Similarity Search with Multiple-Object Queries

Richard Connor*1[0000−0003−4734−8103], Alan Dearle1[0000−0002−1157−2421], David Morrison1[0000−0001−5502−9773], and Edgar Chávez2[0000−0002−0148−695X]

[1] University of St Andrews, St Andrews, Scotland, UK
[2] CICESE, Ensenada, Baja California, Mexico
{rchc, al, dm236}@st-andrews.ac.uk, elchavez@cicese.edu.mx

Abstract. Within the topic of similarity search, all work we know assumes that search is based on a dissimilarity space, where a query comprises a single object in the space.
Here, we examine the possibility of a multiple-object query. There are at least three circumstances where this is useful. First, a user may be seeking results that are more specific than can be captured by a single query object. For example a query image of a yellow hot-air balloon may return other round, yellow objects, and could be specialised by a query using several hot-air balloon images. Secondly, a user may be seeking results that are more general than can be captured by a single query. For example a query image of a Siamese cat may return only other Siamese cats, and could be generalised by a query using several cats of different types. Finally, a user may be seeking objects that are in more than a single class. For example, for a user seeking images containing both hot-air balloons and cats, a query could comprise a set of images each of which contains one or other of these items, in the hope that the results will contain both.
We give an analysis of some different mathematical frameworks which capture the essence of these situations, along with some practical examples in each framework. We report some significant success, but also a number of interesting and unresolved issues. To exemplify the concepts, we restrict our treatment to image embeddings, as they are highly available and the outcomes are visually evident. However the underlying concepts transfer to general search, independent of this domain.

1 Introduction

In recent years, the field of content-based retrieval has witnessed significant advancements in the area of nearest neighbour search, enabling efficient retrieval of similar items from large collections. However, traditional nearest neighbour queries often overlook the inherent relationships between multiple complementary queries, limiting their ability to provide comprehensive results for conjunctive search scenarios. This paper introduces the concept of conjunctive similarity queries, which aims to enhance traditional nearest neighbour search by extending it to handle multiple complementary queries simultaneously. Specifically,

* Corresponding author.

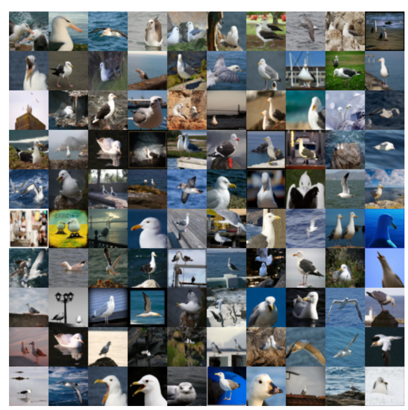
we explore the challenges and potential problems encountered in attempting to build answers for conjunctive queries. By addressing the limitations of traditional nearest neighbour search methods and offering a more comprehensive retrieval approach, conjunctive similarity queries have the potential to revolutionise content-based retrieval systems. This paper sets the foundation for future research in this exciting and promising area, aiming to bridge the gap between isolated queries and more holistic retrieval scenarios.

In this article, to exemplify the concepts, we restrict our treatment to image embeddings, as they are highly available and the outcomes are visually evident. However the underlying concepts transfer to general search, independent of this domain. In the context of images, the proposed approach is complementary to simultaneously searching a collection with both text-based and image-based queries; in this paper we use a pure image-based embedding.

To motivate this paper consider the output of the nearest neighbour search shown in Figure 1a. The query is the top leftmost image - a photograph of an albatross. In total there are 25 images in these results which are of an albatross[3]. By contrast in Figure 1b we show the output of a *conjunctive* query, where the query subject comprises a set formed by the best few results of the first query. The conjunctive query gives 82 images of albatrosses.

We stress that this is conceptually a single query, rather than the collated output of six different queries, and in fact the technique used here has approximately the same evaluation cost as the single-image query.

(a) The 100NN results of a traditional single image query of an albatross. Many of the results are images of swans.



(b) The 100NN results of a six-image conjunctive query for albatross images. The majority of the results are now albatrosses.

There are (at least) three scenarios where conjunctive query could be a useful mechanism:

[3] We describe the data set and how these categorisations were made later in the paper

query specialisation As in the above example, a user may be using a query image in order to find images of albatrosses, in which case a conjunctive query comprising images of several more specific images gives better results.

query generalisation In this scenario, the user may be seeking images of seabirds in general, in which case the initial results may be more specialised than desired. This is an increasingly prevalent scenario as embeddings improve and collections become larger. A conjunctive query comprising images of several different types of bird can give more suitable results in this case.

subject combination query In this scenario, a user has a number of query objects in different subject domains, and wishes to find objects from a collection that are somewhat similar to all of them. For example, the individual query items may be images of seabirds and boats, where the user wishes to find images which contain both of these subjects.

As far as we know, the notion of addressing these issues through a *conjunctive query* mechanism has not previously been explored. Note that in all of these cases, the result of the single conjunctive query should contain results that would not be found from performing separate queries over each element of the conjunctive query set. The intention is that the conjunctive query mechanism performs a single coherent query over some abstraction of its multiple-object argument.

The rest of this article is structured as follows. In Section 2 we outline three different possible formal models for conjunctive query, and in Section 3 we give some examples within each of the models. In Section 4 we report on the results of experiments we have performed. Finally we summarise the progress we have made, and list a number of open issues we have encountered.

2 Formal Models of Conjunctive Query

Similarity search is normally defined over a dissimilarity space (U, d), where U is some universal domain and d is a dissimilarity function. Search is performed with respect to a large finite space $S \subset U$. The general requirement is to efficiently find members of S which are most similar to an arbitrary member of U given as a query, where the function d gives the only way by which any two objects may be compared. [15,1] summarise a large volume of research in this domain.

In this paper we explore an extension of this concept where, instead of searching using a single element of U given as a query, we search using a set of elements $Q = \{q_1, ..., q_n\}$ where $q_i \in U$.

We seek general models of search in this domain; a little care is required to ensure that investigations are grounded in a useful formal framework.

Normally, for a single query q, the desired result can be defined as $\{s \in S \mid d(q, s) \leq t\}$, for some *threshold* value t which gives a useful size of result set[4]. For our context, we require to extend this. We propose that the following will suffice:

[4] Note that this definition encompasses both *range* and *nearest-neighbour* search.

- we maintain the definitions of spaces (U, d) and (S, d) as above
- instead of a query $q \in U$, search is defined in terms of a query $Q = \{q_1, ..., q_n\}$ where $q_i \in U$
- we require some more general numeric dissimilarity function $\theta : \mathcal{P}(U) \times U \to \mathbb{R}^+$ where $\mathcal{P}(U)$ is the powerset of U. θ then defines an ordering on S according to the query Q and each element $s \in S$, so that the result of a query will be $\{s \in S \mid (\theta(Q, s) \leq t\}$ for some suitable threshold t

We have outlined three general techniques for defining θ, as follows.

2.1 Aggregate measures over dissimilarity

Simplest, if we assume a similarity space (U, d), any form of aggregation may be used over d to derive θ. For example, $\theta(Q, s)$ can be very simply defined as $\sum_{q \in Q} d(q, s)$.

2.2 Generative functions

A second general model is that, from a query set Q, a single numeric function is generated, and used to order the finite data set. This requires a generative function $\sigma : \mathcal{P}(U) \to (U \to \mathbb{R})$. The result of the conjunctive query is then $\{s \in S \mid f(s) \leq t\}$, where $f = \sigma(Q)$, for some appropriate t.

One particular case of this is when f re-applies the domain distance function d. The idea is to find a single element $q' \in U$ that best represents the conjunctive query Q in the original dissimilarity domain. In this case a function σ^- maps Q to a value q', and then $f(s) = d(s, q')$, or more formally $\sigma(Q) = \lambda u.d(u, \sigma^-(Q))$.

2.3 Divergence functions

The term divergence is not, in general, formally defined. Here, we define a *divergence* function δ as a (positive, numeric) dissimilarity measure over a finite subset[5] of a domain U, i.e. $\delta : \mathcal{P}(U) \to \mathbb{R}$. The notion is that the function returns some indicator of a general dissimilarity over all the elements of its argument; for example, a collection of identical objects should return 0, while a collection of objects that have little in common with each other should return a high value.

A normal binary dissimilarity function would thus be a special case of this more general divergence: given a divergence function δ, a dissimilarity function d can be defined as $d(x, y) = \delta(\{x, y\})$.

In the context of conjunctive query, a divergence function δ can be used by ordering the search space in terms of the divergence of the query Q, with each element of S added in turn: the nearest neighbour to a query Q is thus the database object which gives the smallest divergence when that object is added to the query.

[5] As for full generality we do not wish to exclude repeated elements, we are really discussing *bags* rather than sets.

Formally, the solution to a conjunctive query Q over a finite set S is therefore

$$\{s \in S \mid \delta(Q + s) \leq t\}$$

for some appropriate value of t.

2.4 Evaluation cost

Finally in this section we note a significant difference in the potential cost of mechanisms within the different categories. For any aggregation mechanism, it will be necessary to calculate all distances between every object within Q and every object within S. For any generative mechanism which reverts to a normal distance function, not only is the cost that of a single distance against S, but any mechanism used to optimise a normal similarity search can be re-used for this purpose. Divergence functions may vary between these two extremes, and in fact our examples include one function within each cost category.

3 Conjunctive Search mechanisms

Here we outline some examples in each of the above categories.

3.1 Aggregation Functions

As mentioned, $\theta(Q, s)$ can be very simply defined as $\sum_{q \in Q} d(q, s)$. Other functions such as harmonic or geometric means are obvious contenders and have been used in agglomerative clustering [11]. A problem with such approaches is that the distances between every element of S and every element of Q require to be evaluated.

[8] gives a mechanism for extracting database objects with multiple properties, where the database has pre-calculated orderings for objects in some predefined categories. Fagin's algorithm gives an optimisation for searching the multiple lists to find the best-ranking solutions. Such an approach could be reapplied in this domain if the conjunctive queries were drawn from a fixed set of domains.

3.2 Generative Functions

One example of a generative function is the "perfect point" strategy. Suppose for example $U = (\mathbb{R}^n, \ell_2)$. For each element of the query set Q, an effective near-neighbour distance is calculated, i.e. a distance within which similar objects are found in S. From this set of distances, a point $q' \in \mathbb{R}^n$ can be calculated, with the distance from q' to each $q_i \in Q$ being equal to the near-neighbour distance of q_i. The result of a conjunctive search based on this principle is then $\{s \in S \mid \ell_2(q', s) \leq t\}$.

We note that the point q' almost undoubtedly will not exist within (S, d), especially if the elements of Q are disparate. However q' is a value which should be similar to any element of U which is equally similar to all elements of Q, and the idea is that the nearest elements of S should therefore inherit this property.

3.3 Divergence functions

We have experimented with two divergence functions, *MSED* and *nSimplex volume*. MSED can be applied to any *probability* space, that is a vector space where each vector contains only positive or zero values that sum to 1. nSimplex volume can be applied to any space that is finitely embeddable in Hilbert space, which includes Euclidean, Cosine, Jensen-Shannon and Quadratic Form spaces.

MSED The information-theoretic metric Structural Entropic Distance (SED) was first introduced in [4]. While initially proposed as a metric over labelled tree structures, the core evaluation is over probability vectors, that is any domain $\mathbb{R}^n$ where for each element v, $v_i \geq 0$ and $\sum_i v_i = 1$. This more general metric was evaluated in [13].

SED as a pairwise distance metric at its heart compares the Shannon entropy[6] of two vectors with that of their arithmetic mean. The key observation is that, if the two are equal, then so also is their mean; however the less similar they are, the higher the relative entropy of their mean. This function is normalised by the form

$$SED(v, w) = \frac{C(\frac{v+w}{2})}{\sqrt{(C(v) \cdot C(w)}} - 1$$

where $C(x) = e^{H(x)}$, to give an outcome in the range $[0, 1]$. 0 implies the two input vectors are identical, and 1 implies that no individual dimension has a non-zero value in both input vectors, i.e. their dot product is zero and they are therefore orthogonal.

In [13] we observed that this function generalises to a variadic input, rather than just a pair of values. A normalised form of this notion may derived, for a set of n probability vectors V, as:

$$MSED(V) = \frac{1}{n-1}\left(\frac{C\left(\sum_i \frac{V_i}{n}\right)}{\sqrt[n]{\Pi_i C(V_i)}} - 1\right)$$

Again, an outcome of 0 implies all elements of V are identical, and an outcome of 1 implies that all elements of V are mutually orthogonal. It is noteworthy that, for calculations of the term $MSED(Q + s)$ where Q is fixed for many different values of s, the majority of the cost can be amortised given prior knowledge of Q.

nSimplex volume In [6,5] it is shown that, in many metric spaces, a finite set of objects can be used to form an *nSimplex*. An nSimplex is a specific mapping from a set of n objects to a simplex[7] in $(n-1)$-dimensional Euclidean space.

[6] defined by $H(v) = -\sum_i v_i \ln v_i$. We take $0 \ln 0$ to be 0, see [2].

[7] A simplex is an object constructed from a set of points in n-dimensional space, by considering each point as a vertex which is joined to all of the other points. For example, a tetrahedron is a simplex formed from four points in 3D space.

The observation underlying the divergence mechanism is that, for a set of similar objects, the volume of this simplex will be relatively small.

The specific construction of the nSimplex is iterative over the objects used as its basis, at each stage forming a simplex whose last Euclidean point forms an apex of the previous simplex, in one further dimension. That is, an nSimplex in n dimensions can be formed from an nSimplex σ in $(n-1)$ dimensions and an object u, by creating a new point in n dimensions according to the distances measured between u and each object already represented in σ. Thus, the representation of u in the new simplex is the only point with a non-zero coordinate in the nth dimension, with this final coordinate representing the altitude of that point over the base simplex σ.

In the context of conjunctive search, the divergence function is the volume of this simplex. A base simplex is formed from the elements of Q, and each $s \in S$ is used to construct a new apex. The volume of each resulting simplex is therefore directly proportional to the altitude of this apex, that is its final coordinate, making the ordering of the volumes very simple to extract.

Note that although the cost of constructing the base simplex is incurred only once per conjunctive query, for each object of the database a distance must be calculated to each element of Q, making this mechanism potentially expensive as with aggregate functions.

4 Experiments

In this section we describe some of the experiments we have conducted to explore the concept of conjunctive queries[8]. We first describe the experimental setup in general and some of the infrastructure used to qualitatively assess the efficacy of the mechanisms.

The data used for all the experiments consists of one million images from the MIRFLICKR-1M image collection [10,12]. We have encoded these images using two different convolutional neural networks: one to provide a set of feature vectors to be be used in the search process, and another to provide categorical data to be used as ground truth on image similarity.

The feature vectors used in the experiments are encoded using the Dino2 ViT-S/14 network[14]. Dino2 is a state-of-the-art self-supervised pre-training method for computer vision tasks. Cosine distance is the normal metric to use over these embeddings, and we used this in most cases. MSED however is defined over probability vectors. To obtain these we applied the RELU transform to the raw data, followed by ℓ_1 normalisation. We measured the use of SED over this transformed data for normal search to be almost, but not quite, as good as Cosine distance over the raw data.

We used the Resnet18 model [9] to extract categorical labels for the images. Resnet18 is trained over ImageNet [7] data and categories.

[8] All the code for these experiments can be found on github: https://github.com/MetricSearch/sisap2023.git

For testing all of the conjunctive mechanisms outlined in Section 3, we have used the Dino2 embeddings. Many of our observations are based on the relatively subjective judgements we are able to make by repeating many different queries and looking at the outcomes. For generalisation and combination queries, we do not know of a better judgement mechanism. We re-emphasise that the data and code are both publicly available for interested researchers to do their own experiments.

4.1 Measuring specialisation queries

For specialisation queries, we have developed an objective test methodology based on the Resnet18 categorisation of the data. Given this, a significant reason for using Dino2 embeddings is that the network is trained independently of the ImageNet categorisation. That is, we only use Dino2 to perform any search task, and we use Resnet18 only to measure the quality of the search.

We take as an assumption that a high-quality classification implies a strong semantic similarity in cases where the categorisation is based on a particularly high *softmax* score. That is, two data items placed in the same category, both with high softmax values according to the classifier, are very likely to be visually similar. Furthermore, we assume that two objects, one with a high softmax score and the other with a low score, are very unlikely to be visually similar. Both of these assumptions are relatively straightforward to test.

The MIRFLICKR collection is large enough that many images are categorised correctly, even although the image set was extracted pseudo-randomly from uncategorised Flickr photographs, independent of subject matter. Over 55,000 images in the collection have a *softmax* value of greater than 0.9. In the experiments we use queries drawn from a subset of the data for which the relevant category contains between 100 and 184 such images; there are 100 such categories. For each individual test, we judge the success of a query by the number of images returned within the 100 nearest neighbours which have the same categorisation as the query, according to the softmax values.

For example, referring back to figures 1a and 1b, there are 147 images in the data set for which "albatross" is the highest category produced by the softmax layer. In the results of the single query 13 of them are categorised the same as the query. For the conjunctive query, 82 of the top 100 results are categorised as "albatross".

In each case where we have used this strategy we have also visually checked the outcomes; in fact the visual inspection in almost all cases looks rather better than is implied by the categorical scoring technique; however the latter is entirely objective.

4.2 Specialisation query experiments

In this section we compare conventional metric search, using query by example, with the conjunctive query approach with multiple query images, using the methodology described in Section 4.1.

Using the Dino2 embeddings we performed normal nearest-neighbour queries using Cosine distance to act as a baseline for the evaluation of the different conjunctive techniques.

We also ran four different variants of conjunctive queries: nSimplex, MSED, a simple average, and the perfect point strategy, all as documented in Section 3.

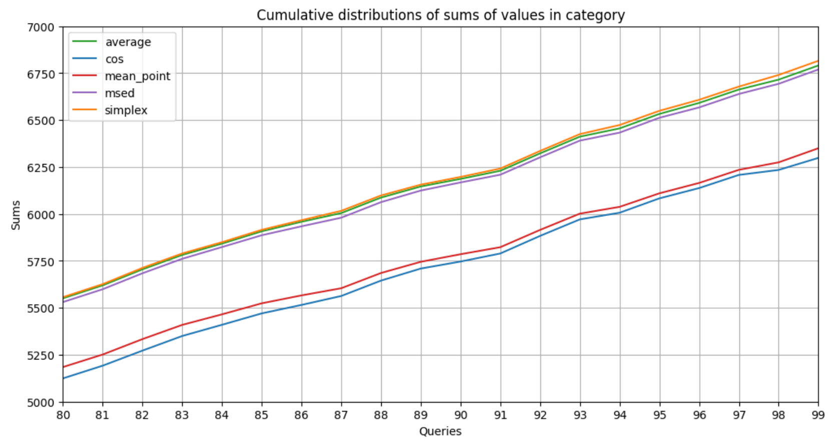


Fig. 2: Cumulative sums of Specialism Queries

Figure 2 shows the cumulative sums from the 100 experiments conducted as described above. To make the results clearer to read we show the cumulative results for the last 20 queries. The plots are of the cumulative sums for the nearest 100 results which are in the same category as the query. For reference the maximum possible cumulative total from the database is 9744.

From the figure it can be seen that three of the conjunctive queries (aggregate, nSimplex and MSED) are significantly better than the traditional single-object queries. The perfect point strategy improves a little on the single-object query. nSimplex performs the best overall, closely matched by aggregate average and MSED; we are not confident that these differences are statistically significant.

It is also worth noting that the computational cost of both nSimplex and aggregate is much higher than perfect point or MSED, both of which cost approximately the same as the single-object query. For nSimplex and aggregate, it is necessary to measure the distance between each point in the conjunctive query to all points in the database.

4.3 Generalisation query experiments

To demonstrate how generalisation may be used with conjunctive queries consider Figure 3a. The first figure shows the results from a traditional nearest neighbour for query using the image in the top left hand corner of the grid of

(a) The results of a traditional single image NN query for Siamese cat images

(b) The results of a six image NN conjunctive query for mixed cats

Fig. 3: Results from Siamese cats and general cat searches

a Siamese cat (category 284). All the results are of Siamese cats. This is an excellent result if the user were indeed looking for Siamese cats. However, they may have been seeking cats in general. The second image in Figure 3b shows the result of a conjunctive query using images drawn from the following ImageNet categories: tabby cat (281), tiger (282), Persian cat (283), Siamese cat (284), Egyptian cat (285) and leopard (288).

The results show cats drawn from all of the categories in the query, and indeed other types of cats which are not in any of the categories. They are subjectively very obviously different from the single-category results shown in Figure 3a.

We do not report quantitatively on the efficacy of the searches from this and the following section as we have not as yet performed any objective measurement. However similar results are obtained for other similar search tasks, for example by grouping different categories of dogs, fish etc. We have noted that MSED appears to function rather better than any of the other techniques we have applied to this task, but have no measurement to justify this observation.

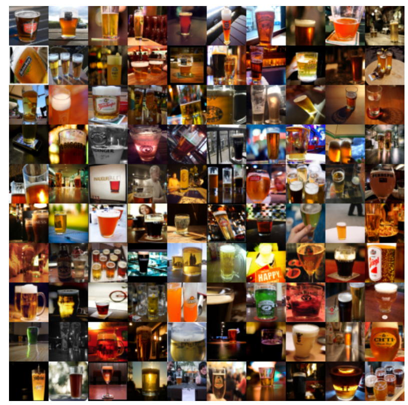
4.4 Combination query experiments

Combination queries are perhaps the most obvious use for conjunctive queries in general, and indeed this is where we started work within this domain.

After many attempts, we had had no success with any combination query mechanism. We were very hopeful that both nSimplex volume and perfect point queries would give us results in this category, but neither did. However at that point we tried MSED and the results were, in relative terms, almost unbelievably better. There are still many issues to understand, but at this point we believe MSED is the only mechanism that works well for this task, at least within the domain of image embeddings.

We give a single example of this to illustrate how the technique may be employed. The Resnet18 categories 440 and 441 are “beer bottle” and “beer glass”. The 100 images with the highest categorical scores in each of these categories are shown in Figures 4a and 4b below. Note that Figure 4a contains 3 cups or glasses and Figure 4b contains 5 bottles. Figure 5 shows the top 100 nearest neighbours from a conjunctive query formed from the best three images from each category. These images show: 19 images containing just glasses, 41 images containing just bottles (or cans) and 30 containing both categories.

(a) The images with the highest scores in category 440 (beer bottles)



(b) The images with the highest scores in category 441(beer glasses)

Fig. 4: Results from beer-bottles and beer-glasses searches

5 Conclusions

Our proposal of querying with multiple objects enables a more comprehensive and refined exploration of an object space than can be achieved by traditional dissimilarity queries. By allowing a set of objects as a query, a system can capture the collective characteristics and features present in the data. This allows for a more holistic representation of the desired concept or theme. As a result, the ranked set of objects in the solution provides a more focused and targeted selection that aligns with the specific attributes present in the query set. At this early stage we have concentrated on results in the single domain of search over image embeddings. Although our techniques are general and not image-specific, it is of course possible that they do not transfer to other semantic domains.

We have identified three specific classes of user task for which conjunctive querying provides solutions: specialisation, where a single-object search gives overly general results; generalisation, where a single object search gives overly specific results; and combination, where results which satisfy more than a single

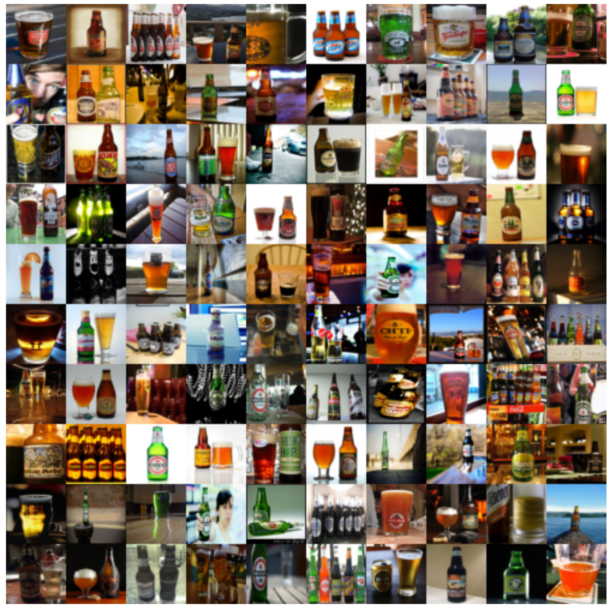


Fig. 5: Results from the conjunctive query for bottles and glasses

search topic are desired. We have shown successful examples in each of these, where a relevant user task has been satisfied.

We have identified a formal framework which serves to allow investigation into the domain, as well as identifying three different classes of functions. The classification highlights some issues of the cost of conjunctive query; for example aggregate functions require up to nk distance evaluations for n data and k query objects, while generative and some divergence functions require a maximum of only n calculations.

Objective measurement of outcomes is a significant problem. We have outlined a mechanism for measuring query specialisation, but generalisation and combination are more difficult, which makes it impossible to give any very strong and general conclusions as to the different mechanisms tested.

Our early results can be summarised as follows. For query specialisation, the outcomes are in two groups: the aggregate sum, nSimplex volume, and MSED all give notably better results than the baseline similarity query and the "perfect point" strategy. For generalisation and combination, our perception is that MSED gives significantly better results than any of the other mechanisms. This perception is very clear, however we do not know a way of measuring it convincingly. We invite interested researchers to access our code base.

In terms of query efficiency, MSED and "perfect point" both give a relatively tolerable $\mathcal{O}(n)$ calculation cost, compared with the the $\mathcal{O}(kn)$ incurred by all of the other mechanisms, for n data and a conjunctive query with k objects. Furthermore some of these strategies can be significantly optimised; for example generative functions result in a standard distance-based search, and the inverted index incremental evaluation strategy shown in [3] can be applied to MSED.

6 Further Work

This article represents a first effort to implement and evaluate a number of different mechanisms towards the novel concept of conjunctive search. We have made significant progress, but there are very many open issues.

Confirming initial results In particular for subject combination queries, our results are patchy: for some combinations of queries they work very well, for others not at all. There are many different possible explanations for this, some technical and some based simply on unknown limitations of the data sets we have used so far.

Other semantic domains In particular, we have so far experimented only with images, and in fact a specific set of images. We have used a number of different embeddings and found similar results, so we predict these results will carry forward to other image sets and queries, and also to other spaces represented by embeddings. However this requires to be checked. We can easily do so with word embeddings, high quality sets of which are freely available. It would be very interesting to try language model embeddings, where the three query types have clear and useful parallels.

Objective measurement of outcomes For specialisation queries on images, we have found a reasonable working model of an objective test of different techniques. As yet we are less clear about how to test generalisation or combination queries. Without such objective tests it is difficult to be very confident in comparing mechanisms with each other.

Efficient conjunctive search Almost all of the effort expended in the similarity search domain has been on how to evaluate queries efficiently against very large data sets. Many of these techniques transfer directly to this domain, in particular for techniques which result in a normal similarity search against the database. However it may be that sets of restrictive postulates exist in this domain, as alternatives to e.g. the metric postulates, which allow scaleable or other more efficient search to avoid exhaustive calculation against the database, in which case the more expensive metrics such as nSimplex volume may become relatively more usable.

We are currently exploring avenues for exploiting these results. One of these is in data-linkage in which it is advantageous to be able to search for groups of related records; for example the birth records of siblings in a single family. Recommender systems could allow suggestions for future purchases based on a conjunctive query comprising items already purchased within a category, which for example may capture a certain fashion sense. In the domain of drug discovery, a researcher might be interested in finding a peptide that possesses antibacterial and anticancer properties: a conjunctive similarity query might aim to retrieve such peptides. Lastly in the field of histology, it may be possible to generalise over a number of different pathological cell images in order to detect others of a similar type with a different visual presentation.

Acknowledgements

This work is partly supported by ESRC grant ES/W010321/1 "2022-2026 ADR UK Programme".

References

1. Chávez, E., Navarro, G., Baeza-Yates, R., Marroquín, J.L.: Searching in metric spaces **33**(3), 273–321. https://doi.org/10.1145/502807.502808, https://dl.acm.org/doi/10.1145/502807.502808
2. Connor, R.: A tale of four metrics. In: 9th International Conference on Similarity Search and Applications, SISAP 2016. pp. 210–217. Springer (2016)
3. Connor, R., Cardillo, F.A., Moss, R., Rabitti, F.: Evaluation of jensen-shannon distance over sparse data. In: Similarity Search and Applications: 6th International Conference, SISAP 2013, A Coruña, Spain, October 2-4, 2013, Proceedings 6. pp. 163–168. Springer (2013)
4. Connor, R., Simeoni, F., Iakovos, M., Moss, R.: A bounded distance metric for comparing tree structure. Information Systems **36**(4), 748–764 (2011)
5. Connor, R., Vadicamo, L.: nsimplex zen: A novel dimensionality reduction for euclidean and hilbert spaces. arXiv preprint arXiv:2302.11508 (2023)
6. Connor, R., Vadicamo, L., Rabitti, F.: High-dimensional simplexes for supermetric search. In: Similarity Search and Applications: 10th International Conference, SISAP 2017, Munich, Germany, October 4-6, 2017, Proceedings 10. pp. 96–109. Springer (2017)
7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
8. Fagin, R., Lotem, A., Naor, M.: Optimal aggregation algorithms for middleware. In: PODS. p. 102–113. PODS '01, Association for Computing Machinery, New York, NY, USA (2001). https://doi.org/10.1145/375551.375567
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. CoRR **abs/1512.03385** (2015), http://arxiv.org/abs/1512.03385
10. Huiskes, M.J., Lew, M.S.: The mir flickr retrieval evaluation. In: MIR '08: Proceedings of the 2008 ACM International Conference on Multimedia Information Retrieval. ACM, New York, NY, USA (2008)
11. Jain, A.K., Murty, M.N., Flynn, P.J.: Data clustering: A review **31**(3), 264–323 (sep 1999). https://doi.org/10.1145/331499.331504, https://doi.org/10.1145/331499.331504
12. Mark J. Huiskes, B.T., Lew, M.S.: https://press.liacs.nl/mirflickr/
13. Moss, R., Connor, R.: A multi-way divergence metric for vector spaces. In: Proceedings of the 6th International Conference on Similarity Search and Applications - Volume 8199. p. 169–174. SISAP 2013, Springer-Verlag, Berlin, Heidelberg (2013)
14. Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., et al.: Dinov2: Learning robust visual features without supervision (2023), https://arxiv.org/abs/2304.07193
15. Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity search: the metric space approach, vol. 32. Springer Science & Business Media (2006)