Hot or Not: Interactive Content Search Using Comparisons

Amin Karbasi
School of Computer and Communication Sciences
Ecole Polytechnique Fédérale de Lausanne (EPFL)
Lausanne, Switzerland

Stratis Ioannidis
Technicolor
Palo Alto
USA

Laurent Massoulié
Technicolor
Paris
France

***Abstract*—In interactive content search through comparisons, a user searching for a target object in a database is asked to select the object most similar to her target from a small list of objects. A new object list is then presented to the user based on her earlier selections. This process is repeated until the target is included in the list presented, at which point the search terminates.**

We study this problem under the scenario of *heterogeneous demand*, where target objects are selected from a non-uniform probability distribution. We also assume that objects are embedded in a *doubling metric space* which is fully observable to the search algorithm. Based on these assumptions, we devise an efficient comparison-based search algorithm whose cost in terms of the number of queries can be bounded by the doubling constant of the embedding c, and the entropy of demand distribution, H. More precisely, we show that the average search costs scales $\bar{C}_{\mathcal{F}} = O(c^5 H)$, which improves upon the previously best known bound and is order optimal for constant c.

I. Introduction

In interactive content search through comparisons, a user navigates within a database to find a target object in the following fashion. The search iterates over multiple phases and, in each phase, a small list of objects is presented to the user. The user selects among the list the object closest to the target; a new object list is then presented to the user based on her earlier selections. This process continues until the target is included in the list presented, at which point she retrieves this item and the search terminates.

This kind of interactive navigation, also known as exploratory search, has numerous real-life applications [1], [2], [3]. One example is navigating through a database of pictures of people photographed in an uncontrolled environment [4], such as Fickr or Picasa. Automated methods may fail to extract meaningful features from such photos. Moreover, in many practical cases, images that present similar low-level descriptors (such as SIFT features [5]) may have very different semantic content and high level descriptions, and thus be perceived differently by users [6], [7].

On the other hand, a human searching for a particular person can easily select from a list of pictures the subject most similar to the person she has in mind. Formally, the behavior of a human user can be modelled by a so-called *comparison oracle*, introduced by Goyal *et al.* [8]. In particular, assume that that the database of pictures is represented by a set $\mathcal{N}$ endowed with a distance metric d. This metric captures the "distance" or "dissimilarity" between pictures of different people. The oracle/human has a specific target $t \in \mathcal{N}$ in mind, and can answer questions of the following kind:

> "Between two objects x and y in N, which one is closest to t under the metric d?"

The goal of interactive content search through comparisons is thus to find a sequence of proposed pairs of objects to the oracle/human that leads the target object with as few queries as possible.

In this paper, we consider the problem under the scenario of *heterogeneous demand*, where the target object $t \in \mathcal{N}$ is sampled from a probability distribution μ. In this setting, interactive content search through comparisons has a strong relationship to the classic "twenty-questions game" problem. In particular, a *membership oracle* [9] is an oracle that can answer queries of the following form:

> "Given a subset $A \subseteq \mathcal{N}$, does t belong to A?"

It is well known that to find a target t one needs to submit at least $H(\mu)$ queries, on average, to a membership oracle, where $H(\mu)$ is the entropy of μ. Moreover, there exists an algorithm (Huffman coding) that finds the target with only $H(\mu)+1$ queries on average [9].

Content search through comparisons departs from the above setup in assuming that the database $\mathcal{N}$ is endowed with the metric d. A membership oracle is stronger than a comparison oracle as, if the distance metric d is known, comparison queries can be simulated through membership queries. On the other hand, a membership oracle is harder to implement in practice: unless A can be expressed in a concise fashion, a user will answer a membership query in linear time in $|A|$. This is in contrast to a comparison oracle, for which answers can be given in constant time. In short, our study of search through comparisons seeks similar performance bounds to the classic setup (a) for an oracle that is easier to implement and (b) under an additional assumption on the structure of the database (namely, that it is endowed with a distance metric).

Intuitively, the performance of searching for an object through comparisons will depend not only on the entropy of the target distribution, but also on the topology of the target set $\mathcal{N}$, as described by the metric d. In particular, in earlier work [10], we established that $\Omega(cH(\mu))$ queries are necessary, in expectation, to locate a target using a comparison oracle, where c is the so-called *doubling-constant* of the metric d. Moreover, we provided an algorithm that locates the target in $O(c^3 H \log(1/\mu^*))$ queries, in expectation, where $\mu^* = \min_{x \in \mathcal{N}} \mu(x)$. In this paper, we improve on the previous bound by proposing an algorithm that locates the target with $O(c^5 H(\mu))$ queries, in expectation.

The remainder of this paper is organized as follows. In Section II we provide an overview of the related work in this area. In Sections III and IV we introduce our notation and formally state the problem that is the focus of this work. We present our previous contributions in Section V and our main result in Section VI. We finally conclude in Section VII.

II. Related Work

Content search through comparisons is a special case of nearest neighbour search (NNS), a problem that has been extensively studied [11], [12]. Our work can be seen as an extension of earlier work [13], [14], [11] considering the NNS problem for objects embedded in a metric space. It is also assumed that the embedding has a small intrinsic dimension, an assumption that is supported by many practical studies [15], [16]. In particular, [14] introduces navigating nets, a deterministic data structure for supporting NNS in doubling metric spaces. A similar technique was considered by [11] for objects embedded in a space satisfying a certain sphere-packing property, while [13] relied on growth restricted metrics; all of the above assumptions have connections to the doubling constant we consider in this paper. In all of the above works the demand over the target objects is assumed to be homogeneous.

NNS with access to a comparison oracle was first introduced by [8], and further explored by [17] and [4]. A considerable advantage of the above works is that the assumption that objects are a-priori embedded in a metric space is removed; rather than requiring that similarity between objects is captured by a distance metric, the above works only assume that any two objects can be ranked in terms of their similarity to any target by the comparison oracle. Nevertheless, these works also assume homogeneous demand, so our work can be seen as an extension of searching with comparisons to heterogeneity. In this respect, the closet works to ours are [10], [18], where authors also assume heterogeneous demand distribution. Under the assumptions that a metric space exists and the search algorithm is aware of it, we provide better results in terms of the average search cost. The main problem with [10] is that their approach is memoryless, i.e., it does not make use of previous comparisons, whereas in our work we solve this problem by deploying an ϵ-net data structure.

The first practical scheme for image retrieval, based on the pairwise comparisons between images, was proposed by [19]. It was then extended by [20] and [21] to the context of content search. The use of comparison oracle is not limited only to content retrieval/search. As explained in [22], [23], the individuals' rating scale tends to fluctuate a lot. In addition, ratings scales may vary between people. For these reasons it is more natural to use the pairwise comparisons as the basis for the recommendation systems. The advantages of this approach and the challenges of how to make such a system operational is well described in [24].

III. Definitions and Notation

Consider a set of objects $\mathcal{N}$, where $|\mathcal{N}| = n$. We assume that there exists a metric space $(\mathcal{M}, d)$, where $d(x, y)$ denotes the distance between $x, y \in \mathcal{M}$, such that objects in $\mathcal{N}$ are embedded in $(\mathcal{M}, d)$: *i.e.*, there exists a one-to-one mapping from $\mathcal{N}$ to a subset of $\mathcal{M}$.

The objects in $\mathcal{N}$ may represent, for example, pictures in a database. The metric embedding can be thought of as a mapping of the database entries to a set of features (*e.g.*, the age of person depicted, her hair and eye color, *etc.*). The distance between two objects would then capture how "similar" two objects are w.r.t. these features. In what follows, we will abuse notation and write $\mathcal{N} \subseteq \mathcal{M}$, keeping in mind that there might be difference between the physical objects (the pictures) and their embedding (the attributes that characterize them).

A. Comparison Oracle

A *comparison oracle* [8] is an oracle that, given two objects x, y and a target t, returns the closest object to t. More formally,

$$\text{Oracle}(x, y, t) = \begin{cases} x & \text{if } d(x,t) < d(y,t), \\ y & \text{if } d(x,t) > d(y,t) \\ x \text{ or } y & \text{if } d(x,t) = d(y,t). \end{cases} \tag{1}$$

Observe that if $x = \text{Oracle}(x, y, t)$ then $d(x, t) \leq d(y, t)$; this does not necessarily imply however that $d(x, t) < d(y, t)$.

It is important to note here that although we write $\text{Oracle}(x, y, t)$ to stress that a query always takes place with respect to some target t, in practice the target is hidden and only known by the oracle. Alternatively, following the "oracle as human" analogy, the human user has a target in mind and uses it to compare the two objects, but never discloses it until actually being presented with it.

B. Demand, Entropy and Doubling Constant

We will consider a probability distribution μ over the set of objects in $\mathcal{N}$ which we will call the *demand*. In other words, μ will be a non-negative function such that $\sum_{t \in \mathcal{N}} \mu(t) = 1$. In general, the demand can be *heterogeneous* as $\mu(t)$ may vary across different targets. As we will see in Section V, the target distribution μ will play an important role in our analysis. In particular, two quantities that affect the performance of searching in our scheme will be the *entropy* and the *doubling constant* of the target distribution. We introduce these two notions formally below.

The *entropy* of μ is defined as

$$H(\mu) = \sum_{x \in \mathsf{supp}(\mu)} \mu(x) \log \frac{1}{\mu(x)}, \tag{2}$$

where $\mathsf{supp}(\mu)$ is the support of μ. We define the *max-entropy* of μ as

$$H_{\max}(\mu) = \max_{x \in \mathsf{supp}(\mu)} \log \frac{1}{\mu(x)}. \tag{3}$$

TABLE I
SUMMARY OF NOTATION

$\mathcal{N}$	Set of objects
$(\mathcal{M}, d)$	Metric space
$d(x, y)$	Distance between $x, y \in \mathcal{M}$
μ	The demand distribution
$H(\mu)$	The entropy of μ
$H_{\max}(\mu)$	The max-entropy of μ
$B_x(r)$	The ball of radius r centered at x
$c(\mu)$	The doubling constant of μ

Given an object $x \in \mathcal{N}$, we denote by

$$B_x(R) = \{y \in \mathcal{M} : d(x, y) \leq R\} \tag{4}$$

the closed ball of radius $R \geq 0$ around x. Given a set $A \subset \mathcal{N}$ let

$$\mu(A) = \sum_{x \in A} \mu(x).$$

We define the *doubling constant* $c(\mu)$ of a distribution μ to be the minimum $c > 0$ for which

$$\mu(B_x(2R)) \leq c \cdot \mu(B_x(R)), \tag{5}$$

for any $x \in \mathsf{supp}(\mu)$ and any $R \geq 0$. Moreover, will say that μ is *c-doubling* if $c(\mu) = c$.

Note that, contrary to the entropy $H(\mu)$, the doubling constant $c(\mu)$ depends on the topology of $\mathsf{supp}(\mu)$, determined by the embedding of $\mathcal{N}$ in the metric space $(\mathcal{M}, d)$.

IV. PROBLEM STATEMENT

In formulating our problem, we follow the notation in [10]. Given access to a comparison oracle, we would like to navigate through $\mathcal{N}$ until we find a target object. In particular, we define *greedy content search* as follows. Let t be the target object and s some object that serves as a starting point. The greedy content search algorithm proposes an object w and asks the oracle to select, between s and w, the object closest to the target t, *i.e.*, it evokes Oracle(s, w, t). This process is repeated until the oracle returns something other than s, *i.e.*, the proposed object is "more similar" to the target t. Once this happens, say at the proposal of some w', if $w' \neq t$, the greedy content search repeats the same process now from w'. If at any point the proposed object is t, the process terminates.

More formally, let x_k, y_k be the k-th pair of objects submitted to the oracle: x_k is the *current object*, which greedy content search is trying to improve upon, and y_k is the *proposed object*, submitted to the oracle for comparison with x_k. Let

$$o_k = \text{Oracle}(x_k, y_k, t) \in \{x_k, y_k\}$$

be the oracle's response, and define

$$\mathcal{H}_k = \{(x_i, y_i, o_i)\}_{i=1}^{k}, \quad k = 1, 2, \ldots$$

be the sequence of the first k inputs given to the oracle, as well as the responses obtained; $\mathcal{H}_k$ is the "history" of the content search up to and including the k-th access to the oracle.

The starting object is always one of the first two objects submitted to the oracle, *i.e.*, $x_1 = s$. Moreover, in greedy content search,

$$x_{k+1} = o_k, \quad k = 1, 2, \ldots$$

i.e., the current object is always the closest to the target among the ones submitted so far.

On the other hand, the selection of the proposed object y_{k+1} will be determined by the history $\mathcal{H}_k$ and the object x_k. In particular, given $\mathcal{H}_k$ and the current object x_k there exists a mapping $(\mathcal{H}_k, x_k) \mapsto \mathcal{F}(\mathcal{H}_k, x_k) \in \mathcal{N}$ such that

$$y_{k+1} = \mathcal{F}(\mathcal{H}_k, x_k), \quad k = 0, 1, \ldots,$$

where here we take $x_0 = s \in \mathcal{N}$ (the starting object) and $\mathcal{H}_0 = \emptyset$ (*i.e.*, before any comparison takes place, there is no history).

We will call the mapping $\mathcal{F}$ the *selection policy* of the greedy content search. In general, we will allow the selection policy to be randomized; in this case, the object returned by $\mathcal{F}(\mathcal{H}_k, x_k)$ will be a random variable, whose distribution

$$\Pr(\mathcal{F}(\mathcal{H}_k, x_k) = w), \quad w \in \mathcal{N}, \tag{6}$$

is fully determined by $(\mathcal{H}_k, x_k)$. Observe that $\mathcal{F}$ depends on the target t only indirectly, through $\mathcal{H}_k$ and x_k; this is consistent with our assumption that t is only "revealed" when it is eventually located.

We will say that a selection policy is *memoryless* if it depends on x_k but not on the history $\mathcal{H}_k$. In other words, the distribution (6) is the same when $x_k = x \in \mathcal{N}$, irrespectively of the comparisons performed prior to reaching x_k.

Assuming that when $x_k = t$, the search effectively terminates (*i.e.*, the human reveals that this is indeed the target), our goal is to select $\mathcal{F}$ so that we minimize the number of accesses to the oracle. In particular, given a a target t and a selection policy $\mathcal{F}$, we define the search cost

$$C_{\mathcal{F}}(t) = \inf\{k : x_k = t\}$$

to be the number of proposals to the oracle until t is found. This is a random variable, as $\mathcal{F}$ is randomized; let $\mathbb{E}[C_{\mathcal{F}}(t)]$ be its expectation. The Content Search Through Comparisons problem is then defined as follows:

> CONTENT SEARCH THROUGH COMPARISONS (CSTC): Given an embedding of $\mathcal{N}$ into $(\mathcal{M}, d)$ and a demand distribution $\mu(t)$, select $\mathcal{F}$ that minimizes the expected search cost
>
> $$\bar{C}_{\mathcal{F}} = \sum_{t \in \mathcal{N}} \mu(t) \mathbb{E}[C_{\mathcal{F}}(t)].$$

Note that, as $\mathcal{F}$ is randomized, the free variable in the above optimization problem is the distribution (6).

Algorithm 1 Memoryless Content Search

```
Input: oracle($\cdot$,$\cdot$,$t$) , demand distribution $\mu$, starting object $s$.
Output: target $t$.
x ← s
while $x \neq t$ do
    Sample $y \in \mathcal{N}$ from the probability distribution
        $\Pr_x(y) \propto \frac{\mu(y)}{\mu(B_x(d(x,y)))}$.    (8)
    $x \leftarrow$ Oracle($x, y, t$).
end while
```

V. A Lower Bound and a Memoryless Algorithm

We first present our previous results established in [18], [10]. Our first result, whose proof is in [10], establishes a lower bound on the expected number of queries that one needs to submit to a comparison oracle to locate a target t.

Theorem 1. *For any integer K and D, there exists a metric space $(\mathcal{M}, d)$ and a target measure μ with entropy $H(\mu) = K\log(D)$ and doubling constant $c(\mu) = D$ such that the average search cost of any selection policy $\mathcal{F}$ satisfies*

$$\bar{C}_{\mathcal{F}} \geq H(\mu)\frac{c(\mu)-1}{2\log(c(\mu))}. \tag{7}$$

Interestingly, a simple *memoryless* selection policy, shown in Algorithm 1, satisfies an upper bound that is within an $O(c^2(\mu)H_{\max}(\mu))$ factor of this bound.

Theorem 2. *The expected search cost of Algorithm 1 is bounded by $\bar{C}_{\mathcal{F}} \leq 6c^3(\mu) \cdot H(\mu) \cdot H_{\max}(\mu)$.*

The proof of this theorem can also be found in [10]. There are several interesting observations to be made about Algorithm 1. To begin with, the memoryless selection policy (8) has the following appealing properties. For two objects y, z that have the same distance from x, if $\mu(y) > \mu(z)$ then y has a higher probability of being proposed. When two objects y, z are equally likely to be targets, if $d(y, x) < d(z, x)$ then y has a higher chance of being proposed. The distribution (8) thus biases both towards objects close to x as well as towards objects that are likely to be targets.

Moreover, in implementing the policy outlined in Algorithm 1, it is assumed that, at each x, a random y can be sampled from distribution (8). This assumes that the distribution μ and the embedding $\mathcal{M}$ (or the distance metric d) are a-priori known. However, it is in fact true that Algoritm 1 can be implemented even if only *the ordering relationships between objects*, rather than their actual distances between targets, are known [10]. This is important, as the latter can be obtained *by only accessing a comparison oracle*. In particular, all such ordering relationships can be revealed by asking $|\mathcal{N}|\log|\mathcal{N}|$ oracle queries offline (*e.g.*, during a training phase).

As noted, the main discrepancy factor between the upper bound in Theorem 2 and the lower bound in Theorem 1 is of the order of $c^3 H_{\max}$. Our next result, appearing in the next Section eliminates the $H_{\max}$ term at the expense of a dependence on the doubling dimension through an $O(c^5)$ term.

VI. An Algorithm Based on ϵ-nets

Our objective in this section is to establish that comparison-based search can complete in identifying an object target $t \in \mathcal{N}$ initially sampled according to probability distribution μ in a number of steps $C_{\mathcal{F}}$ whose average value $\bar{C}_{\mathcal{F}}$ verifies

$$\bar{C}_{\mathcal{F}} \leq H(\mu)c^k(\mu)$$

for some fixed exponent k to be identified. To this end, we establish a number of intermediate results.

A. ϵ-Nets

We define ϵ-Nets as follows:

Definition 1. *An ϵ-net of a subset $A \subset \mathcal{N}$ is a maximal collection of points $\{x_1, \ldots, x_k\}$ of A such that for $i \neq j$, $d(x_i, x_j) > \epsilon$.*

In order to construct an ϵ-net, one needs to have access to the underlying metric space and the distance d between any two points. The construction of the net can happen in a greedy fashion in $O(K|A|)$ time, where K the size of the ϵ-net. There are in fact efficient algorithms that can construct such nets; since this is not the focus of this paper, we refer the interested readers to [11].

Lemma 1. *Given a ball $B_x(R) \subset \mathcal{N}$, and an integer $\ell > 0$, any $(R/2^\ell)$-net $\{x_1, \ldots, x_k\}$ of $B_x(R)$ is such that:*

$$B_x(R) \subset \cup_{i=1}^{k} B_{x_i}(R/2^\ell), \tag{9}$$

and for all $i \neq j$

$$B_{x_i}(R/2^{\ell+1}) \cap B_{x_j}(R/2^{\ell+1}) = \emptyset. \tag{10}$$

Moreover, the cardinality k of any such $(R/2^\ell)$-net is at most $c^{\ell+3}$.

Proof: If (9) does not hold, then there exists y in $B_x(R)$ such that $d(y, x_i) > R/2^\ell$ for all $i = 1, \ldots, k$. This contradicts the maximality of $\{x_1, \ldots, x_k\}$.

For all $i \neq j$, any point z in the intersection $B_{x_i}(R/2^{\ell+1}) \cap B_{x_j}(R/2^{\ell+1})$ is such that

$$d(x_i, x_j) \leq d(x_i, z) + d(x_j, z) \leq 2R/2^{\ell+1} = R/2^\ell.$$

This contradicts the property that $d(x_i, x_j) > R/2^\ell$, hence the intersection $B_{x_i}(R/2^{\ell+1}) \cap B_{x_j}(R/2^{\ell+1})$ is necessarily empty.

Finally, property (10) implies

$$\mu(\cup_{i=1}^{k} B_{x_i}(R/2^{\ell+1})) = \sum_{i=1}^{k} \mu(B_{x_i}(R/2^{\ell+1})).$$

On the other hand, applying $\ell + 2$ times the fact that μ is c-doubling, we deduce that for all $i = 1, \ldots, k$,

$$\begin{aligned} \mu B_{x_i}(R/2^{\ell+1}) &\geq c^{-\ell-2}\mu B_{x_i}(2R) \\ &\geq c^{-\ell-2}\mu B_x(R), \end{aligned}$$

where we used the fact that $B_x(R) \subset B_{x_i}(2R)$, which follows from $x_i \in B_x(R)$. To conclude, note that

$$\cup_{i=1}^k B_{x_i}(R/2^{\ell+1}) \subset B_x(2R).$$

We thus have:

$$\begin{aligned} c\mu(B_x(R)) &\geq \mu(B_x(2R)) \\ &\geq \mu(\cup_{i=1}^k B_{x_i}(R/2^{\ell+1})) \\ &\geq kc^{-\ell-2}\mu(B_x(R)). \end{aligned}$$

The upper bound $k \leq c^{\ell+3}$ follows immediately. ■

We now need the following.

Lemma 2. *Let $\delta \in (0,1)$ verify $\delta > 1/3$. Let the ball $B_x(R)$ be such that there exists a $y \in \mathcal{N}$ for which $d(x,y) = R$ and $\mu(\{y\}) > 0$. Then the following holds. Let $\rho > 0$ be such that $\rho < \min(\delta, (1-\delta)/2)R$, and let $\ell > 0$ be a positive integer such that*

$$2^\ell \left(\frac{R}{2} - \frac{\rho}{1-\delta} \right) > R\frac{2-\delta}{1-\delta}. \tag{11}$$

Then for any $z \in B_x(R)$, one has

$$\mu \left(B_z \left(\frac{\rho}{1-\delta} \right) \right) \leq (1-c^{-\ell})\mu \left(B_x \left(\frac{R}{1-\delta} \right) \right). \tag{12}$$

Proof: Let $z \in B_x(R)$ be fixed. We let $B' := B_z(\frac{\rho}{1-\delta})$. Note that by the assumption that $\rho \leq \delta R$, it follows that B' is included in the ball $B := B_x(\frac{R}{1-\delta})$.

By assumption, there exists $y \in \mathcal{N}$ such that $d(x,y) = R$ and $\mu(\{y\}) > 0$. Thus either $d(x,z)$ or $d(y,z)$ is lower-bounded by $R/2$: indeed, by the triangle inequality,

$$d(x,y) = R \leq d(x,z) + d(y,z).$$

Assume first that $d(x,z) \geq R/2$. By the triangle inequality again, for any $z' \in B'$, one has

$$d(x,z) \leq d(x,z') + d(z,z')$$

so that

$$d(x,z') \geq \frac{R}{2} - \frac{\rho}{1-\delta}.$$

Note that the lower bound $R/2 - \rho/(1-\delta)$ is positive under the assumptions $\rho < (1-\delta)/2R$. In other words, for any $\alpha > 0$, the ball B' is disjoint from the ball B'' defined as

$$B'' := B_x \left(\frac{R}{2} - \frac{\rho}{(1-\delta)} - \alpha \right).$$

This entails that

$$\mu(B'') \leq \mu(B) - \mu(B'). \tag{13}$$

Let now ℓ be an integer verifying (11). A fortiori, ℓ is such that, for some small enough positive α,

$$2^\ell \left(\frac{R}{2} - \frac{\rho}{1-\delta} - \alpha \right) \geq \frac{R}{1-\delta}.$$

This entails that

$$\mu(B) \leq \mu \left(B_x \left(2^\ell \left(\frac{R}{2} - \frac{\rho}{1-\delta} - \alpha \right) \right) \right).$$

Applying ℓ times the c-doubling property of μ, this inequality further implies

$$\mu(B) \leq c^\ell \mu(B'').$$

Combined with (13), this last inequality leads to

$$\mu(B') \leq (1-c^{-\ell})\mu(B),$$

which is the desired bound (12).

Assume next that $d(x,z) < R/2$, so that necessarily $d(y,z) \geq R/2$. Now for any $z' \in B'$, by the triangle inequality one has

$$d(y,z) \leq d(y,z') + d(z,z'),$$

so that, defining now B''' to be

$$B''' := B_y \left(\frac{R}{2} - \frac{\rho}{(1-\delta)} - \alpha \right).$$

For some arbitrarily small $\alpha > 0$, the two balls B' and B''' are disjoint. Note further that B''' is contained B, since for any $z''' \in B'''$, one has

$$d(x,z''') \leq d(x,y) + d(y,z''') \leq R + R/2,$$

and the assumption $\delta > 1/3$ ensures that $(3/2)R \leq R/(1-\delta)$, which is the radius of B.

Similar to (13) we thus have

$$\mu(B''') \leq \mu(B) - \mu(B').$$

Let now ℓ be a positive integer verifying (11). An application of the triangle inequality implies that the inclusion

$$B \subset B_l \left(2^\ell \left(\frac{R}{2} - \frac{\rho}{1-\delta} - \alpha \right) \right)$$

must hold for small enough $\alpha > 0$. Indeed, for any point $x' \in B$, one has

$$d(y,x') \leq R + \frac{R}{1-\delta} = R\frac{2-\delta}{1-\delta},$$

and property (11) guarantees that x' is in the corresponding ball $B_y(2^\ell(R/2 - \rho/(1-\delta) - \alpha))$. Finally, using ℓ times the c-doubling property of μ allows to establish that $\mu(B) \leq c^\ell \mu(B''')$; combined with (13), this leads as in the previous case to the desired property (12). ■

Remark 1. *For a given $R > 0$, the assumptions of Lemma 2 are verified if one takes $\rho = R/4$, $\delta = 1/3+\epsilon$ for small enough $\epsilon > 0$, and $\ell = 5$. Indeed, the condition $\rho < \min(\delta, (1-\delta)/2)R$ holds because $1/4 < 1/3$. Writing $(1-\delta)^{-1} = (3/2)+\epsilon'$ for some arbitrary small positive ϵ', Condition (11) reads after simplification by R:*

$$2^\ell(1/2 - (1/4)(3/2+\epsilon')) > 1 + 3/2 + \epsilon',$$

which is clearly verified for $\ell = 5$ and $\epsilon' > 0$ small enough.

Algorithm 2 ϵ-Net Content Search

```
Input: Oracle$(\cdot,\cdot,t)$, demand distribution $\mu$, starting object $s$,
    embedding $(\mathcal{M},d)$.
Output: target t.
Initialize $x_0 \leftarrow s$.
Initial the search radius $R_0$ according to $R_0 := \sup_{y\in\mathcal{N}} d(x_0,y)$.
$j \leftarrow 0$.
while $x_j \neq t$ do
    Construct an $\left(\frac{R_i}{4}\right)$-net.
    By using the comparison oracle, find the closest object $x_{j+1}$ to the target $t$ among the points in the $\left(\frac{R_j}{4}\right)$-net and $x_j$.
    Update the search radius
        $R_{j+1} = \inf\{R : \mu(B_{x_{j+1}}(R)) = \mu(B_{x_{j+1}}(R_j/4))\}$.
    $j \leftarrow j+1$.
end while
```

B. Algorithm and Upper Bound

The algorithm we propose based on ϵ-nets can be found in Algorithm 2. In short, the search strategy we consider proceeds in stages. We shall denote these stages as $j = 1,\ldots,S$. At the beginning of a stage j, we are given the current best exemplar, denoted x_j, and the current radius of the search, R_j, which is such that in view of the searcher's previous answers, the search target is necessarily within the ball $B_j := B_{x_j}(R_j)$. We shall further impose that at each stage j, the search radius R_j is such that there exists a point $y_j \in \mathcal{N}$ such that $\mu(\{y_j\}) > 0$ and $d(x_j, y_j) = R_j$, *i.e.*, the demand distribution μ puts some mass on the boundary of B_j.

The first stage is initialized by picking an arbitrary initial candidate $x_1 \in \mathcal{N}$. The corresponding initial search radius is then defined as $R_1 := \sup_{y\in\mathsf{supp}(\mu)} d(x_1,y)$. Hence, by construction, this initial ball B_1 indeed has non-zero mass at its boundary.

The search during an arbitrary stage j proceeds as follows. The current search center x_j is completed by additional points of B_j to form a ρ_j-net of B_j, where $\rho_j = R_j/4$. Then the searcher is asked to perform one comparison between her last choice and each of the points of the net that are distinct from x_j. By the end of these comparisons, let x'_j be the last selection of the user. Clearly, this selection is among the points of the net, that which is closest to the target of the search.

Since (in view of Lemma 1) the union of balls centred at the points of the net, and with radius ρ_j, covers entirely the current search ground B_j, it follows that necessarily the target must lie in the ball $B_{x'_j}(\rho_j)$.

We need only one last operation to specify how the next stage $j+1$ is initialized. The center of search at stage $j+1$ will be set to $x_{j+1} := x'_j$. We know that the target lies within $B_{x_{j+1}}(\rho_j)$. We shall then specify the search radius R_{j+1} to be the smallest R such that $\mu(B_{x_{j+1}}(R)) = \mu(B_{x_{j+1}}(\rho_j))$. Thus necessarily, $R_{j+1} \leq \rho_j$, and moreover the minimality of R_{j+1} implies that measure μ puts some mass on the boundary of the resulting search ball B_{j+1}. As such, our algorithm has indeed ensured by construction that at any stage j (a) the target lies in the current ball B_j and (b) the ball contains an object of non-zero mass at its boundary.

We are now ready to bound the number of queries submitted to the oracle by Algorithm 2.

Theorem 3. *The expected search cost of Algorithm 2 can be bounded by*

$$\bar{C}_{\mathcal{F}} \leq \left(c^5 - 1\right)\left(1 + \frac{H(\mu)}{\log(1/(1-c^{-5}))}\right). \qquad (14)$$

Proof: To begin with, observe that at each stage j the searcher is asked to perform one comparison between her last choice and each of the points of the ρ_j-net that are distinct from x_j. The size of this ρ_j-net is, by Lemma 1, at most c^5. Thus, at most $c^5 - 1$ binary comparisons are needed at each stage.

Denote again by x'_j be the last selection of the user at stage j. Let us now denote by $\pi_j := \mu(B_{x_j}(R_j/(1-\delta)))$ the mass put by measure μ on the search ground B_j, after enlarging its radius by a factor $1/(1-\delta)$, where $\delta = 1/3 + \epsilon$, for some small ϵ chosen as in Remark 1. It now follows by Lemma 2 and Remark 1 that necessarily,

$$\mu(B_{x'_j}(\rho_j/(1-\delta))) \leq (1-c^{-5})\pi_j.$$

Note also that, critically, by Lemma 2 and an induction argument, it is guaranteed that at each stage j of the search

$$\pi_j = \mu(B_{x_j}(R_j/(1-\delta))) \leq (1-c^{-5})^{j-1}.$$

To conclude the proof, condition on the target element $z \in \mathcal{N}$. Considering its probability $\mu(\{z\})$ and the previous bound on the probability of the search range after j stages, clearly the search will have completed after j stages provided

$$(1-c^{-5})^{j-1} \leq \mu(\{z\}),$$

or equivalently, provided

$$j \geq 1 + \frac{\log(1/\mu(\{z\}))}{\log(1/(1-c^{-5}))}.$$

The average number of stages, $\bar{S}$, is then upper-bounded by

$$\begin{aligned}\bar{S} &\leq \sum_{z\in\mathcal{N}} \mu(\{z\})\left(1 + \frac{\log(1/\mu(\{z\}))}{\log(1/(1-c^{-5}))}\right)\\ &= 1 + \frac{H(\mu)}{\log(1/(1-c^{-5}))}.\end{aligned}$$

Noting that, within a stage, at most $c^5 - 1$ comparisons are performed, the upper-bound (14) follows. ■

We note again that Theorem 3 gives an upper bound which is matching lower bound (7), up to a discrepancy in the exponent of the doubling constant c. In contrast however to Algorithm 1, which could be implemented only using ordering relationships between objects rather than exact distances, Algorithm 2 indeed requires full knowledge of the underlying metric space. Interestingly, Algorithm 2 *does not require* *knowledge of the target distribution μ.* All steps in the algorithm (and, in particular, the shrinking of the ball B_j to ensure it has non-zero mass at the boundary) can be implemented as long as the support $\mathsf{supp}(\mu)$ is known.

VII. Conclusions

In this work, we studied the problem of content search through comparisons (CSTC) under heterogeneous demands, tying performance to the topology and the entropy of the target distribution. Our study leaves several open problems. The search strategy considered in Algorithm 2 relies on the construction of ϵ-nets at different stage of the search, which necessitates access to detailed information about the geometry of the search space $(\mathcal{M}, d)$, but no information about the demand distribution μ. A challenge could then be to propose simpler strategies, relying on less information about the geometry of the search space. Earlier work on comparison oracles eschewed metric spaces altogether, exploiting what where referred to as *disorder inequalities* [8], [17], [25]. Applying these under heterogeneity is also a promising research direction. Another important issue is to study CSTC under unreliable responses of the oracle. This is in particular very important in practice.

References

[1] R. White and R. Roth, *Exploratory Search: Beyond the Query-Response Paradigm*. Morgan & Claypool, 2009.

[2] G. Marchionini, "Exploratory search: from finding to understanding," *Communications of the ACM*, vol. 49, no. 4, pp. 41–46, 2006.

[3] I. Ruthven, "Interactive information retrieval," *Annual Review of Information Science and Technology*, vol. 42, no. 1, pp. 43–91, 2008.

[4] D. Tschopp, S. N. Diggavi, P. Delgosha, and S. Mohajer, "Randomized algorithms for comparison-based search," in *Neural Information Processing Systems (NIPS)*, 2011.

[5] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the International Conference on Computer Vision*, 1999.

[6] A. W. M. Smeulders, S. Member, M. Worring, S. Santini, A. Gupta, and R. Jain, "Content-based image retrieval at the end of the early years," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 1349–1380, 2000.

[7] M. S. Lew, N. Sebe, C. Djeraba, and R. Jain, "Content-based multimedia information retrieval: State of the art and challenges," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 2, February 2006.

[8] N. Goyal, Y. Lifshits, and H. Schutze, "Disorder inequality: a combinatorial approach to nearest neighbor search." in *ACM International Conference on Web Search and Data Mining (WSDM)*, 2008.

[9] T. M. Cover and J. Thomas, *Elements of Information Theory*. Wiley, 1991.

[10] A. Karbasi, S. Ioannidis, and L. Massoulié, "Content search through comparisons," in *38th International Colloquium Automata, Languages and Programming (ICALP)*, 2011, pp. 601–612.

[11] K. L. Clarkson, "Nearest-neighbor searching and metric space dimensions," in *Nearest-Neighbor Methods for Learning and Vision: Theory and Practice*, G. Shakhnarovich, T. Darrell, and P. Indyk, Eds. MIT Press, 2006, pp. 15–59.

[12] P. Indyk and R. Motwani, "Approximate nearest neighbors: Towards removing the curse of dimensionality," in *ACM Symposium on Theory of Computing (STOC)*, 1998, pp. 604–613.

[13] D. Karger and M. Ruhl, "Finding nearest neighbors in growth-restricted metrics," in *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2002.

[14] R. Krauthgamer and J. R. Lee, "Navigating nets: simple algorithms for proximity search," in *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2004.

[15] B. McFee and G. Lanckriet, "Partial order embedding with multiple kernels," in *Proceedings of the 26th Annual International Conference on Machine Learning (ICML)*. ACM, 2009, pp. 721–728.

[16] I. C. Gormley and T. B. Murphy, "A latent space model for rank data," in *Proceedings of the 2006 conference on Statistical network analysis*. Springer-Verlag, 2007, pp. 90–102.

[17] Y. Lifshits and S. Zhang, "Combinatorial algorithms for nearest neighbors, near-duplicates and small-world design," in *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2009.

[18] A. Karbasi, S. Ioannidis, and L. Massoulié, "Adaptive content search through comparisons," *CoRR*, 2011.

[19] I. J. Cox, M. L. Miller, T. P. Minka, T. V. Papathomas, and P. N. Yianilos, "The bayesian image retrieval system, pichunter: Theory, implementation, and psychophysical experiments," *IEEE Transactions on Image Processing*, vol. 9, no. 1, pp. 20–37, 2000.

[20] Y. Fang and D. Geman, "Experiments in mental face retrieval," in *In Proc. of Audio- and Video-based Biometric Person Authentication*, 2005, pp. 637–646.

[21] M. Ferecatu and D. Geman, "Interactive search for image categories by mental matching," in *IEEE 11th International Conference on Computer Vision (ICCV)*, 2007.

[22] S. Jagabathula and D. Shah, "Inferring rankings under constrained sensing," in *Neural Information Processing Systems (NIPS)*, 2008.

[23] ——, "Inferring rankings under constrained sensing," in *IEEE Transactions on Information Theory*, 2011.

[24] V. F. Farias, S. Jagabathula, and D. shah, "A nonparametric approach to modeling choice with limited data," 2011.

[25] D. Tschopp and S. N. Diggavi, "Approximate nearest neighbor search through comparisons," 2009.