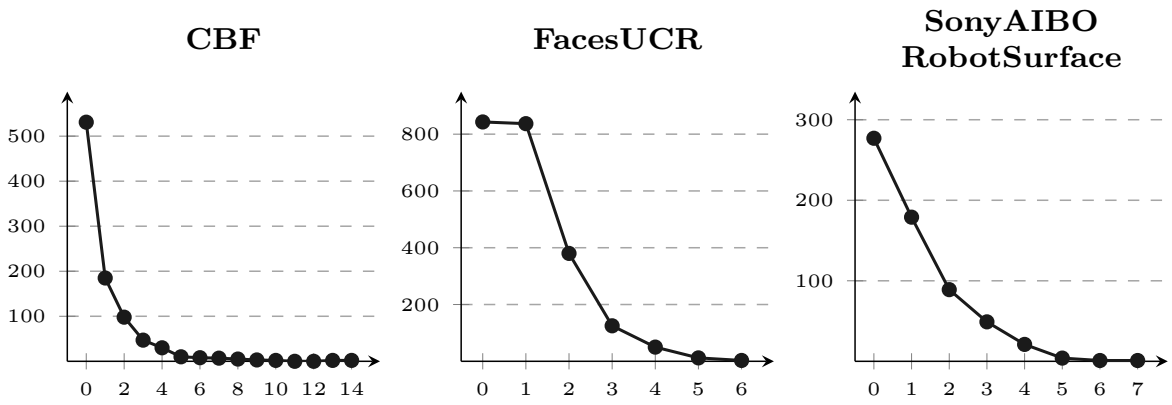

Figure 2: Distribution of $GN_1(x)$ for some time series datasets. The horizontal axis corresponds to the values of $GN_1(x)$, while on the vertical axis one can see how many instances have that value.

eliminate the issue of bad hubs, unless it induces significant information loss by reducing to a very low dimensional space - which often ends up hurting system performance even more [22], [25].

4 Hubness-aware Instance Selection

One drawback of nearest neighbor classifiers is that in case of large datasets it may become computationally expensive to classify instances if the instance to be classified has to be compared with all the instances of the training dataset, especially if the distance measure is computationally expensive such as dynamic time warping in case of time-series classification [13].

Attempts to speed up nearest neighbor classification fall into four major categories: i) speed-up the calculation of the distance measure, ii) indexing, iii) manipulation of the instances (such as reduction of the length of time series in case of time-series classification), and iv) instance selection. As discussed in [13], these approaches are orthogonal, i.e., they can be combined with each other. Here, we focus on instance selection.

Instance selection (also known as *numerosity reduction* or *prototype selection*) aims at discarding most of the training time series while keeping only the most informative ones, which are then used to classify unlabeled instances. Previous works on instance selection for nearest-neighbor classification include [26], [27], [28], [29], [30], [31].

Our hubness-aware instance selection technique, INSIGHT [32], performs instance selection by assigning a score to each instance and selecting instances with the highest scores. Therefore the "intelligence" of INSIGHT is hidden in the applied score function. In this section, we explain the suitability of score functions in the light of the hubness property.

Good 1-occurrence Score INSIGHT can use scores that take the good 1-occurrence of an instance x into account. Thus, a simple score function is the *good 1-occurrence score* $g_G(x)$:

$$g_G(x) = GN_1(x) \tag{3}$$

Relative Score Even if an instance x is a good hub, it may appear as bad neighbor of several other instances. Thus, INSIGHT can also consider scores that take bad occurrences into account. This leads to scores that relate the good occurrence of an instance x to either its total occurrence or to its bad occurrence. For simplicity, we use the following *relative score*, however, other variants are possible too:

Relative score $g_R(x)$ of a time series x is the fraction of good 1-occurrences and total occurrences plus one (plus one in the denominator avoids potential division by zero):

$$g_R(x) = \frac{GN_1(x)}{N_1(x) + 1} \tag{4}$$

Xi's Score Interestingly, $GN_k(x)$ and $BN_k(x)$ allows us to interpret the ranking criterion used by Xi et al. in FastAWARD [31] as another form of score for relative hubness:

$$g_{Xi}(x) = GN_1(x) - 2BN_1(x) \tag{5}$$

Despite its simplicity, INSIGHT was reported to achieve surprisingly good classification accuracy, outperforming the FastAWARD instance selection technique both in terms of accuracy and runtime [32].

5 Hubness-aware classification and clustering

The phenomenon of hubness may also be taken into account in order to increase classification accuracy. Therefore, hubness-aware classifiers have been proposed, such as hw-kNN [21], [23], Hubness-based Fuzzy Nearest Neighbor (HFNN) [36], Naive Hubness Bayesian Nearest Neighbor (NHBNN) [37], and Hubness Information k-Nearest Neighbor (HIKNN) [38]. These classifiers were surveyed in [39]. Instance selection has been studied in context of the aforementioned hubness-aware classifiers and it has been found that parameters, such as $N_k(x)$, $GN_k(x)$ and $BN_k(x)$ are worth to be estimated on the *entire* dataset, however, at classification time it is enough if the classifiers work with the selected instances which results in computationally less expensive classification [40]. Hubness-aware classifiers were studied in case of class-imbalanced and noisy data [10],[41]. According to recent results, hubness seems to be relevant in context of semi-supervised classification and ensemble learning [42],[43]. Regarding further data mining tasks, we mention that clustering algorithms have been introduced recently [44]. Implementations of hubness-aware data mining algorithms are available in the HubMiner library at

http://ailab.ijs.si/nenad_tomasev/hub-miner-library/
and
https://github.com/datapoet/hubminer .

6 Conclusion and Outlook

In this paper, we described hubness as an interesting property of nearest neighbor graphs. We argued that this property, especially the presence of bad hubs, should be taken into account for various data mining tasks and reviewed the most relevant works in the literature, many of them being empirical studies. Studies on the theoretical foundations of hubness focus on the intrinsic dimensionality of the datasets. Much less is known about the mechanisms that generate nearest neighbor graphs containing hubs: although, based on preferential selection, Barabasi gave a generative model which is able to explain many natural networks containing hubs [16], nearest neighbor graphs are special because each node has an out-degree of k which is not captured by the aforementioned generative model. The author is not aware of a generative model that are able to properly describe nearest neighbor graphs containing hubs.

References

[1] C. Gruber, M. Coduro, and B. Sick, Signature Verification with Dynamic RBF Networks and Time Series Motifs, *10th International Workshop on Frontiers in Handwriting Recognition* (2006)

[2] R. Martens and L. Claesen, On-line Signature Verification by Dynamic Time-Warping, *Proceedings of the 13th International Conference on Pattern Recognition* (1996) **3**, pp. 38–42. IEEE

[3] R. Niels, Dynamic Time Warping: an Intuitive Way of Handwriting Recognition? *Master Thesis, Radboud University Nijmegen, The Netherlands* (2004)

[4] R. Plamondon and S.N. Srihari, Online and Off-line Handwriting Recognition: a Comprehensive Survey, *Pattern Analysis and Machine Intelligence* (2002) **22(1)** pp. 63–84

[5] H. Sakoe and S. Chiba, Dynamic Programming Algorithm Optimization for Spoken Word Recognition, *Acoustics, Speech and Signal Processing* (1978) **26(1)** pp. 43–49

[6] K. Buza, J. Koller, K. Marussy, PROCESS: Projection-based Classification of Electroencephalograph Signals, *14th International Conference on Artificial Intelligence and Soft Computing, Lecture Notes in Artificial Intelligence* (2015)

[7] K. Buza, A. Nanopoulos, L. Schmidt-Thieme, J. Koller, Fast Classification of Electrocardiograph Signals via Instance Selection, *First IEEE Conference on Healthcare Informatics, Imaging, and Systems Biology* (2011)

[8] W.-J. Lin, J.J. Chen, Class-imbalanced classifiers for high-dimensional data, *Briefings in Bioinformatics* (2012) **bbs006**

[9] N. Tomasev, J. Rupnik, D. Mladenic, The Role of Hubs in Cross-Lingual Supervised Document Retrieval, *Advances in Knowledge Discovery and Data Mining, Lecture Notes in Computer Science Volume* (2013) **7819** pp. 185–196

[10] N. Tomasev, D. Mladenic, Class imbalance and the curse of minority hubs, *Knowledge-Based Systems* (2013) **53** pp. 157–172

[11] L. Devroye, L. Györfi, G. Lugosi, *A probabilistic theory of pattern recognition*, Springer (1996)

[12] G.H. Chen, S. Nikolov, D. Shah, A latent source model for nonparametric time series classification, *Advances in Neural Information Processing Systems* (2013)

[13] K. Buza, *Fusion methods for time-series classification*, Peter Lang Verlag (2011)

[14] V.I. Levenshtein, Binary codes capable of correcting deletions, insertions, and reversals, *Soviet physics doklady* (1966) **10(8)**

[15] T.F. Smith, M.S. Waterman, Identification of common molecular subsequences, *Journal of molecular biology* (1981) **147.1** pp. 195–197

[16] A.-L. Barabasi, *Linked: How everything is connected to everything else and what it means*, Plume Editors (2002)

[17] N. Tomašev, *The Role of Hubness in High-Dimensional Data Analysis*, PhD thesis, Jožef Stefan International Postgraduate School (2013)

[18] A. Flexer, D. Schnitzer, Can shared nearest neighbors reduce hubness in high-dimensional spaces? *IEEE International Conference on Data Mining Workshops* (2013)

[19] A. Flexer, D. Schnitzer, J. Schlüter, A MIREX Meta-analysis of Hubness in Audio Music Similarity *13th International Society for Music Information Retrieval Conference* (2012)

[20] D. Schnitzer, A. Flexer, N. Tomašev, A Case for Hubness Removal in High-Dimensional Multimedia Retrieval *Advances in Information Retrieval* (2014) pp. 687–692.

[21] M. Radovanović, A. Nanopoulos, M. Ivanović, Time-Series Classification in Many Intrinsic Dimensions, *10th SIAM International Conference on Data Mining* (2010), pp. 677–688

[22] M. Radovanović, *Representations and metrics in high-dimensional data mining*, Izdavacka knjižarnica Zorana Stojanovica, Novi Sad, Serbia (2011)

[23] M. Radovanović, A. Nanopoulos, M. Ivanović, Nearest neighbors in high-dimensional data: The emergence and influence of hubs, *Proceedings of the 26th Annual International Conference on Machine Learning* (2009)

[24] M. Radovanović, A. Nanopoulos, M. Ivanović, Hubs in space: Popular nearest neighbors in high-dimensional data, *The Journal of Machine Learning Research* (2010) **11** pp. 2487–2531

[25] K. Marussy, The Curse of Intrinsic Dimensionality in Genome Expression Classification, *Students' Scientific Conference, Budapest University of Technology and Economics* (2014)

[26] D.W. Aha, D. Kibler, M.K. Albert, Instance-Based Learning Algorithms, *Machine Learning* (1991) **6(1)** pp. 37–66

[27] H. Brighton and C. Mellish, Advances in Instance Selection for Instance-Based Learning Algorithms, *Data Mining and Knowledge Discovery* (2002) **6(2)** pp. 153–172

[28] N. Jankowski and M. Grochowski, Comparison of Instance Selection Algorithms I. Algorithms Survey, *Artificial Intelligence and Soft Computing-ICAISC 2004* (2004) pp. 598–603

[29] M. Grochowski and N. Jankowski, Comparison of Instance Selection Algorithms II. Results and Comments, *Artificial Intelligence and Soft Computing-ICAISC 2004* (2004) pp. 580–585

[30] H. Liu and H. Motoda, On issues of Instance Selection, *Data Mining and Knowledge Discovery* (2002) **6(2)** pp. 115–130

[31] X. Xi, E. Keogh, C. Shelton, L. Wei, and C.A. Ratanamahatana, Fast Time Series Classification Using Numerosity Reduction, *23rd International Conference on Machine Learning* (2006) pp. 1033–1040

[32] K. Buza, A. Nanopoulos, L. Schmidt-Thieme, INSIGHT: Efficient and effective instance selection for time-series classification, *Advances in Knowledge Discovery and Data Mining* (2011) pp. 149–160

[33] S. Ougiaroglou, A. Nanopoulos, A. Papadopoulos, Y. Manolopoulos, T. Welzer-Druzovec, Adaptive k-Nearest-Neighbor Classification Using a Dynamic Number of Nearest Neighbors, *Advances in Databases and Information Systems* (2007) pp. 66–82

[34] D. Wettschereck and T.G. Dietterich, Locally Adaptive Nearest Neighbor Algorithms. *Advances in Neural Information Processing Systems* (1994) pp. 184–184

[35] K. Buza, A. Nanopoulos, and L. Schmidt-Thieme, Time-Series Classification Based on Individualised Error Prediction, *International Conference on Computational Science and Engineering* (2010)

[36] N. Tomašev, M. Radovanović, D. Mladenić, M. Ivanović, Hubness-based fuzzy measures for high-dimensional k-nearest neighbor classification, *International Journal of Machine Learning and Cybernetics* (2013)

[37] N. Tomašev, M. Radovanović, D. Mladenić, M. Ivanović, A probabilistic approach to nearest neighbor classification: Naive hubness Bayesian k-nearest neighbor, *Conference on Information and Knowledge Management* (2011)

[38] N. Tomašev, D. Mladenić, Nearest neighbor voting in high dimensional data: Learning from past occurrences. *Computer Science and Information Systems* (2012) **9**, pp. 691–712

[39] N. Tomasev, K. Buza, K. Marussy, P.B. Kis, Hubness-aware Classification, Instance Selection and Feature Construction: Survey and Extensions to Time-Series, *Feature selection for data and pattern recognition* (2015), pp. 231–262

[40] N. Tomasev, K. Buza, Correcting the Hub Occurrence Prediction Bias in Many Dimensions, *Computer Science and Information Systems (under review)*

[41] N. Tomasev, K. Buza, Hubness-aware kNN classification of high-dimensional data in presence of label noise, *Neurocomputing* (2015)

[42] K. Marussy, K. Buza, Hubness-based indicators for semi-supervised time-series classification, *8th Japanese-Hungarian Symposium on Discrete Mathematics and Its Applications* (2013)

[43] N. Tomašev Boosting for Vote Learning in High-Dimensional kNN Classification, *IEEE International Conference on Data Mining Workshops* (2014)

[44] N. Tomašev, M. Radovanović, D. Mladenić, M. Ivanović, Hubness-based clustering of high-dimensional data, *Partitional Clustering Algorithms* (2015) pp. 353-386