Informed initial pivot selection for approximating and eliminating search algorithms

technical report

Daniel Blank and Andreas Henrich
Media Informatics Group
University of Bamberg, Germany
{daniel.blank | andreas.henrich}@uni-bamberg.de

11th December 2015

Abstract

The ever increasing amount of data and the growing diversity in data types requires effective and efficient retrieval techniques. With the diversity in data types also a variety of techniques for measuring the (dis)similarity between the data objects arises. Metric access methods (MAMs) make no assumption about the data representation. MAMs only require the dissimilarity function to be a metric and thus support a flexible indexing paradigm. However, distance metrics can be expensive to compute. Thus, a main goal of MAMs is to reduce the number of distance computations.

We replace the random initial pivot selection step of the approximating and eliminating search algorithm (AESA), a MAM capable to dramatically reduce the number of required distance computations. Our approach relies on the concept of the intrinsic dimension and comes with no additional cost during search. Besides showing that the performance of AESA can be improved, we also present a conceptually simple technique to measure the influence of intrinsic dimension definitions on search efficiency. This opens doors for the analysis of alternative intrinsic dimension definitions in future work.

1 Introduction

It is hard to design efficient indexing techniques for general distance/dissimilarity spaces (cf. [1]). However, many similarity search problems can be modeled in metric space. Here, the underlying dissimilarity space is a metric space where the distance function *dist* satisfies the metric properties. Metric access methods (MAMs) target the indexing of arbitrary metric spaces.

The variety of application areas for MAMs ranges from content-based media retrieval (e.g. search for unstructured and structured text, image, and video)

to search for protein sequences, business process traces and models, 3D object models, or function-call graphs for detecting malware programs, to name only a few. The distance metrics applied in these application domains are often more complex in their computation than the standard Minkowski distance metrics. Thus, the primary goal of MAMs is to reduce the number of distance computations—disk I/O and other cost factors are often of secondary importance [2].

The family of MAMs can be broadly classified by applying the concepts of *pivoting* and *aggregation* [3]. MAMs using *pivoting* store distances between database objects and reference objects (also called pivots) and prune database objects during search through *pivot filtering* based on the pre-computed object-to-pivot distances. Approaches using *aggregation*—occasionally in addition to *pivoting*—structure the feature space into multiple (possibly overlapping) regions in order to prune irrelevant regions during search.

Among these approaches, the approximating and eliminating search algorithm (AESA) [4] is known to be able to index datasets with the minimum number of distance computations during search at the cost of a quadratic space and construction time complexity. Thus, AESA is only applicable in application settings with a relatively small database size when the time complexity of the distance measure is extraordinary high. Such an application scenario could for example be the high precision search for media objects on cell phones.

The goal of this paper is twofold. First, it tries to improve the efficiency of AESA in terms of reducing the number of required distance computations. Second, by applying a prominent quantitative definition of the intrinsic dimension for replacing the random initial pivot selection step of AESA, the approach presented in this paper provides an adequate test bed for the application of intrinsic dimension definitions within MAMs. The "elusive concept" [5, *p.* 273] of the intrinsic dimension thus becomes more tangible.

Our proposal is to explicitly apply the concept of the intrinsic dimension for replacing the uninformed initial pivot selection step in the AESA. The paper is structured as follows. Section 2 presents the AESA, some of its variants and improvements relevant to our work, and a frequently used intrinsic dimension definition exemplary applied here. Section 3 discusses our AESA variant and shows preliminary results. Section 4 gives an outlook on future work.

2 Related Work

The following section 2.1 outlines pivot filtering and how it is used by the AESA for efficient search. Improvements to AESA are briefly discussed in section 2.2. Afterwards, in section 2.3 a frequently applied definition of the intrinsic dimension—often referred to as a concept for describing the difficulty of an indexing task—is introduced.

2.1 Pivot filtering and AESA

MAMs applying pivoting usually store $dist(p_i, o_j)$ values for multiple database objects $o_j \in O$ and pivots $p_i \in P$. Hence, when searching for all objects with a distance of at most r to a query object q the potentially expensive computation of $dist(q, o_j)$ can be skipped if the condition in formula 1 based on the triangle inequality is fulfilled.

$$\underbrace{\max_{p_i} |dist(p_i, q) - dist(p_i, o_j)|}_{g(o_j, q)} > r \qquad \textit{pivot filtering} \tag{1}$$

Fig. 1 visualizes this constraint. On the one hand, $dist(p_1, q) - dist(p_1, o_j) > r$ holds for database objects inside the inner white ball around center p_1 which has radius $dist(p_1, q) - r$. On the other hand, $dist(p_1, o_j) - dist(p_1, q) > r$ holds for database objects outside the outer ball around p_1 with radius $dist(p_1, q) + r$. Thus, $|dist(p_1, q) - dist(p_1, o_j)| \leq r$ holds for database objects which lie inside the shell containing the query ball. These objects cannot be pruned from search based on p_1. When applying further pivots, the region of possible database objects within the search radius can be restricted by intersections of multiple shells [3].

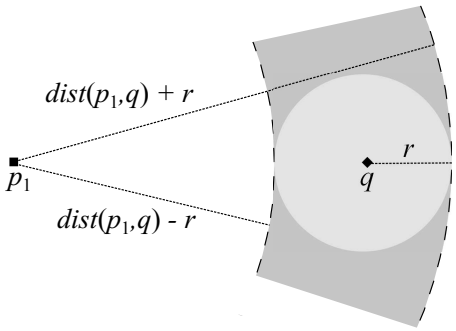

Figure 1: Pivot filtering (adapted from [3]).

The AESA [4] is known as the MAM with the minimum number of required distance computations. According to [6, *p.* 1516], it "has been for 20 years the baseline method in terms of saved distance computations". AESA relies on pivot filtering where every database object $o_j \in O$ $(1 \leq j \leq n)$ can potentially become a pivot. To do so, a distance matrix $D = \mathcal{R}^{n \times n}$ with $n(n-1)/2$ pairwise distances is maintained. Thus, space and construction time complexities of AESA are $O(n^2)$.

Algorithm 1 (notation based on [6]) outlines a search for the nearest neighbor (1-NN search). It starts with an infinite search radius r which is reduced in several rounds in order to find the 1-NN. $g(o_j, q)$ which captures the left hand side of inequation 1 is initially set to 0 for all $o_j \in O$ (lines 3–4). As shown in line 6, the search algorithm iterates over the database objects as long as not all of

them have been considered. A new pivot object is selected in each round (line 7). Usually, as can be seen in the `getNextPivot` procedure, the database object with the smallest bound $g(o_j, q)$ is considered next, i.e. $\arg\min_{o_j \in O} g(o_j, q)$. However, initially in the first round, an arbitrary decision is made, since $g(o_j, q) = 0$ for all $o_j \in O$ in this case. The selected pivot is then removed from the set of database objects to consider (line 8). For the selected pivot, the distance between the pivot and the query object is computed (line 9). If this distance $d_{current}$ is smaller than the current search radius, a new intermediate 1-NN has been found, i.e. the pivot selected in this round. Thus nn and r are updated (lines 11-12). From line 14 onward, $g(o_j, q)$ is updated and pivot filtering is applied to prune as many database objects as possible in each round.

Algorithm 1: The AESA 1-NN algorithm.

Input: $dist$: the distance metric
O: the database
q: the query
$D = \mathcal{R}^{|O| \times |O|}$: the matrix of pairwise distances
Output: $nn \in O$: the nearest neighbor to q

```
1  Algorithm search1NN()
2      r = ∞
3      foreach o_j ∈ O do
4          g(o_j, q) = 0
5      round = 1
6      while O ≠ ∅ do
7          p = getNextPivot(round)
8          O = O − {p}
9          d_current = dist(q, p)
10         if d_current < r then
11             nn = p
12             r = d_current
13         foreach o_j ∈ O do
14             g(o_j, q) = max(g(o_j, q), |D(o_j, p) − d_current|)
15             if g(o_j, q) ≥ r then
16                 O = O − {o_j}
17         round = round + 1
18     return nn
19 Procedure getNextPivot(round)
20     if round == 1 then
21         p = getRandomObject()
22     else
23         p = arg min_{o_j ∈ O} g(o_j, q)
24     return p
```

2.2 Improvements to the AESA

The Linear AESA (LAESA) [7] is proposed to overcome the quadratic space and construction time complexities of the AESA. In contrast to the AESA, the LAESA applies a set of $m \ll |O|$ pivots. Only object-to-pivot distances are stored and used for pruning database objects from search by applying pivot filtering. It is thus important to choose good pivots. There is plenty of work on pivot selection in the field of MAMs addressing the question which pivots to use (for a brief overview see e.g. [8]). Another important aspect is to determine the order in which the pivots are applied. Both, LAESA and AESA start with a random selection of the first pivot. It is important to note that this is the problem we address in this paper—*the initial selection of the first pivot*. Our technique is thus also applicable to LAESA and its multiple improvements. An analysis in this regard is part of future work. The task of selecting further pivots in future rounds of the algorithm (e.g. pivots close to the query [4, 7]; or both, close to or far from the query in an alternate fashion [9]) is beyond the scope of our present work. Furthermore, we do not address coarsening approaches, i.e. storing distance values with less precision, storing only some and not necessarily all of the object-to-pivot distances per database object, etc. (for references w.r.t. coarsening see [8]).

Two improvements to the original AESA which also operate on the complete distance matrix of all pairwise distances are proposed. iAESA [10] uses permutations of pivot IDs maintained per database object and chooses the object whose permutation list is most similar to the permutation list of the query as the next pivot. However, the random selection step in the first round remains. Experiments on real world datasets in [6] indicate that iAESA is less efficient than standard AESA in two out of three cases.

PiAESA [6] can be perceived as a procedure which precedes AESA. A set of R pivots is used upfront for estimating the query radius. Afterwards, PiAESA switches to standard AESA. The parameter $R \ll |O|$ needs to be adequately specified and experiments in [6] show that this is crucial. Otherwise, the performance of PiAESA degrades and can become worse than standard AESA. Several existing approaches for selecting the R pivots are tested in [6]. Interestingly, for the real world datasets, a random selection performs reasonably well. Here, our technique can also be applied. A comparison of our approach, standalone as well as integrated into iAESA and PiAESA, is future work.

2.3 The Intrinsic Dimension(ality)

The *intrinsic dimension(ality)* ρ, in opposition to the representational dimension δ of a δ-dimensional vector space, is an "elusive concept" [5, *p.* 273]. It is frequently used to quantify the difficulty of a metric space indexing task. It is also applied for determining the number of pivots to use within a MAM [11], for selectivity and performance estimation of MAMs [12], feature selection in vector datasets [13], and for estimating the query radius of k-NN queries [14]. To our knowledge, it has so far not been used for selecting the initial pivot

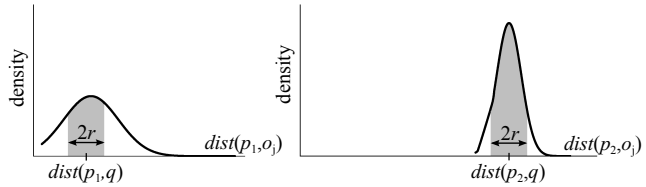

Figure 2: Two exemplary distance distributions (adapted from [5, *p.* 302]).

in approximating and eliminating search algorithms. Doing so allows for an empirical testing of different intrinsic dimension definitions (cf. sect. 3).

In this short paper, as a first step, we apply the widely used definition $\rho = \frac{\mu^2}{2\sigma^2}$ presented in [5, *p.* 303]. The computation of ρ relies on statistics obtained from a distance distribution; ρ increases if the mean μ of the distance distribution increases and/or the variance σ^2 shrinks. Figure 2 visualizes the rationale behind this definition. According to ρ, the histogram on the right reflects a higher intrinsic dimensionality than the histogram on the left. When performing a range query for q with a search radius r and applying pivot filtering, database objects $o_j \in O$ with $dist(p, o_j) \in [dist(p, q) - r, dist(p, q) + r]$ cannot be discarded. The amount of database objects which must be exhaustively searched is proportional to the gray shaded area in figure 2 which is in fact bigger in case of p_2 than p_1. Thus, if these histograms capture the distance distributions of two pivots p_1 and p_2, selecting p_1 might be the better choice. More objects can be discarded (being proportional to the white area under the curve) because of a larger variance with the histogram being less concentrated around its mean.

In addition, there are search scenarios where an increase of the mean distance μ requires a larger search radius r, for example when retrieving a fixed number of database objects. With other things being equal, an increase of r leads to fewer potential for object pruning since the gray shaded area under a curve increases. This supports the rationale why μ is contained in the numerator of the ρ formula [15, *p.* 370].

Alternative intrinsic dimension definitions such as the so called distance exponent [12] and the method based on principal component analysis proposed in [16] will be applied in future work. As an interesting finding opening doors for alternative intrinsic dimension definitions, Mao et al. [16] show that methods defined for $\mathcal{R}^n$ can be applied to AESA-like settings. Thus, techniques for determining the intrinsic dimension of vector spaces become applicable for MAMs.

3 Applying the Intrinsic Dimension to AESA

AESA stores all $\mathcal{O}(n^2)$ distances between database objects. Thus, we can compute ρ_j scores for every database object $o_j \in O$ based on all $dist(o_j, o_l)$ with

$1 \leq l \leq n$ and $j \neq l$. As explained earlier in section 2.3, we choose the first pivot o_j as the one with $\arg\min_j \rho_j$. This is a slight modification of line 21 of the `getNextPivot` procedure in algorithm 1. It is important to note that search cost is not affected, since $\arg\min_j \rho_j$ is computed during indexing. The construction cost of the index remains in $\mathcal{O}(n^2)$.

In our experiments, we test four distance metrics on four test collections resulting however in only 14 tested combinations, since two combinations are not applicable because the Hellinger distance cannot be applied on negative feature vector components. The distance metrics we tested are outlined in the first column of figure 3. As feature vectors, we applied subsets of four publicly available corpora:

NUS: Here we use parts of the image collection described in [17]. We apply the *Test_Normalized_CH.dat* file with 107,859 64-dimensional feature vectors.

MIRFLICKR: The Mirflickr collection can be found on `http://press.liacs.nl/mirflickr/`. We use the first 25,000 objects from `http://mirflickr.liacs.nl/ht_descriptors.zip`. The representational dimension of the dataset is 43.

COLORS: This dataset is made up of 112,682 feature vectors with 112 dimensions per vector. The dataset can be obtained from `http://www.sisap.org/`.

NASA: This corpus consists of 40,150 database objects represented as 20-dimensional feature vectors. The dataset can also be obtained from `http://www.sisap.org/`.

In our experiments, we tried to mimic the setting in [6]. We use a database with $|O| = 15{,}000$ feature objects randomly sampled from the resp. collection. 1,000 queries are performed in each run. We search for the nearest neighbor to the query (1-NN search). Queries are randomly chosen from the remaining objects of the resp. collection. We perform 10 runs with 1,000 queries each and measure the average number of performed distance computations.

Experimental results are displayed in figure 3. The gray shaded results are the ones of standard AESA with the random initialization step. In addition, on white background, we display results for initially selecting the database object o_j with $\arg\min_j \rho_j$ (top number of each white cell), a selection based on median values (middle number of each white cell), and $\arg\max_j \rho_j$ (bottom number of each white cell). The rationale is that the top figures should improve the standard AESA baseline, numbers in the middle should roughly indicate baseline performance, and numbers on the bottom of each white box should indicate clearly worse performance than randomly selecting the first pivot. The approach which initially selects $\arg\min_j \rho_j$ outperforms standard AESA in all 14 cases[1]. In 12 out of 14 cases this improvement is significant when performing a Student's t-test on a significance level of 0.5%. These cases are underlined in figure 3.

The best approach per cell is highlighted in bold face. An interesting approach in this regard is the *Manhattan-MIRFLICKR* combination which seems

[1]We visualized the differences in the distance distributions between $\arg\min_j \rho_j$ and $\arg\max_j \rho_j$ for one of the 14 scenarios. In fact, for a random run, $\arg\min_j \rho_j$ leads to the left distribution in figure 2 and $\arg\max_j \rho_j$ to the right distribution.

		NUS	*MIRFLICKR*	*COLORS*	*NASA*
Canberra	AESA	1111.1	148.0	3365.4	1130.1
	min ρ_j	**1109.7**	**147.5**	**3361.3**	**1129.2**
	median ρ_j	1111.2	147.9	3365.9	1130.2
	max ρ_j	1112.0	150.5	3366.5	1130.6
Hellinger	AESA	-	134.7	401.7	-
	min ρ_j		**134.4**	**398.0**	
	median ρ_j		134.8	401.9	
	max ρ_j		135.5	402.8	
Euclidean	AESA	209.6	137.0	124.3	45.6
	min ρ_j	**207.1**	**136.8**	**123.6**	**44.9**
	median ρ_j	209.6	137.0	124.0	45.7
	max ρ_j	212.1	137.7	126.6	46.2
Manhattan	AESA	157.7	151.2	149.9	46.5
	min ρ_j	**155.9**	151.1	149.0	**46.2**
	median ρ_j	157.5	151.3	**148.9**	46.5
	max ρ_j	160.5	**149.2**	151.0	46.8

Figure 3: Evaluation on four image collections with four distance metrics.

to invert retrieval results. It is part of future work to analyze this issue and figure out why the definition of ρ does not work in this case. We believe that this offers a promising test bed for the comparison of different intrinsic dimension definitions.

4 Conclusions

We presented an improvement to the random initialization step of approximating and eliminating search algorithms. Instead of randomly choosing the first pivot, we propose an informed selection based on the distribution of object-to-pivot distances and to apply intrinsic dimension definitions on this distribution. Our improvements are small, however they are significant and come at no additional search cost. In future work, we plan to analyze different intrinsic dimension definitions and their influence on search performance.

[16] show that methods defined for $\mathcal{R}^n$ can be applied to AESA-like settings. Thus, techniques for determining the intrinsic dimension of vector spaces become applicable to general metric spaces and for example principal component (PCA) analysis is used in [16] for pivot selection and the determination of the intrinsic dimension of a dataset. Inspired by [16], we plan to base the initial selection step of AESA on PCA. Another alternative intrinsic dimension definition, the so called distance exponent proposed in [12], will also be applied in future work.

Furthermore, we will apply our approach to other MAMs than AESA, such as for example LAESA and its improvements.

References

[1] T. Skopal and B. Bustos, "On nonmetric similarity search problems in complex domains," *ACM Comput. Surv.*, vol. 43, no. 4, pp. 34:1–34:50, Oct. 2011.

[2] T. Skopal, "Where are you heading, metric access methods?: A provocative survey," in *Proc. of the 3rd Intl. Conf. on SImilarity Search and APplications*. New York, NY, USA: ACM, 2010, pp. 13–21.

[3] M. Hetland, "The basic principles of metric indexing," in *Swarm Intelligence for Multi-objective Problems in Data Mining*, ser. Studies in Computational Intelligence, C. Coello, S. Dehuri, and S. Ghosh, Eds. Springer Berlin Heidelberg, 2009, vol. 242, pp. 199–232.

[4] E. V. Ruiz, "An algorithm for finding nearest neighbours in (approximately) constant average time," *Pattern Recogn. Lett.*, vol. 4, no. 3, pp. 145–157, Jul. 1986.

[5] E. Chávez, G. Navarro, R. Baeza-Yates, and J. L. Marroquín, "Searching in metric spaces," *ACM Comput. Surv.*, vol. 33, no. 3, pp. 273–321, Sep. 2001.

[6] R. Socorro, L. Micó, and J. Oncina, "A fast pivot-based indexing algorithm for metric spaces," *Pattern Recogn. Lett.*, vol. 32, no. 11, pp. 1511–1516, Aug. 2011.

[7] M. L. Micó, J. Oncina, and E. Vidal, "A new version of the nearest-neighbour approximating and eliminating search algorithm (aesa) with linear preprocessing time and memory requirements," *Pattern Recogn. Lett.*, vol. 15, no. 1, pp. 9–17, Jan. 1994.

[8] L. G. Ares, N. R. Brisaboa, M. F. Esteller, O. Pedreira, and A. S. Places, "Optimal pivots to minimize the index size for metric access methods," in *2nd Intl. Workshop on Similarity Search and Applications*. IEEE, 2009, pp. 74–80.

[9] C. Celik, "New approaches to similarity searching in metric spaces," Ph.D. dissertation, College Park, MD, USA, 2006, aAI3212572.

[10] K. Figueroa, E. Chavez, G. Navarro, and R. Paredes, "Speeding up spatial approximation search in metric spaces," *J. Exp. Algorithmics*, vol. 14, pp. 6:3.6–6:3.21, Jan. 2010.

[11] R. Filho, A. Traina, A. Traina, and C. Faloutsos, "Similarity search without tears: the omni-family of all-purpose access methods," in *Proc. of the 17th Intl. Conf. on Data Engineering*, 2001, pp. 623–630.

[12] C. Traina Jr., A. J. M. Traina, and C. Faloutsos, "Distance exponent: A new concept for selectivity estimation in metric trees," in *Proc. of the 16th Intl. Conf. on Data Engineering*, 2000, pp. 195–195.

[13] C. Traina Jr., A. J. M. Traina, L. Wu, and C. Faloutsos, "Fast feature selection using fractal dimension." in *Brasilian Symposium on Databases*, 2000, pp. 158–171.

[14] A. S. Arantes, M. R. Vieira, A. J. Traina, and C. Traina Jr, "The fractal dimension making similarity queries more efficient," in *Proc. of the 2nd ACM SIGKDD Workshop on Fractals, Power Laws and Other Next Generation Data Mining Tools*, 2003, pp. 12–17.

[15] E. Chávez and G. Navarro, "Metric databases," in *Encyclopedia of Database Technologies and Applications*. IGI Global, 2005, pp. 366–371.

[16] R. Mao, W. L. Miranker, and D. P. Miranker, "Dimension reduction for distance-based indexing," in *Proc. of the 3rd Intl. Conf. on Similarity Search and Applications*. New York, NY, USA: ACM, 2010, pp. 25–32.

[17] T.-S. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y.-T. Zheng, "Nus-wide: A real-world web image database from national university of singapore," in *Proc. of the ACM Intl. Conf. on Image and Video Retrieval*, Santorini, Greece, 2009.