Metric Embedding into the Hamming space with the n-Simplex Projection

Lucia Vadicamo[1], Vladimir Mic[2], Fabrizio Falchi[1], and Pavel Zezula[2]

[1] Institute of Information Science and Technologies (ISTI), CNR, Pisa, Italy
{lucia.vadicamo, fabrizio.falchi}@isti.cnr.it
[2] Masaryk University, Brno, Czech Republic
{xmic, zezula}@fi.muni.cz

Abstract. Transformations of data objects into the Hamming space are often exploited to speed-up the similarity search in metric spaces. Techniques applicable in generic metric spaces require expensive learning, e.g., selection of pivoting objects. However, when searching in common Euclidean space, the best performance is usually achieved by transformations specifically designed for this space. We propose a novel transformation technique that provides a good trade-off between the applicability and the quality of the space approximation. It uses the *n-Simplex projection* to transform metric objects into a low-dimensional Euclidean space, and then transform this space to the Hamming space. We compare our approach theoretically and experimentally with several techniques of the metric embedding into the Hamming space. We focus on the applicability, learning cost, and the quality of search space approximation.

Keywords: sketch · metric search · metric embedding · n-point property

1 Introduction

The *metric search* problem aims at finding the most similar data objects to a given query object under the assumption that there exists a metric function assessing the dissimilarity of any two objects. The broad applicability of the metric space similarity model makes the metric search a challenging task, since the distance function is the only operation that can be exploited to compare two objects. One way to speed-up the metric searching is to transform the space to use a cheaper similarity function or to reduce data object sizes [4,9,14,19]. Recently, Connor et al. proposed the *n-Simplex projection* that transforms the metric space into a finite-dimensional Euclidean space [9,8]. Here, specialised similarity search techniques can be applied. Moreover, the Euclidean distance is more efficient to evaluate than many distance functions.

Another class of metric space transformations is formed by *sketching techniques* that transform data objects into short bit-strings called *sketches* [4,17,19]. The similarity of sketches is expressed by the Hamming distance, and sketches are exploited to prune the search space during query executions [19,18]. While some sketching techniques are applicable in generic metric spaces, others are designed for specific spaces [4]. The metric-based sketching techniques are broadly applicable, but their performance is often worse than that of the vector-based sketching approaches when dealing with the vector spaces [4,17].

We propose a novel sketching technique *NSP_50* that combines advantages of both approaches: wide applicability and good space approximation. It is applicable to the large class of metric spaces meeting the *n-point property* [7,3], and it consists of the projection of the search space into a low-dimensional Euclidean space (*n-Simplex projection*) and the binarization of the vectors. The NSP_50 technique is particularly advantageous for expensive metric functions, since the learning of the projection requires a low number of distance computations. The main contribution of the NSP_50 is a better trade-off between its applicability, quality of the space approximation, and the pre-processing cost.

2 Background and Related Work

We focus on the similarity search in domains modelled by the *metric space* (D, d), with the domain of objects D and the metric (*distance*) function $d : D \times D \to \mathbb{R}^+$ [21] that expresses the dissimilarity of objects $o \in D$. We consider the data set $S \subseteq D$, and the so-called *kNN queries* that search for the k closest objects from S to a *query* object $q \in D$. Similarity queries are often evaluated in an approximate manner since the slightly imprecise results are sufficient in many real-life applications and they can be delivered significantly faster than the precise ones. Many metric space transformations have been proposed to speed-up the approximate similarity searching, including those producing the Hamming space [4,5,11,18,19], Euclidean space [9,16] and Permutation space [1,6,20]. We further restrict our attention to the metric embedding into the Hamming space.

2.1 Bit String Sketches for Speeding-up Similarity Search

Sketching techniques $sk(\cdot)$ transform the metric space (D, d) to the Hamming space $\big(\{0,1\}^\lambda, h\big)$ to approximate it with smaller objects and more efficient distance function. We denote the produced bit strings as *sketches* of length λ. Many sketching techniques were proposed – see for instance the survey [4]. Their main features are: (1) Quality, i.e., the ability to approximate the original metric space; (2) Applicability to various search spaces; (3) Robustness with respect to data (intrinsic) dimensionality; (4) Cost of the object-to-sketch transformation; (5) Cost of the transformation learning. In the following, we summarise concepts of three techniques that we later compare with the newly proposed NSP_50 technique. They all produce sketches with *balanced bits*, i.e. each bit i is set to 1 in one half of the sketches $sk(o), o \in S$. This is denoted by the suffix *_50* in their notations.

GHP_50 technique [18] uses λ pairs of reference objects (*pivots*), that define λ instances of the *Generalized Hyperplane Partitioning* (GHP) [21] of the dataset S. Therefore, each GHP instance splits the dataset into two parts according to the closer pivot, and these parts define values of one bit of all sketches $sk(o), o \in S$. The pivots are selected to produce balanced and low correlated bits [18]: (1) an initial set of pivots $P_{sup} \in D$ is selected in random, (2) the balance of the GHP is evaluated for all pivot pairs using a sample set T of S, (3) set P_{bal} is formed by pivot pairs that divide T into parts balanced to at least 45 % to 55 %, and corresponding sketches sk_{bal} are created, (4) the correlation matrix M with absolute values of the Pearson correlation coefficient is evaluated for all pairs of bits of sketches sk_{bal}, and (5) a heuristic is applied to select rows and columns of M which form its sub-matrix with low values and size $\lambda \times \lambda$. (6) Finally, the λ pivot *pairs* that produce the corresponding low correlated bits define sketches $sk(o), o \in S$.

BP_50 uses the *Ball Partitioning* (BP) instead of the GHP [18]. BP uses one pivot and a radius to split data into two parts, that again define the values in one bit of sketches $sk(o), o \in S$. Pivots are selected again via a random set of pivots P_{sup}, for which we evaluate radii dividing the sample set T into halves. The same heuristic as in case of the technique GHP_50 is than employed to select λ pivots that produces low correlated bits.

PCA_50 is a simple sketching technique surprisingly well approximating the Euclidean spaces [12,15,17,13,4]. It uses the *Principal Component Analysis* (*PCA*) to shrink the original vectors, which are then rotated using a random matrix and binarized by the thresholding. The i-th bit of sketch $sk(o)$ thus expresses whether the i-th value in the shortened vector is bigger then the median computed on a sample set T. If sketches longer than the original vectors are desired, we propose to apply the PCA and to rotate transformed vectors using independent random matrices. Then we concatenate corresponding binarized vectors.

Sketching techniques applicable to generic metric spaces, e.g., GHP_50 and BP_50, are usually of a worse quality than vector-based sketching techniques when dealing with the vectors spaces [17,4]. Moreover, they require an expensive learning of the transformation. We propose the sketching technique NSP_50 to provide a better trade-off between the quality of the space approximation, applicability of the sketching, and the pre-processing cost.

2.2 The n-Simplex projection

The *n-Simplex projection* [9] associated with a set of n pivots $\mathcal{P}_n$ is a space transformation $\phi_{\mathcal{P}_n} : (D, d) \to (\mathbb{R}^n, \ell_2)$ that maps the original metric space to a n-dimensional Euclidean space. It can be applied to any metric space with the *n-point property*, which states that any n points $o_1, ..o_n$ of the space can be *isometrically* embedded in the $(n-1)$-dimensional Euclidean space. Many often used metric spaces such as Euclidean spaces of any dimension, spaces with the Triangular or Jensen-Shannon distances, and, more generally, any Hilbert-embeddable spaces meet the n-point property [7]. The n-Simplex projection is properly described in [9]. Here, we sketch just the main concepts.

First, the n-point property guarantees that there exists an isometric embedding of the n pivots into $(\mathbb{R}^{n-1}, \ell_2)$ space, i.e., it is possible to construct the vertices $v_{p_i} \in \mathbb{R}^{n-1}$ such that $\ell_2(v_{p_i}, v_{p_j}) = d(p_i, p_j)$ for all $i, j \in \{1, \ldots, n\}$. These vertices form the so-called *base simplex*. Second, for any other object $o \in D$, the $(n+1)$-point property guarantees that there exists a vertex $v_o \in \mathbb{R}^n$ such that $\ell_2(v_o, v_{p_i}) = d(o, p_i)$ for all $i = 1, \ldots, n$. The n-Simplex projection assigns such v_o to o, and Connor et al. [9] provide an iterative algorithm to compute the coordinates of the vertices v_{p_i} of the simplex base as well as the coordinates of the vector v_o associated to $o \in D$. The base simplex is computed once and reused to project all data objects $o \in S$. Moreover, the Euclidean distance between any two projected vectors $v_{o_1}, v_{o_2} \in \mathbb{R}^n$ is a lower-bound of their actual distance, and this bound becomes tighter with increasing number of pivots n [9].

3 The n-Simplex Sketching: Proposal & Comparison

We propose the sketching technique *NSP_50* that transforms metric spaces with the n-point property to the Hamming space. It uses the n-Simplex projection with λ pivots to project objects into λ-dimensional Euclidean space; the obtained vectors are then randomly rotated and binarized using the median values in each coordinate. These medians are evaluated on the data sample set. The random rotation is applied to distribute information equally over the vectors, as the n-Simplex projection returns vectors with decreasing values along the dimensions.

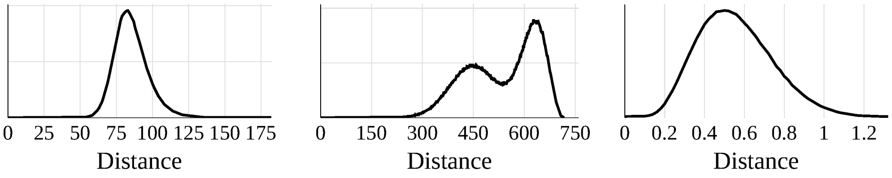
For each data set S, there exists a finite number of pivots $\tilde{n}$ such that $\phi_{\mathcal{P}_{\tilde{n}}}$ is an isometric space embedding[3]. The identification of the minimum $\tilde{n}$ with this property is still an open problem. The convergence is achieved when all the projected data points have a zero value in their last component, so the NSP_50 technique as described above cannot produce meaningful sketches of length $\lambda > \tilde{n}$. We overcome this issue by a concatenation of smaller sketches obtained using different rotation matrices.

The proposed NSP_50 technique is inspired by the PCA_50 approach, but provides significantly broader applicability, as it can transform all the metric spaces with the n-point property. This includes spaces with very expensive distance functions, as mentioned in Section 2.2. Sketching techniques also require transformation learning of a significantly different complexity. We compare the novel NSP_50 technique with the GHP_50, BP_50 and PCA_50 approaches and we provide the table summarising the main features of these sketching techniques, including the costs of the learning and object to sketch transformations in terms of floating point operations and distance computations. This table is provided online[4], due to the paper length limitation.

The GHP_50 and BP_50 techniques require an expensive pivot learning. Specifically, the GHP_50 requires (1) to examine the balance of the GHPs defined by various pivot pairs to create long sketches with the balanced bits, (2) an analysis of the pairwise bit correlations made for these sketches, and (3) a

[3] The proof is made trivially by a selection of all objects from the data set S as pivots.

[4] `http://www.nmis.isti.cnr.it/falchi/SISAP19SM.pdf`



(a) DeCAF (b) SIFT (c) SQFD

Fig. 1: Distance densities for DeCAF, SIFT and SQFD data sets

selection of low correlated bits. The learning of the BP_50 is cheaper, since the proper radii are selected for a set of pivots directly. The rest of the learning is the same as in case of the GHP_50. The cost of the PCA_50 learning is given by the PCA learning cost and evaluation of the medians over the transformed vectors. We compute the PCA matrix using the Singular Value Decomposition (SVD) over the centred data. The learning of the NSP_50 is the cheapest one; it consists of the n-Simplex projection that has the quadratic cost with respect to the number of pivots n, and the binarization, which consists of the medians evaluations over coordinates of vectors in the sample set T.

4 Experiments

We evaluate the search quality of the NSP_50 technique on three data sets and we compare it with the sketching techniques PCA_50, GHP_50 and BP_50. We use three real-life data sets of visual features extracted from images:

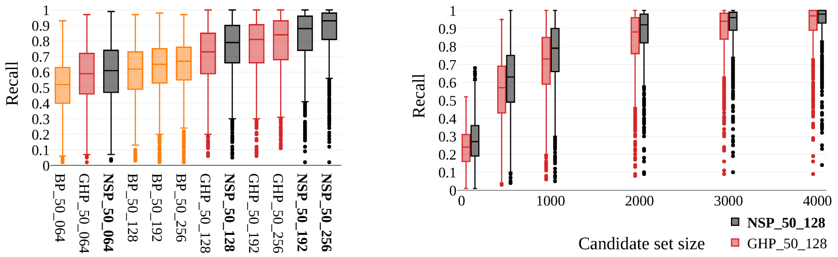
SQFD: 1 million *adaptive-binning feature histograms* [2] extracted from the *Profiset collection*[5]. Each signature consists of, on average, 60 cluster centroids in a 7-dimensional space. A weight is associate to each cluster, and the signatures are compared by the Signature Quadratic Form Distance [2]. Note that this metric is a cheaper alternative to Earth Movers Distance, nevertheless, the cost of the Signature Quadratic Form Distance evaluation is quadratic with respect to the number of cluster centroids.

DeCAF: 1 million deep features extracted from the *Profiset collection* using the Deep Convolutional Neural Network described in [10]. Each feature is a 4,096-dimensional vector of values from the last hidden layer (*fc7*) of the neural network. The deep features use the ReLU activation function and are not ℓ_2-normalised. These features are compared with the Euclidean distance.

SIFT: 1 million SIFT descriptors from the *ANN data set*[6]. Each descriptor is a 128-dimensional vector. The Euclidean distance is used for the comparison.

[5] http://disa.fi.muni.cz/profiset/

[6] http://corpus-texmex.irisa.fr/



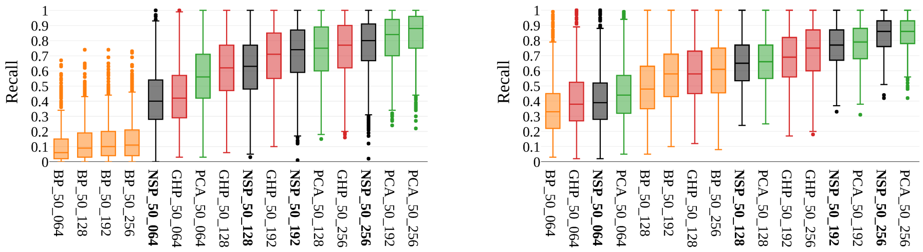
(a) Sketching techniques and lengths (b) Various candidate set sizes

Fig. 2: SQFD data set: Quality of 3 sketching techniques varying sketch lengths (2a), comparison of 128bit sketches using various candidate set sizes (2b).

Figure 1 shows particular distance densities. We express the quality of the sketching techniques by the *recall* of the k-NN queries evaluated using a simple sketch-based filtering. More specifically, sketches are applied to select the candidate set $CandSet(q)$ for each query object $q \in D$ that consists of a fixed number of the most similar sketches to the query sketch $sk(q)$; then, the candidate set is refined by the distance $d(q, o)$, $o \in CandSet(q)$ to return the k most similar objects o to q with the sketches in the candidate set $CandSet(q)$. This approximate answer is compared with the precise one that consists of the k closest objects $o \in S$ to q. The candidate sets consist of 2,000 sketches in the case of DeCAF and SIFT data sets, and 1,000 sketches in the case of the SQFD data set.

We evaluate experiments using 1,000 randomly selected query objects $q \in D$, and we depict results by *Tukey box plots* to show distributions of the recall values for particular query objects: the lower- and upper-bounds of the box show the quartiles, and the lines inside the boxes depict the medians of the recall values. The ends of the whiskers represent the minimum and the maximum non-outliers, and dots show the outlying recall values. In all cases, we examine 100 nearest neighbours queries to investigate properly the variance of the recall values over particular query objects. We use sketches of lengths $\lambda \in \{64, 128, 196, 256\}$.

Results. Figure 2a shows results for the SQFD data set. The colours of the box plots distinguish particular sketching techniques, the suffix of the column names denotes the length of sketches. The proposed NSP_50 technique significantly outperforms both, GHP_50 and BP_50 techniques, fixing the sketch length. The PCA_50 approach is not applicable for this data set, as we search different than the Euclidean space. The BP_50 technique performs worst and provides the median recall just 0.67 in case of 256bit sketches. The NSP_50 and GHP_50 approaches achieve a solid median recall of 0.88 and 0.81, respectively, even in case of 192bit sketches. We show also a coherence of the results when varying the candidate set size. Figure 2b reports the recalls for the candidate set sizes $c \in \{100, 500, 1000, 2000, 3000, 4000\}$ and sketches of length 128 bits made by the sketching techniques NSP_50 and GHP_50. This figure shows that a given



(a) DeCAF data set (b) SIFT data set

Fig. 3: Quality of sketching techniques varying sketch lengths

recall value can be achieved by the NSP_50 technique using a smaller candidate set than in case of the GHP_50.

The recall values for the DeCAF and SIFT data sets are depicted in Figure 3. The BP_50 technique is less robust concerning the dimensionality of the data, so it achieves poor recalls in case of DeCAF descriptors, but it is still reasonable for the SIFT data set. The quality of the newly proposed NSP_50 technique is slightly better then that of the GHP_50 technique in case of the DeCAF data set. Both are, however, outperformed by the PCA_50 technique, which is specialised for the Euclidean space. This interpretation is valid for all the sketch lengths λ we have tested. The differences between the NSP_50 and PCA_50 techniques practically dismiss in case of the SIFT data set. Both these techniques achieve significantly better recall than the BP_50 and the GHP_50 techniques.

5 Conclusions

We contribute to the area of the metric space embeddings into the Hamming space. We propose the NSP_50 technique that leverages the n-Simplex projection to transform metric objects into bit-string sketches. We compare the NSP_50 technique with three other state-of-the-art sketching techniques designed either for the general metric space or the Euclidean vector space. The experiments are conducted on three real life data sets of visual features using four different sketch lengths. We show that our technique provides advantages of both metric-based and specialised vector-based techniques, as it provides a good trade-off between the quality of the space approximation, applicability, and transformation learning cost.

Acknowledgements

The work was partially supported by VISECH ARCO-CNR, CUP B56J17001330004, and AI4EU project, funded by the EC (H2020 - Contract n. 825619). This research was supported by ERDF ”CyberSecurity, CyberCrime and Critical Information Infrastructures Center of Excellence” (No. CZ.02.1.01/0.0/0.0/ 16_019/0000822).

References

1. Amato, G., Gennaro, C., Savino, P.: MI-File: Using inverted files for scalable approximate similarity search. Multimed. Tools Appl. **71**(3), 1333–1362 (2014)
2. Beecks, C., Uysal, M.S., Seidl, T.: Signature quadratic form distance. In: Proceedings of the ACM-CIVR 2010. pp. 438–445. ACM (2010)
3. Blumenthal, L.M.: Theory and applications of distance geometry. Clarendon Press (1953)
4. Cao, Y., Qi, H., Zhou, W., Kato, J., Li, K., Liu, X., Gui, J.: Binary hashing for approximate nearest neighbor search on big data:A survey. IEEE Access **6**, 2039–2054 (2018)
5. Charikar, M.S.: Similarity estimation techniques from rounding algorithms. In: Proceedings of ACM-STOC 2002. ACM (2002)
6. Chávez, E., Figueroa, K., Navarro, G.: Effective proximity retrieval by ordering permutations. IEEE Trans. Pattern Anal. Mach. Intell. **30**(9), 1647–1658 (2008)
7. Connor, R., Cardillo, F.A., Vadicamo, L., Rabitti, F.: Hilbert Exclusion: Improved metric search through finite isometric embeddings. ACM Trans. Inf. Syst. **35**(3), 17:1–17:27 (Dec 2016)
8. Connor, R., Vadicamo, L., Cardillo, F.A., Rabitti, F.: Supermetric search. Information Systems (2018)
9. Connor, R., Vadicamo, L., Rabitti, F.: High-dimensional simplexes for supermetric search. In: Proceedings of SISAP 2017. pp. 96–109. Springer (2017)
10. Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., Darrell, T.: DeCAF: A deep convolutional activation feature for generic visual recognition. In: Proooceeding of ICML 2014. vol. 32, pp. 647–655 (2014)
11. Douze, M., Jégou, H., Perronnin, F.: Polysemous codes. In: ECCV - 14th European Conference, Netherlands, 2016, Proceedings, Part II. pp. 785–801 (2016)
12. Gong, Y., Lazebnik, S., Gordo, A., Perronnin, F.: Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. IEEE Trans. Pattern Anal. Mach. Intell. **35**(12), 2916–2929 (2013)
13. Gordo, A., Perronnin, F., Gong, Y., Lazebnik, S.: Asymmetric distances for binary embeddings. IEEE Trans. Pattern Anal. Mach. Intell. **36**(1), 33–47 (2014)
14. Indyk, P., Motwani, R.: Approximate nearest neighbors: Towards removing the curse of dimensionality. In: Proceedings of ACM STOC 1998. pp. 604–613 (1998)
15. Jégou, H., Douze, M., Schmid, C., Pérez, P.: Aggregating local descriptors into a compact image representation. In: Proceedings of CVPR 2010. pp. 3304–3311. IEEE (2010)
16. Kruskal, J.B.: Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis. Psychometrika **29**(1), 1–27 (Mar 1964)
17. Mic, V., Novak, D., Vadicamo, L., Zezula, P.: Selecting sketches for similarity search. In: Proceedings of ADBIS 2018. pp. 127–141 (2018)
18. Mic, V., Novak, D., Zezula, P.: Designing sketches for similarity filtering. In: Proceedings of IEEE ICDM Workshops. pp. 655–662 (Dec 2016)
19. Mic, V., Novak, D., Zezula, P.: Binary sketches for secondary filtering. ACM Trans. Inf. Syst. **37**(1), 1:1–1:28 (Dec 2018)
20. Novak, D., Zezula, P.: PPP-codes for large-scale similarity searching. In: TLDKS XXIV. pp. 61–87. Springer (2016)
21. Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity search: the metric space approach, vol. 32. Springer Science & Business Media (2006)