An empirical assessment of quality metrics for diversified similarity searching[1]

Camila R. Lopes[1], Lúcio F. D. Santos[1], Daniel L. Jasbick[2], Daniel de Oliveira[2], and Marcos Bedo[3],[4]

[1] Federal Institute of North of Minas Gerais – Montes Claros, Brazil
{camila,lucio.santos}@ifnmg.edu.br
[2] Institute of Computing, Fluminense Federal University – Niterói, Brazil
{danieljasbick@id, danielcmo@ic}.uff.br
[3] Fluminense Northwest Institute, Fluminense Federal University – S. A. Pádua, Brazil
marcosbedo@id.uff.br

Abstract. A diversified similarity search retrieves elements that are simultaneously similar to a query object and akin to the different collections within the explored data. While several methods in information retrieval, data clustering, and similarity searching have tackled the problem of adding diversity into result sets, the experimental comparison of their performances is still an open issue mainly because the quality metrics are "borrowed" from those different research areas, bringing their biases alongside. In this manuscript, we investigate a series of such metrics and experimentally discuss their trends and limitations. We conclude diversity is better addressed by a set of measures rather than a single quality index and introduce the concept of _Diversity Features Model_ (DFM), which combines the viewpoints of biased metrics into a multidimensional representation. Experimental evaluations indicate _(i)_ DFM enables comparing different result diversification algorithms by considering multiple criteria, and _(ii)_ the most suitable searching methods for a particular dataset are spotted by combining DFM with ranking aggregation and parallel coordinates maps.

1. INTRODUCTION

Similarity searching is the information retrieval process in which the query includes an object, and the answer is composed of a set of elements that are somewhat similar to the query object [Hetland 2020]. Those types of queries are particularly useful for searching large and multidimensional data, _e.g._, audio, images, and videos, which arise from different application domains, such as social networks, biology, and astronomy [Pouyanfar et al. 2018]. An efficient approach for handling similarity searches is the Metric Spaces Model [Hetland 2009], in which data objects are mapped into a known domain where they are compared by a distance function. While feature-learning enables mapping complex domains into simpler spaces [Vincent et al. 2008], _e.g._, images to multidimensional vectors, _similarity criteria_ must be defined for the retrieval of elements according to their distances to the query object [Santos et al. 2013; Zezula et al. 2010; Aggarwal 2015].

The most common criterion for similarity searches is that of _Neighborhood_ (k-NN) _queries_, which retrieves the k-nearest elements to a query object. Neighborhood queries are efficiently executed by

---

index-and-search algorithms [Hjaltason and Samet 2003; Chen et al. 2017; Hetland 2020], but they may present a semantic drawback for searching dense datasets. For instance, suppose a composer runs a similarity search for the *five most similar tunes to the "Beatles Hey Jude"* in a social-network repository and retrieves versions and parodies of the same song. Although correct from a k-NN point-of-view, the answer is likely frustrating to musicians that already known the queried tune in detail. A *diversified similarity search*, however, could consider not only the closest pieces of music but also those of distinct styles, *i.e.*, different *collections* within the data repository.

Result diversification methods fulfill that search criteria according to three main perspectives of diversity, namely *(i)* distance-based, *(ii)* novelty-based, and *(iii)* coverage-based [Drosou et al. 2017]. The first group aim at maximizing a single objective function regarding the distances between the elements within the result set [Zheng et al. 2017], whereas approaches of the second group rely on a two-phase execution in which an enlarged subset of candidates is chosen and then filtered to ensure diversification. Such a two-phase strategy is reduced to the problem of solving a bi-criteria objective function in which similarity and diversity compete ruled by a user-defined linear parameter [Vieira et al. 2011]. Lastly, coverage-based strategies separate objects on-the-fly by following a similarity threshold, which creates *dynamic* clusters in the search space [Santos et al. 2013]. A plethora of result diversification methods are found in the literature [Drosou et al. 2017], but *quality metrics* for assessing their efficacy *are rather scarce* [Santos et al. 2013; Zheng et al. 2017]. In fact, most studies borrow and adapt metrics and indexes originally designed for other information retrieval and similarity searching processes [Drosou and Pitoura 2013; Vieira et al. 2011]. Examples of those metrics include *information retrieval* measures NDCG-IA (*Intent-Aware Normalized Discounted Cumulative Gain*) [Agrawal et al. 2009] and similarity-oriented approaches RB (*Relative Benefit*) and OEM (*Overlap Evaluation Method*) [Smyth and McClave 2001; Santos et al. 2013].

This manuscript extends our previous study entitled *"Quality metrics for diversified similarity searching: What they stand for?"* [Lopes et al. 2020] by *(i)* discussing the biases of existing quality metrics for result diversification, *(ii)* adjusting a set of cluster-oriented metrics, and *(iii)* proposing a new multidimensional model, coined *Diversity Features Model* (DFM), which combines previous relevant metrics from data clustering and similarity-searching for assessing the quality of diversified similarity searching. The intuition behind DFM is that result diversification is better addressed by a multidimensional measure based on the distance distributions collected from both retrieved and data sets rather than a single quality index. The model can also be interpreted as multiple lists of preferences whose top performances are either *(i)* visualized through parallel coordinates maps or *(ii)* identified by ranking aggregation methods, *e.g.*, Median-Rank [Fagin et al. 2003]. As a final contribution, we evaluated DFM by using five result diversification approaches to query real-world datasets, and results indicated the model is capable of comparing diversity search methods by using multiple criteria.

The remainder of the manuscript is divided as follows. Section 2 introduces preliminary concepts on diversity searching and result diversification, Section 3 presents the DFM model, and Section 4 provides the experimental comparisons of metrics and algorithms. Finally, Section 5 concludes our study.

2. BACKGROUND AND RELATED WORK

2.1 Similarity searching

Similarity searches are usually modeled after the *Metric Spaces Model* [Hetland 2009; Chen et al. 2017] due to its *(i)* distance-based semantics [Zezula et al. 2010] and *(ii)* computational-bounded complexity [Hetland 2020]. Formally, a metric space is a pair $\langle \mathbb{O}, \delta \rangle$ of a data domain $\mathbb{O}$ and a distance function δ, which complies with the following properties for any objects $o_q, o_i, o_j \in \mathbb{O}$.

(1) *Symmetry*: $\delta(o_i, o_j) = \delta(o_j, o_i)$,
(2) *Non-negativity*: $\delta(o_i, o_j) \geq 0$,

(3) *Identity*: $\delta(o_i, o_j) = 0 \Leftrightarrow o_i = o_j$, and

(4) *Triangle inequality*: $\delta(o_i, o_j) \leq \delta(o_i, o_g) + \delta(o_g, o_j)$

Examples of such functions include the Minkowski family L_p and the Cosine distance for d-dimensional data, *i.e.*, $\mathbb{O} = \mathbb{R}^d$. Given two objects $o_i, o_j \in \mathbb{R}^d$ and a value $p \in \mathbb{N}_*$, an L_p function measures their distance as in Eq. (1), while the Cosine function expresses their closeness as in Eq. (2).

$$L_p(o_i, o_j) = \left(\sum_{v=1}^{d} (o_{iv} - o_{jv})^p \right)^{\frac{1}{p}} \tag{1}$$

$$Cosine(o_i, o_j) = \left(\sum_{v=1}^{d} o_{iv} \cdot o_{jv} \right) / \left((\sum_{v=1}^{d} o_{iv}^2)^{\frac{1}{2}} \cdot (\sum_{v=1}^{d} o_{jv}^2)^{\frac{1}{2}} \right) \tag{2}$$

For a particular dataset $\mathcal{O} \subseteq \mathbb{O}$ and a query object $o_q \in \mathbb{O}$, the distances from o_q to the elements $o_i \in \mathcal{O}$ can be used as *filters* for the retrieval of objects related to o_q according to distance-based query criteria. Two of those common criteria are the *(i)* Range and *(ii)* Neighborhood queries.

Range Query. A *Range Query* (*Rq*) retrieves the elements in $\mathcal{O}$ that are near to a query object $o_q \in \mathbb{O}$ up to a given distance threshold $\xi \in \mathbb{R}_+$. Accordingly, the result set for a Range query is defined as follows $\mathcal{R} = Rq(o_q, \xi, \mathcal{O}, \delta) = \{o_i \mid o_i \in \mathcal{O}, \delta(o_i, o_q) \leq \xi\}$.

Neighborhood query. A *Neighborhood query* (k-NN) retrieves $k \in \mathbb{N}_*$ elements in $\mathcal{O}$ whose distances to query object $o_q \in \mathbb{O}$ are the smallest. Formally, the result set of a Neighborhood query $\mathcal{R} =$ k-NN$(o_q, k, \delta, \mathcal{O}) = \{o_1, o_2, \ldots, o_k\}$ is incrementally constructed as follows.

$$o_1 = \{o_i \in \mathcal{O} \mid \forall\, o_j \in \mathcal{O},\ \delta(o_i, o_q) \leq \delta(o_j, o_q)\};$$
$$o_2 = \{o_i \in \mathcal{O} \setminus \{o_1\} \mid \forall\, o_j \in \mathcal{O} \setminus \{o_1\},\ \delta(o_i, o_q) \leq \delta(o_j, o_q)\},$$
$$\ldots$$
$$o_k = \{o_i \in \mathcal{O} \setminus \{o_1, o_2, \ldots, o_{k-1}\} \mid \forall\, o_j \in \mathcal{O} \setminus \{o_1, o_2, \ldots, o_{k-1}\}, \delta(o_i, o_q) \leq \delta(o_j, o_q)\}.$$

Neighborhood can be reduced to Range queries by assuming *(i)* the query radius is set to $\xi = \delta(o_q, o_k)$ and *(ii)* the constraint $|Rq| = k$ is ensured[2] [Hjaltason and Samet 2003; Hetland 2009]. Therefore, both Range and k-NN queries may struggle to search high-density datasets, where small variations of radius ξ tend to generate large variations in the result set cardinality because most elements are virtually equidistant to each other [Pestov 2013]. As a consequence, the utility of similarity searches for the exploration of high-density datasets is reduced, and k-NN queries may become unstable because the result sets are likely non-unique[2] for such browsing scenarios. An alternative to soften those problems (without derailing the behavior of the search) is the adding of a *diversity degree* into similarity queries [Drosou et al. 2017; Santos et al. 2018; Jasbick et al. 2020].

2.2 Diversified similarity searching

A generic formulation to include diversity into a result set $\mathcal{R}$ of a k-NN query can be stated as the following optimization problem over a diversity metric *div*.

Result diversification problem. Given a diversity metric *div* and a value $k \leq |\mathcal{O}|, \mathcal{O} \subseteq \mathbb{O}, k \in \mathbb{N}_*$, a diversified result set $\mathcal{R} \subseteq \mathcal{O}$ is the set of k elements in $\mathcal{O}$ so that $\mathcal{R} = \text{argmax}_{\mathcal{R}' \subseteq \mathcal{O}, |\mathcal{R}'|=k} div(\mathcal{R}')$.

Approaches for solving such an optimization problem are categorized in a taxonomy of three groups, namely *(i)* distance-based, *(ii)* novelty-based, and *(iii)* coverage-based [Drosou et al. 2017; Zheng

---

[2]Ties at the kth position are broken arbitrarily.

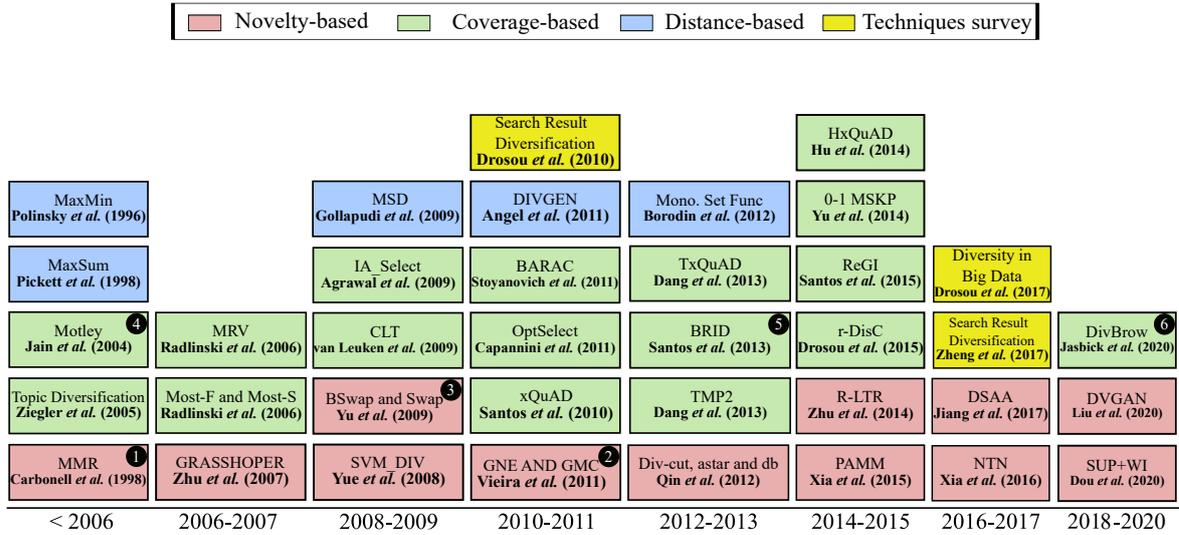| Novelty-based | Coverage-based | Distance-based | Techniques survey |

Fig. 1. A timeline of representative diversity techniques according to the taxonomy proposed by Drosou et al. (2017). We implemented the numbered approaches for the empirical comparisons reported in Section 4.

et al. 2017]. Figure 1 presents a timeline of representative result diversification techniques by highlighting their categories and identifying the approaches (❶, ❷, ❸, ❹, ❺, and ❻) we implemented for the empirical comparisons reported in this manuscript.

Distance-based methods. Those approaches rely on the straightforward definition of the argmax function for the optimization problem. For instance, methods MaxMin and MaxSum aim at maximizing argmax as the minimal and maximal sum of div values, respectively [Polinsky et al. 1996]. Such methods fit poorly into Range and Neighborhood queries because they examine the entire dataset $\mathcal{O}$ regardless of the query object [Zheng et al. 2017], i.e., the result set is the same for any query element, which expressively reduces the search semantics.

Novelty-based methods. Those approaches model the optimization problem as a *dual-criteria* function in which *similarity* and *diversity* compete against each other ruled by a linear λ parameter defined in the $[0, 1]$ interval. Setting parameter λ to 0 (i.e., marking diversity as irrelevant) turns the optimization problem into a Neighborhood query, whereas the increasing of λ parameter will force the retrieved elements to be apart from the query object. The finding of the optimal λ value is an NP-hard problem [Drosou et al. 2017], and, therefore, practical solutions rely on either heuristics or meta-heuristics for producing suitable outputs [Vieira et al. 2011].

The *Maximal Marginal Relevance* (MMR – ❶) method [Carbonell and Goldstein 1998] models the optimization problem for Neighborhood queries by considering the query object $o_q \in \mathbb{O}$ and by assigning a score for each element o_i in the dataset $\mathcal{O} \subseteq \mathbb{O}$ according to the objective function argmax $= MMR(o_i, o_q) = (1 - \lambda) \cdot \delta_{sim}(o_i, o_q) + 2 \cdot \lambda \cdot \sum_{o_j \in \mathcal{R}} \delta_{div}(o_i, o_j)$, where *similarity* and *diversity* are calculated by two distance functions δ_{sim} and δ_{div}, respectively. The result set $\mathcal{R}$ is constructed in k greedy and incremental steps, in which the element $o_i \in \mathcal{O} \setminus \mathcal{R}$ with highest $MMR(o_i, o_q)$ value is selected as the *next* nearest and diversified neighbor.

Analogously, the approach *Greedy with Marginal Contribution* (GMC – ❷) [Vieira et al. 2011] models the objective function as argmax $= MMC(o_i, o_q) = (1-\lambda) \cdot \delta_{sim}(o_i, o_q) + 2 \cdot \lambda \cdot \sum_{o_j \in \mathcal{R}} \delta_{div}(o_i, o_j) + 2 \cdot \lambda \cdot \sum_{o_h \in \mathcal{R}'}^{\mathcal{R}' \subseteq \mathcal{O} \setminus \mathcal{R}, |\mathcal{R}'| = h - |\mathcal{R}|} \delta_{div_2}(o_i, o_h)$ for calculating the weighted contribution of elements o_h *outside* the result set $\mathcal{R}$ by using the distance functions δ_{div} and δ_{div_2}. Figures 2(b) and (e) show an example of a GMC-oriented query in comparison to that of a k-NN query without result diversification.
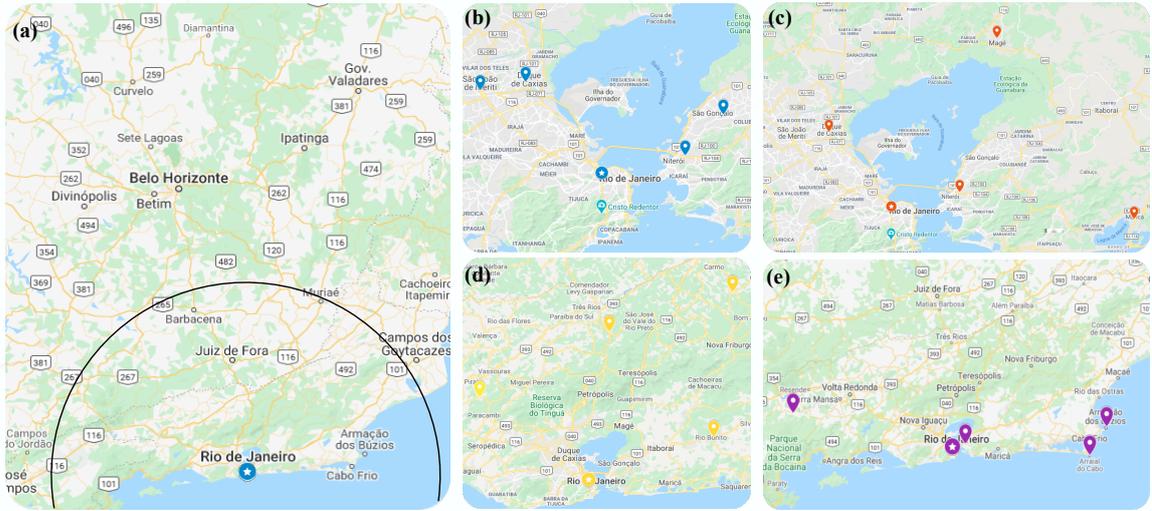
Fig. 2. Result sets for the query "Find the 04 nearest and diversified cities to Rio de Janeiro" (a) Query set $\mathcal{O}'$. (b) k-NN. (c) Motley. (d) BRID. (e) GMC. Search parameters are $\delta_{sim} = \delta_{div} = \delta_{div_2} = L_2$, $\lambda = 0.5$, and $r = 0.5$.

The *Swap* (Swap – ❸) method [Yu et al. 2009] relies on distance-based permutations to avoid being stuck with local maxima result sets. Initially, Swap builds a baseline result set $\mathcal{R}$ with the k closest elements to the query object, *i.e.*, a standard k-NN query. The elements $o_i \in \mathcal{O} \setminus \mathcal{R}$ are then *sorted* by distance δ_{sim} values, and individually swapped with $\mathcal{R}$ objects according to a given objective function, *e.g.*, MaxSum, MMR, or MMC. Whenever the swap produces a better function score, the changes in $\mathcal{R}$ are made permanent in a greedy procedure. The method stops if every object in $\mathcal{O}$ is swapped at least once.

Since heuristic-driven methods such as GMC and Swap are designed to generate approximate answers, data sampling is commonly used for speeding-up their execution in practice. Accordingly, those methods execute two-stage processing whose first phase is choosing a query set $\mathcal{O}'$ from the original dataset $\mathcal{O}$, $|\mathcal{O}'| \ll |\mathcal{O}|$ to be used as the input for the optimization problem.

Coverage-based methods. This approach differs from the others as its main premise is creating a distance-separation between elements in the search space. Accordingly, the model considers as diverse the objects which are set apart by a threshold that complies with pre-defined criteria.

Method *Motley* (Motley – ❹) [Jain et al. 2004] relies on using a user-provided separation distance $r \in \mathbb{R}_+$ for retrieving nearest neighbors set apart from each other. Given a query object $o_q \in \mathbb{O}$, *Motley* sorts the elements $o_i \in \mathcal{O}$ regarding their distances to o_q and includes the first nearest neighbor into a partial result set $\mathcal{R}'$. Then, it incrementally examines the sorted list of nearest candidates $o_i \in \mathcal{O}$. Element o_i is added to the partial result set only if it is far from any object in $\mathcal{R}'$ by at least r, *i.e.*, $\delta(o_i, o_j) > r, \forall\, o_j \in \mathcal{R}'$. The procedure continues until *(i)* the remaining $k-1$ diverse neighbors are found or *(ii)* the list of candidates is empty.

Better Results with Influence Diversification (BRID – ❺) [Santos et al. 2013] extends the Motley strategy to handle dynamic thresholds by benefiting from properties of the Metric Space Model. As a result, BRID creates dynamic distance-separation limits and eliminates the need for extra user-provided parameters. BRID main concept is using the *inverse* of the distance between two objects to counter-balance similarity and diversity. This concept is known as the *Influence* function – $I(o_i, o_j)$, and measures the mutual impact of the distance between two objects $o_i, o_j \in \mathbb{O}$, as in Eq. (3).

$$I(o_i, o_j) = 1/\delta(o_i, o_j) \tag{3}$$

The method extensively uses the Influence function to create a *ternary* relationship between a tuple of three objects $o_h, o_i, o_j \in \mathbb{O}$, in which o_h is said to be *more influenced* by o_i than by o_j whenever $I(o_h, o_i) \geq I(o_h, o_j)$. Therefore, given a query object $o_q \in \mathbb{O}$ and a data element $o_i \in \mathcal{O} \subseteq \mathbb{O}$, BRID creates a *region of exclusion*, coined *strong influence set* $-\ddot{I}_{o_q, o_i}$, which leverages the distance between the pair $\langle o_q, o_i \rangle$ and delimits their influence coverage set as in Eq. (4).

$$\ddot{I}_{o_q, o_i} = \{o_j \in \mathcal{O} \mid (I(o_i, o_j) \geq I(o_i, o_q)) \wedge (I(o_j, o_i) \geq I(o_j, o_q))\} \tag{4}$$

BRID applies those regions of exclusion for binding a query object o_q, a dataset $\mathcal{O}$ and a result set $\mathcal{R}$, and enables the retrieval of the k closest elements to o_q in $\mathcal{O}$ that are also outside the strong influence sets of the objects in $\mathcal{R}$. As a result, Neighborhood queries are seamlessly extended into Diversified Neighborhood queries by the BRID approach as follows.

Diversified Neighborhood query. A *Diversified Neighborhood query* retrieves $k \in \mathbb{N}_*$ non-influenced elements in $\mathcal{O}$ whose distances to query object $o_q \in \mathbb{O}$ are the smallest so that $\mathcal{R} = \{r_i \in \mathcal{O} \mid \forall o_j \in \mathcal{R} : r_i \notin \ddot{I}_{o_j, o_q} \wedge \forall o_i \in \mathcal{O} \setminus \mathcal{R} : \left(\delta(r_i, o_q) \leq \delta(o_i, o_q) \vee \exists o_j \in \mathcal{R} : o_i \in \ddot{I}_{o_j, o_q} \right) \wedge |\mathcal{R}| \leq k\}$.

Method *Diversity Browsing* (DivBrow – ❻) [Jasbick et al. 2020] provides an incremental implementation of BRID by taking full advantage of metric indexing. Although faster than BRID for the execution of diversified similarity searches, DivBrow and BRID produce the same outputs[3] because they are both based on *Influence* and *strong influence set* concepts. Figures 2(c) and (d) show a comparison between the result sets retrieved by BRID/DivBrow and that of Motley.

2.3 Quality metrics for diversified similarity searching

Although a plethora of strategies have been proposed to address result diversification, the design of a proper evaluation metric for measuring their intrinsic quality was mostly overlooked [Zheng et al. 2017; Drosou et al. 2017]. In fact, metrics for quantifying results from diversified similarity searching are mainly "borrowed" from information retrieval, similarity searching, and data clustering [Vieira et al. 2011; Drosou and Pitoura 2013; Santos et al. 2014; Aggarwal 2015].

Information Retrieval Metrics. A common quality measure borrowed from information retrieval is the *Intent-Aware Normalized Discounted Cumulative Gain* (NDCG-IA) metric [Agrawal et al. 2009]. It extends the *Normalized Discounted Cumulative Gain* by considering the *labels* of the recovered objects for assigning scores regarding a baseline set of classes. While both NDCG-IA and NDGC provide a good quantification regarding the label-oriented quality of recovered sets, they are unsuitable for handling *unsupervised* datasets, *i.e.*, non-labeled data objects, which is usually the condition of information stored into real-world datasets.

Similarity Searching Metrics. On the other hand, similarity searching metrics disregard labels and examine the distances between the result set elements.

The *Relative Benefit* (RB) [Smyth and McClave 2001] metric quantifies the *trade-off* between similarity and diversity by using k-NN queries as a baseline, *i.e.*, RB assumes the results of a k-NN query represent similarity only so that the diversity brought by another algorithm can be calculated by counting the differences between its result and that of a k-NN. However, while RB is able to spot how much a method diverges from k-NN, it does not quantify the quality of diversity itself, which renders the straight comparison of two result diversification methods harder.

Another approach is the proposal of the *Overlap Evaluation Method* (OEM) [Santos et al. 2013] that relies on calculating the distances between the elements in the dataset and the search object for

---

[3]In the absence of k-th distance ties.

reconstructing their strong influence sets and regions of exclusions. Accordingly, OEM compares the obtained sets against the ideal strong influence sets found at a Diversified Neighborhood query by calculating the overlap among the reconstructed and ideal sets. Such a comparison is carried out as the counting of objects lying at the intersection of both regions, as in Eq. (5).

$$\text{OEM}(\mathcal{R}, o_q) = 1/2 \cdot \sum_{o_i \in \mathcal{R}} \sum_{o_j \in \mathcal{R}} \left(1 - (|\ddot{I}_{o_i,o_q} \cap \ddot{I}_{o_j,o_q}| / |\ddot{I}_{o_i,o_q} \cup \ddot{I}_{o_j,o_q}|)\right), o_i \neq o_j \tag{5}$$

where the higher value of Eq. 5, the better the diversity algorithm. Notice, however, OEM has a clear bias for favoring strategies that produce result sets whose objects do not influence themselves.

A yet similarity-oriented metric is the *Dissimilarity Feature Method* (DisFM) [Santos et al. 2013] that introduces the idea of using multiple quality measures rather than a single index for addressing diversity. As the first step, the quality metric calculates simple statistics calculated from the distance distribution within result sets, namely minimum, maximum, mean, and standard deviation. Next, it aggregates those measures for consolidating the values as a single index through a user-calibrated weighted sum of the normalized statistics.

Cluster Metrics. Relative measures from data clusters provide an alternative for measuring the quality of diversity methods since real-world data are not always labeled. In particular, *Silhoutte*, *Dunn*, and *Davies-Boudin* indexes bring another cohesion/separation bias to complement the analysis of result diversification. Although several methods employ such a rationale for taking advantage of cluster-oriented metrics and their biases [Jain et al. 2004; Drosou and Pitoura 2013; Santos et al. 2014], a formal adjusting of those methods for a fair examination of both novelty-based and coverage-based diversity searching algorithms is still lacking.

3. MATERIAL AND METHODS

3.1 An extension of cluster indexes for diversified similarity searching

Relative measures for cluster validation quantify the differences of cohesion and separation between two data partitions constructed following a group criterion. Formally, let the dataset $\mathcal{O}$ be divided into k clusters $\mathcal{C} = \{C_1, C_2, \ldots, C_k\}, \cup_{i=1}^k C_i = \mathcal{O}$, and each subgroup C_i be represented by an element $o_{c_i} \in \mathcal{O}$, relative metrics target a dual optimization problem in which *(i)* inner cluster distances must be minimal, and *(ii)* distances between clusters' representatives must be maximal[4]. The *Silhouette index* $(Sil(\mathcal{C}))$ [Rousseeuw 1987] models that optimization problem by using the average distances within clusters as a normalization factor – Eq. (6).

$$Sil(\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{C_k \in \mathcal{C}} \bar{S}(C_k); \bar{S}(C_k) = \frac{1}{|C_k|} \sum_{o_i \in C_k} \hat{S}(o_i); \hat{S}_{o_i \in C_k}(o_i) = \frac{\min_{o_i \in C_k}(o_i) - \mu_{o_i \in C_k}(o_i)}{\max\{\min_{o_i \in C_k}(o_i), \mu_{o_i \in C_k}(o_i)\}}$$

$$\min_{o_i \in C_k}(o_i) = \min_{C_j \in \mathcal{C} \setminus C_k}\left(\frac{1}{|C_j|} \sum_{o_h \in C_j} \delta(o_i, o_h)\right); \mu_{o_i \in C_k}(o_i) = \frac{1}{|C_k| - 1} \sum_{o_j \in C_k} \delta(o_i, o_j)$$

$$\tag{6}$$

Such a rationale can also be used for comparing how representative the retrieved elements $\mathcal{R}$ of a diversity search is with regards to the dataset $\mathcal{O}$. Accordingly, we extend $Sil(\mathcal{C})$ into $Sil^*(\mathcal{R}, \mathcal{O}', \mathcal{C})$ for measuring the quality of a result set $\mathcal{R}$ regarding a query set $\mathcal{O}' \subseteq \mathcal{O}$, as in Eq. (7).

---

[4]The representative object o_{c_i} of a cluster C_i is commonly chosen as the group average *medoid*, *i.e.*, the object whose mean distance to other elements in the cluster is minimal.

$$Sil^*(\mathcal{R}, \mathcal{O}', \mathcal{C}) = \frac{\Psi(\mathcal{C})}{|\mathcal{C}|}; \Psi(\mathcal{C}) = \frac{\sum_{o_j \in C_i, C_i \in \mathcal{C}} \delta_{div}(o_j, o_h)}{\psi(C_i)}; \psi(C_i) = \frac{\sum_{o_j \in C_i} \delta_{div}(o_j, o_{c_i})}{|C_i|} \qquad (7)$$

where $o_h \in \mathcal{O}'$ is the representative element of cluster C_h that is the closest to C_i, i.e., $\delta(o_{c_i}, o_h) \leq \delta(o_{c_i}, o_{c_j})$ for any cluster $C_j \in \mathcal{C}$. Another relative-based cluster measure is the *Dunn index* $(Dunn(\mathcal{C}))$, which models inner and outer cluster separation according to their diameters – Eq. (8).

$$Dunn(\mathcal{C}) = \frac{\min_{C_k, C_h \in \mathcal{C}} \left(\min_{o_i \in C_k, o_j \in C_h, o_i \neq o_j} \delta(o_i, o_j) \right)}{\max_{C_k \in \mathcal{C}} \left(\max_{o_i, o_j \in C_k, o_i \neq o_j} \delta(o_i, o_j) \right)} \qquad (8)$$

We extend the Dunn index into $Dunn^*(\mathcal{R}, \mathcal{O}', \mathcal{C})$ by using result set entries $o_i \in \mathcal{O}'$ and their distances to the query object $o_q \in \mathbb{O}$. Accordingly, the largest inner cluster distance is normalized by the diversity to the query element, as in Eq. (9).

$$Dunn^*(\mathcal{R}, \mathcal{O}', \mathcal{C}) = \min_{C_i, C_j \in \mathcal{C}, C_i \neq C_j} \left(\delta_{div}(o_i, o_j) / \max_{C_m \in \mathcal{C}} \left(\max_{o_h \in C_m} \delta_{div}(o_h, o_q) \right) \right) \qquad (9)$$

Finally, the *Davies-Bouldin index* $(DB(\mathcal{C}))$ models the quality of the clusters by using the mean distance among them for normalization – Eq. (10).

$$DB(\mathcal{C}) = \frac{1}{|\mathcal{C}|} \sum_{C_i, C_j \in \mathcal{C}, C_i \neq C_j} \max \left(\frac{\mu(C_i) + \mu(C_j)}{\delta(o_{c_i}, o_{c_j})} \right); \mu(C_k) = \frac{1}{|C_k| - 1} \sum_{o_i \in C_k} \delta(o_i, o_{c_k}) \qquad (10)$$

Accordingly, we extend the Davies-Bouldin index into $DB^*(\mathcal{R}, \mathcal{O}', \mathcal{C})$, as in Eq. (11).

$$DB^*(\mathcal{R}, \mathcal{O}', \mathcal{C}) = \frac{1}{|\mathcal{C}|} \cdot \sum_{C_i, C_j \in \mathcal{C}, C_i \neq C_j} \max \left(\frac{diam(C_i) + diam(C_j)}{\delta_{div}(o_{c_i}, o_{c_j})} \right); diam(C_k) = \max_{o_i, o_j \in C_k} \delta_{div}(o_i, o_j)$$
$$(11)$$

3.2   Uncovering clusters from result sets

Another challenge in adopting cluster-oriented metrics is that diversified similarity searching does not *directly* produce groups on the query set. As an adaptation, retrieved objects can be employed as cluster *representatives* found by the searching algorithm. Accordingly, we adopt the objects of the result set $\mathcal{R}$ as "*medoids*" for the definition of clusters over the query set $\mathcal{O}'$. Such a rationale coupled to novelty-based diversity algorithms generates partitions similar to those of the *k*-Medoids clustering method [Aggarwal 2015], in which $\mathcal{O}'$ elements are assigned to the closest object in $\mathcal{R}$ rather than a *de facto* medoid. Figures 3(c–d) illustrate the approach where $\mathcal{O}'$ delimits a search region in $\mathcal{O}$, and query objects are clustered around $\mathcal{R}$ entries.

Coverage-based algorithms retrieve diversified objects by using distance-separation thresholds. Accordingly, the query set $\mathcal{O}'$ can be seen as the union of the regions in the search space delimited by the center objects in $\mathcal{R}$ and coverage radii equal to r distance thresholds. Figures 3(e–f) show an example of clusters constructed after Motley for a neighborhood value of $k = 3$. Likewise, BRID result set $\mathcal{R}$ can be used for the definition of clusters over $\mathcal{O}'$ according to the dynamic thresholds that define their *influences* to the query object o_q. The difference is the separation distance r for every cluster in BRID is defined by the strong influence sets according to the distance between the non-influenced

elements in $\mathcal{R}$ and o_q. Therefore, the clusters are the regions in the search space centered at result set entries with coverage radii as large as their influences over the query object. Figures 3(g–h) illustrate a cluster formation generated after BRID.
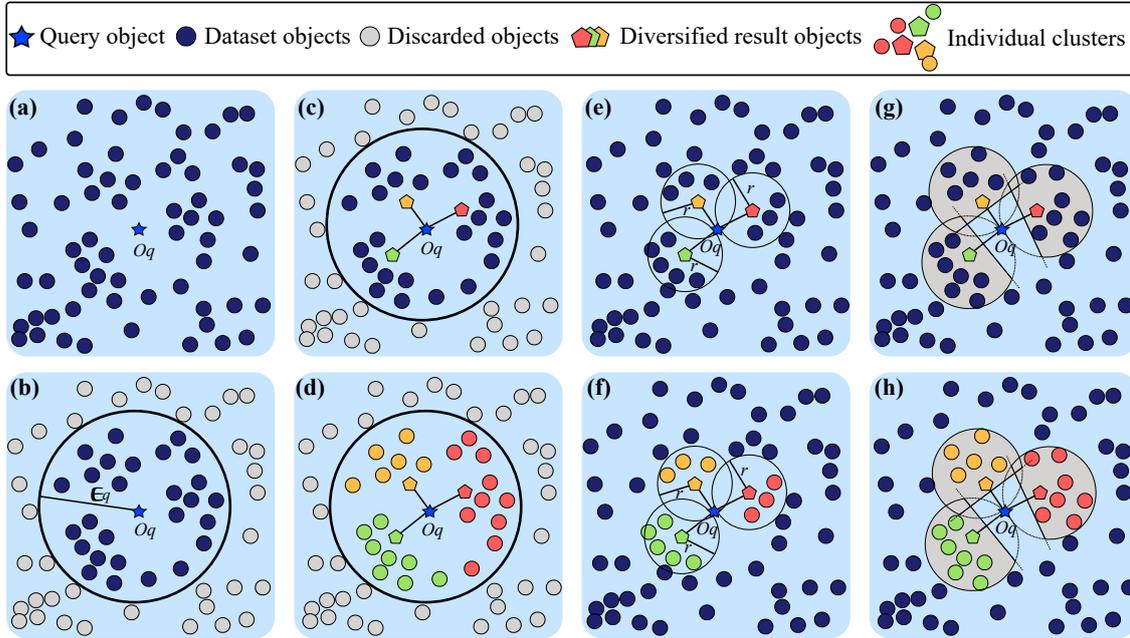

Fig. 3. Clusters for diversified result sets. (a) Dataset $\mathcal{O}$, (b) Query set $\mathcal{O}' \subseteq \mathcal{O}$, (c) $\mathcal{O}'$ defined for novelty-based methods, (d) $k = 3$ clusters constructed after novelty-based methods, (e) $\mathcal{O}'$ defined for Motley, (f) $k = 3$ clusters formed after Motley, (g) $\mathcal{O}'$ defined for BRID, and (h) $k = 3$ clusters defined following BRID *influences*.

3.3   The Diversity Features Model – DFM

Cluster-oriented indexes quantify the cohesion and separation produced by diversity searching on top of query sets, but they may favor approaches that produce cluster-like partitions in the search space, *i.e.*, coverage-based algorithms. This argument is reinforced by the empirical observations reported in Section 4.1, in which cluster indexes spotted Motley or BRID as the most suitable searching routine for result diversification in 9 out of 12 comparisons.

Accordingly, a more global evaluation of *diversity* can be achieved by means of a *set of quality measures* rather than a single index value, which softens the weight of individual biases. Here, we introduce the *Diversity Features* Model (DFM), a multidimensional approach that combines the viewpoints of both cluster and similarity-oriented quality metrics. DFM $(\mathcal{R}, \mathcal{O}', \mathcal{C})$ produces a seven-dimensional output as the quality measure for each data-oriented evaluation of a result diversification algorithm. Such output is calculated by using the distance distributions within both retrieved and query sets according to the cluster-oriented and statistic measures of Eq. (12).

The consolidation of DFM values into a single score depends on the subject domain and can be conducted following two ranking strategies, namely *parallel coordinate maps* [Inselberg and Dimsdale 1990] and *ranking aggregation* [Fagin et al. 2003; Ciaccia and Martinenghi 2017].

(1) *Parallel coordinate maps*: Enables labeling the competing algorithms by assuming every normalized DFM dimension is as equally as relevant. The classification requires the ordering of the plot areas for finding the most suitable competitor.

(2) *Ranking aggregation*: Enables the sorting and ranking of algorithms according to their multidimensional DFM representations, which must comply with previously *ranking* criteria. The Median-Rank [Fagin et al. 2003] is a ranking criterion example that enables finding the competitor with the most frequently high DFM values.

$$\text{DFM}(\mathcal{R}, \mathcal{O}', \mathcal{C}) = \langle Sil^*, Dunn^*, RB^*, \mu_{div}, \sigma_{div}, \mu_{sim}, \sigma_{sim} \rangle$$

$$\mu_{div} = \frac{1}{2 \cdot (k^2 - k)} \cdot \sum_{o_i \in \mathcal{R}} \sum_{o_j \in \mathcal{R}, o_j \neq o_i} \delta_{div}(o_i, o_j)$$

$$\sigma_{div} = \sqrt{\frac{1}{2 \cdot (k^2 - k)} \cdot \sum_{o_i \in \mathcal{R}} \sum_{o_j \in \mathcal{R}, o_j \neq o_i} (\delta_{div}(o_i, o_j) - \mu_{div})^2} \quad (12)$$

$$\mu_{sim} = 1/k \cdot \sum_{o_i \in \mathcal{R}} \delta_{sim}(o_i, o_q)$$

$$\sigma_{sim} = \sqrt{\frac{1}{k - 1} \cdot \sum_{o_i \in \mathcal{R}} (\delta_{sim}(o_i, o_q) - \mu_{sim})^2}$$

4. EXPERIMENTS

This section reports an empirical evaluation of the reviewed quality metrics for diversified similarity searching. Table I describes the seven real-world datasets[5,6,7,8,9] that we used in our trials by detailing their cardinality $|\mathcal{O}|$, dimensionality $\mathbb{R}^d$, intrinsic dimensionality $\lceil \mathcal{D} \rceil$, and distance function[10]. Data intrinsic dimensionality was calculated as the rounded value of $\mathcal{D} = \mu^2(\delta(o_i, o_j))/2 \cdot \sigma^2(\delta(o_i, o_j)); o_i, o_j \in \mathcal{O} \wedge o_i \neq o_j$ according to the formulae in [Chávez et al. 2001; Pestov 2013]. Every experiment was conducted by a *holdout* split in which 100 random elements were selected to be used as query objects, and the remaining data elements were employed as the dataset $\mathcal{O}$. The query set $\mathcal{O}'$ within dataset $\mathcal{O}$ was restricted because we needed to bound the factorial-based number of combinations required by novelty-based algorithms so the evaluations could finish within a reasonable time (weeks). We defined $|\mathcal{O}'|$ by using the average radii that cover $10\times$ the maximum number of neighbors required in a k-NN query with result diversification. Accordingly, $|\mathcal{O}'|$ was limited to 300, 300, 300, 200, 300, and 500 in datasets WINE, US_CITIES, SPAM, PHOTO_F, NASA, and FACES respectively. The Motley distance r threshold was empirically determined for each query set $\mathcal{O}'$ to ensure that exactly k neighbors were returned. Every algorithm was implemented in Python version 3.6.8 in a local machine Linux Mint 19.2 with an $i7$ processor, 16GB RAM, and a 1TB SATA disk.

4.1 Diversity searching quality metrics

We executed 100 neighborhood queries with result diversification by using *(i)* the 100 random query objects separated from the original datasets, and *(ii)* a query parameterization of $k = \{5, 7, \ldots, 17\}$. In the experiments, Motley's parameter was set to $r = 10, 0.3, 10, 1.10^4, 0.3$, and 25 in datasets WINE, US_CITIES, SPAM, PHOTO_F, NASA, and FACES, respectively. The quality metric *Objective function* was set to the MMR scoring function – See Section 2. We also normalized the outputs so that the higher the individual metric value, the better the algorithm in the adding of diversification. As a result, the following plots present the *average* measures calculated after the execution of the 100 searches.

---

[5] WINE dataset is available at `https://archive.ics.uci.edu/ml/datasets/wine`

[6] US_CITIES dataset is available at `https://public.opendatasoft.com/explore/dataset/us-cities-demographics/`

[7] SPAM dataset is available at `https://archive.ics.uci.edu/ml/datasets/spambase`

[8] FACES and NASA datasets are available at `http://www.sisap.org/dbs.html`

[9] PHOTO_F dataset is available at `http://www.informedia.cs.cmu.edu/`

[10] Distance function δ_{sim} and δ_{div} were set to δ.

Table I.   List of evaluated datasets.

Name	$\lvert\mathcal{O}\rvert$	$\mathbb{R}^d$	$\lceil\mathcal{D}\rceil$	δ	Description
WINE [5]	178	13	2	L_2	Chemical features extracted from wines.
US_CITIES [6]	25,375	2	2	L_2	Geographic entries of U.S. cities.
SPAM [7]	4,601	57	3	L_2	Fingerprints within email spams.
PHOTO_F [8]	300	256	5	L_2	Features from photos of human faces.
NASA [9]	40,150	20	6	L_2	Features from NASA/SISAP images.
FACES [8]	1,016	761	52	L_1	Characteristics from face images.

In the parameter exploration stage, we examined the impact of λ parameter over novelty-based algorithms. Due to space limitations, we report only the representative case analysis for a varying range of λ values ($\lambda = \{0.0, 0.3, 0.5, 0.7, 1\}$) and a fixed number of neighbors ($k = 9$) for the dataset US_CITIES, which summarizes the dominant behavior we found by querying other datasets with the same setup. Figure 4 shows the output of quality metrics for novelty-based methods Swap, MMR, and GMC. Cluster-oriented measures Sil^*, $Dunn^*$, and DB^* increased with λ, but similarity-driven metrics peaked with distinct and intermediate λ values. The best average performance was reached with value $\lambda = 0.5$, which we use as the querying setup for the following experiments.

Next, we executed 100 queries over each query set with the competing result diversification algorithms. Figures 5 and 6 present the average values reached by novelty and coverage-based methods by query set. Results show cluster-oriented metrics Sil^*, $Dunn^*$ and DB^* have separated those two groups of approaches by assigning better scores to coverage-based algorithms. Silhouette endorsed BRID as the most suitable choice, *Davies-Bouldin* spotted Motley, and *Dunn* best behavior was either BRID or Motley depending on the k value. Notice, while different cluster-oriented metrics produced different outputs, they all indicate coverage-based methods. Moreover, experimental results pinpoint that cluster-oriented metrics may struggle in the measuring of diversity in high-dimensional result sets, which is observed by the ratio between similarity-only k-NN and coverage-based diversity methods. In the experiments, this ratio reduces (or inverts) for increasing values k regarding high-dimensional query sets PHOTO_F, and FACES, which indicates cluster-oriented measures may be correlated with the intrinsic dimensionality – See Table I.

The highest scores of metric *Objective function* were achieved by algorithm MMR (which aims at maximizing the scoring function itself), but no other significant differences were found between the quality of MMR and either novelty-based (*e.g.*, Swap) or coverage-based approaches (*e.g.*, Motley). Likewise, while the higher scores of influence-based metric OEM were drawn by BRID (which also
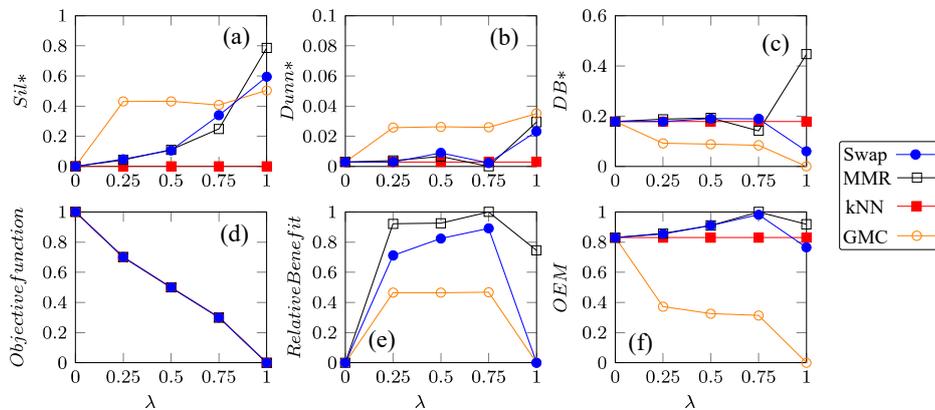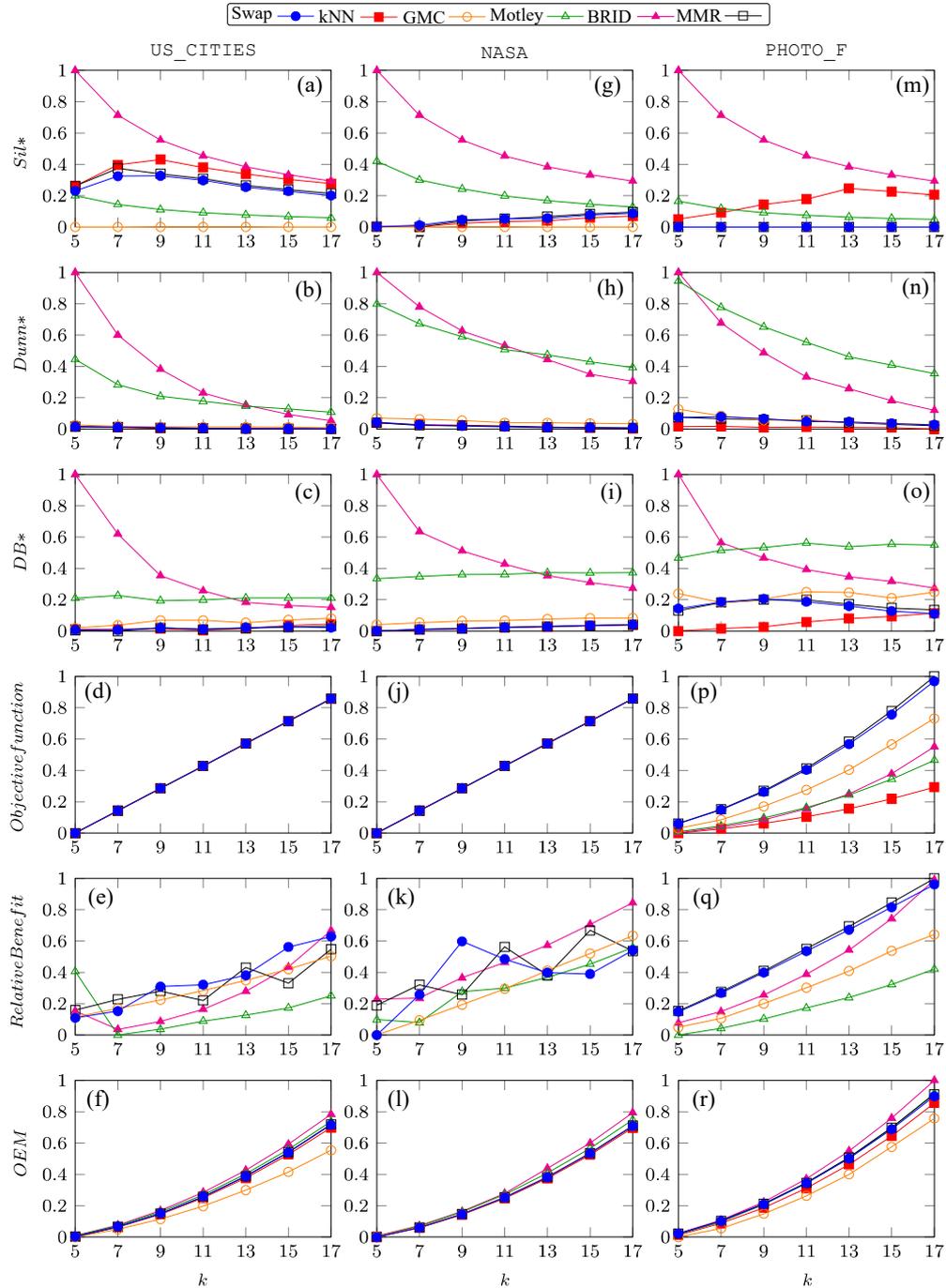

Fig. 4.   Quality scores for different λ values.

Fig. 5.   Quality of searching methods according to the six compared metrics. **Part 1/2 − ↑ Better.**

uses influences for identifying diversity), the metric was unable to spot other coverage-based methods. Altogether, OEM-oriented results indicate influences were also found by novelty-based approaches as in the MMR and Swap performances over US_CITIES. Finally, similarity-based RB metric was also unable to separate novelty and coverage-based methods, as no consistent differences were observed between novelty-based methods over competitors BRID and Motley. Those findings indicate:

(1) cluster-oriented metrics favor coverage-based diversity approaches, which is explained by the separation principle found within clustered data,

(2) cluster-oriented metrics may correlate with data intrinsic dimensionality,

(3) similarity-driven metrics were unable to assert fair and global winners because they promoted algorithms that share their own rule of quality, *e.g.*, OEM and BRID.
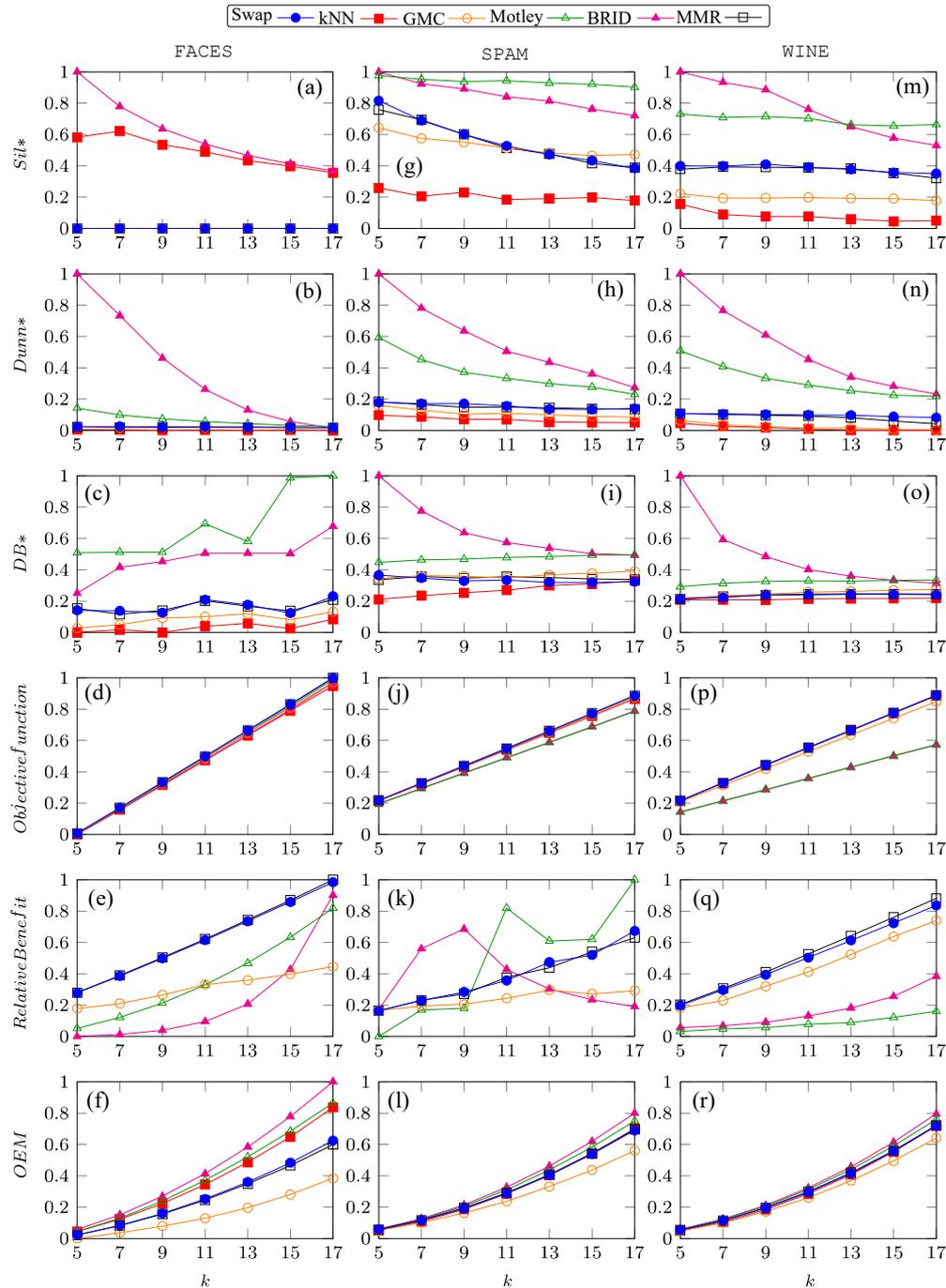

Fig. 6.   Quality of searching methods according to the six compared metrics. **Part 2/2 − ↑ Better.**

4.2 DFM and diversity searching

This final experiment aims at examining *(i)* how those biases can be softened through the DFM multi-dimensional metric, and *(ii)* the behavior of DFM as a single and consolidated index. Accordingly, we compare the quality of distinct diversity searching algorithms through the seven-dimensional metrics of the DFM model. The quality model metrics were extracted from the same experiments of Section 4.1, *i.e.*, a holdout search setup with fixed values for parameters lambda (regarding novelty-based methods) and r (for coverage-based Motley algorithm). Finally, DFM outputs were consolidated and evaluated by means of two different strategies:

(1) a visual quality assessment, through *parallel coordinates maps*, and

(2) a numeric comparison, by using the *Median-Rank* ranking aggregation algorithm.

We also juxtaposed the DFM values corresponding to the compared algorithms and those produced by a k-NN query in both evaluations, *i.e.*, k-NN values were employed as a baseline comparison factor.

Parallel coordinates maps. Figures 7(a–f) present the parallel coordinates maps for DFM outputs regarding the five compared result diversification methods. The plots were generated for the inflexion value $k = 9$ (see Figures 5 and 6) and DFM multidimensional outputs were normalized in the [0, 1] interval. Each entry in the plots represents the average and normalized value of DFM calculated for a given result diversification method after the execution of 100 queries. DFM outputs for baseline k-NN queries form the innermost and smallest polygon inside the parallel coordinates so that distortions towards the map borders indicate better diversification.

Results indicate statistic-driven measures attracted novelty-based methods, whereas cluster-oriented metrics emphasized coverage-based methods. Accordingly, individual winners were found for each examined set. For instance, the area for GMC and Swap methods were the largest regarding query sets SPAM, FACES and PHOTO_F, whereas methods BRID and Motley showed the largest differences in comparison to k-NN in the query sets US_CITIES and NASA. Both approaches presented close areas for the WINE query set, with a slight edge of Swap over BRID. Overall, those findings indicate:
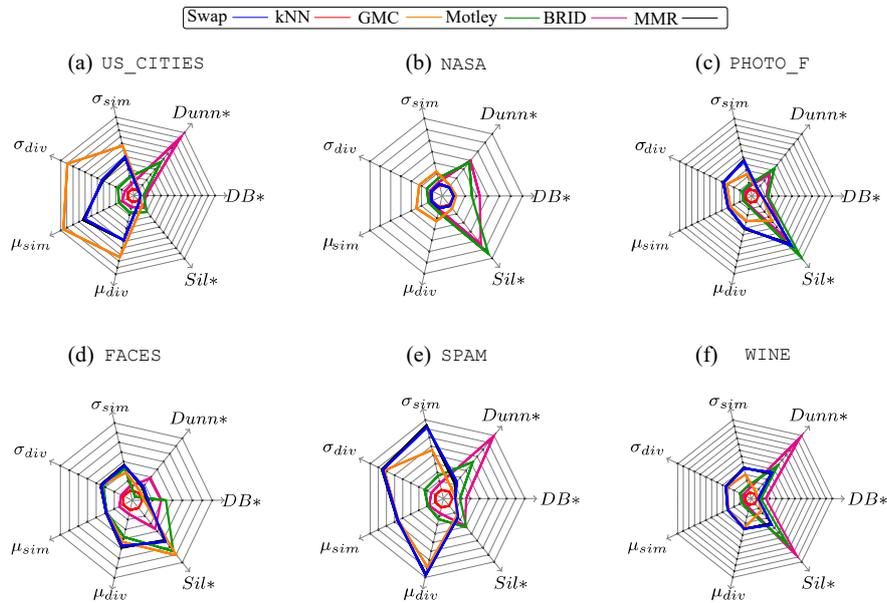

Fig. 7. Visualization of DFM outputs by parallel coordinates maps.

Table II. Top-3 diversity methods according to DFM entries and Median-Rank.

Top-k	US_CITIES	NASA	PHOTO_F	FACES	SPAM	WINE
#1	GMC	GMC	MMR	MMR	MMR	MMR
#2	Swap	Motley	Swap	Swap	Swap	Swap
#3	MMR	BRID	GMC	Motley	GMC	GMC
#4	Motley	MMR	Motley	GMC	Motley	Motley
#5	BRID	Swap	BRID	BRID	BRID	BRID
#6	k-NN	k-NN	k-NN	k-NN	k-NN	k-NN

(1) different datasets may be more efficiently queried by distinct searching algorithms, and

(2) novelty-based methods achieved the best DFM scores in 04 out of 06 comparisons.

Ranking aggregation with Median-Rank. Another flexible consolidation of DFM metrics into a discrete index value can be achieved by *ranking aggregation*. In particular, the Median-Rank strategy [Fagin et al. 2003] provides the fastest method to combine the multidimensional DFM entries into a single index by assuming each dimension as equally as relevant. Accordingly, we used Median-Rank for browsing through individual DFM dimensions to spot the most frequent high-ranked result diversification methods. We also model each DFM dimension as one single and weightless ranking whose positions are determined by the normalized scores achieved by the median of every individual quality metric. Table II shows the consolidated rankings for the competing algorithms according to Median-Rank applied on DFM measures.

The ranking aggregation approach also indicates distinct algorithms may be more efficient in the searching of particular datasets. For instance, the top-3 ranking highlights both GMC and Swap algorithms were suitable for searching query sets US_CITIES WINE, and SPAM, while Motley and BRID were appropriate for querying sets NASA. Those findings are slightly different from those observed in the parallel coordinates maps because the Median-Rank fetches the most frequent high-ranked algorithms rather than those with the largest area differences. Accordingly, such results pinpoint that consolidated DFM values enable the softening of individual biases by following ranking aggregation methods, which can be set towards frequency or area. A final observation is that the result diversification measured by DFM and consolidated with ranking aggregation can also be paired with k-NN queries, whose contrast provides a comparison baseline – See Table II. In summary, the DFM outputs consolidated by the Median-Rank approach indicate:

(1) a set of candidates (top-k) may be used to spot suitable algorithms rather than using a single outcome as the best and isolated result diversification method, and

(2) ranking-oriented approaches can be parameterized to balance DFM dimensions, which renders ranking aggregation methods more flexible than using only parallel coordinates maps.

5. CONCLUSIONS AND FUTURE WORK

This manuscript has discussed diversity searching algorithms regarding the perspective of quality metrics for result diversification. We observed individual indexes tend to favor particular groups of algorithms, which can be softened through the use of multiple metrics. Accordingly, we discussed the implementation of a multidimensional metric model, coined DFM, and its consolidation by means of parallel coordinates maps and ranking aggregation. Experimental findings indicated DFM enable spotting (a set of) suitable result diversification methods according to the characteristics of each dataset. As future work, we will pursue the extension of DFM for measuring the quality of diversified similarity searching in deployment environments of content-based image retrieval applications.

REFERENCES

AGGARWAL, C. C. *Data Mining: The Textbook*. Springer, 2015.

Agrawal, R., Gollapudi, S., Halverson, A., and Ieong, S. Diversifying search results. In *ICWSDM*. pp. 5–14, 2009.

Carbonell, J. and Goldstein, J. The use of MMR, diversity-based reranking for reordering documents and producing summaries. *ACM SIGIR* 1 (1): 335–336, 1998.

Chávez, E., Navarro, G., Baeza-Yates, R., and Marroquín, J. L. Searching in metric spaces. In *Computing Surveys*. Vol. 33. ACM, pp. 273–321, 2001.

Chen, L., Gao, Y., Zheng, B., Jensen, C. S., Yang, H., and Yang, K. Pivot-based metric indexing. *PVLDB* 10 (10), 2017.

Ciaccia, P. and Martinenghi, D. Reconciling skyline and ranking queries. *PVLDB* 10 (11): 1454–1465, 2017.

Drosou, M., Jagadish, H., Pitoura, E., and Stoyanovich, J. Diversity in big data: A review. *Big data* 5 (2): 73–84, 2017.

Drosou, M. and Pitoura, E. Poikilo: a tool for evaluating the results of diversification models and algorithms. *PVLDB* 6 (12): 1246–1249, 2013.

Fagin, R., Kumar, R., and Sivakumar, D. Efficient similarity search and classification via rank aggregation. In *ACM SIGMOD*. pp. 301–312, 2003.

Hetland, M. The Basic Principles of Metric Indexing. In *Swarm Intellingence for Multi-objective Problems in Data Mining*. Springer, pp. 199–232, 2009.

Hetland, M. L. Optimal Metric Search Is Equivalent to the Minimum Dominating Set Problem. In *SISAP*. Springer, pp. 111–125, 2020.

Hjaltason, G. and Samet, H. Index-driven similarity search in metric spaces. *TODS* 28 (4): 517–580, 2003.

Inselberg, A. and Dimsdale, B. Parallel coordinates: a tool for visualizing multi-dimensional geometry. In *Conf. Vis.* IEEE, pp. 361–378, 1990.

Jain, A., Sarda, P., and Haritsa, J. R. Providing diversity in k-nearest neighbor query results. In *PAKDD*. Springer, pp. 404–413, 2004.

Jasbick, D., Santos, L., de Oliveira, D., and Bedo, M. Some Branches May Bear Rotten Fruits: Diversity Browsing VP-Trees. In *SISAP*. Springer, pp. 140–154, 2020.

Lopes, C., Jasbick, D., Bedo, M., and Santos, L. Quality metrics for diversified similarity searching: What they stand for? . In *Simpósio Brasileiro de Bancos de Dados*. SBC, pp. 1–12, 2020.

Pestov, V. Is the k-nn classifier in high dimensions affected by the curse of dimensionality? *Computers & Mathematics with Applications* 65 (10): 1427–1437, 2013.

Polinsky, A., Feinstein, R., Shi, S., and Kuki, A. Librain: Software for automated design of exploratory and targeted combinatorial libraries. *Molecular diversity and combinatorial chemistry: Libraries and drug discovery* vol. 996, pp. 219–232, 1996.

Pouyanfar, S., Yang, Y., Chen, S.-C., Shyu, M.-L., and Iyengar, S. Multimedia big data analytics: A survey. *ACM CSUR* 51 (1): 1–34, 2018.

Rousseeuw, P. J. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* vol. 20, pp. 53–65, 1987.

Santos, L., Blanco, G., Oliveira, D., Traina, A., Traina Jr, C., and Bedo, M. Exploring Diversified Similarity with Kundaha. In *ACM CIKM*. pp. 1903–1906, 2018.

Santos, L., Dias, R. L., Ferreira, M. R., Ribeiro, M. X., Traina, A. J., and Traina Jr, C. Have you met VikS?: A novel framework for visual diversity search analysis. *SBBD Demos*, 2014.

Santos, L., Oliveira, W., Ferreira, M., Cordeiro, R., Traina, A., and Traina Jr, C. Evaluating the diversification of similarity query results. *Journal of Information and Data Management* 4 (3): 188–188, 2013.

Santos, L., Oliveira, W., Ferreira, M., Traina, A., and Traina Jr, C. Parameter-free and domain-independent similarity search with diversity. In *SSDBM*. pp. 1–12, 2013.

Smyth, B. and McClave, P. Similarity vs. diversity. *PICCR* 1 (1): 347–361, 2001.

Vieira, M., Razente, H., Barioni, M., Hadjieleftheriou, M., Srivastava, D., Traina Jr., C., and Tsotras, V. On query result diversification. In *ICDE*. pp. 1163–1174, 2011.

Vincent, P., Larochelle, H., Bengio, Y., and Manzagol, P.-A. Extracting and composing robust features with denoising autoencoders. In *ICML*. pp. 1096–1103, 2008.

Yu, C., Lakshmanan, L. V., and Amer-Yahia, S. Recommendation diversification using explanations. In *ICDE*. IEEE, pp. 1299–1302, 2009.

Zezula, P., Amato, G., Dohnal, V., and Batko, M. *Similarity Search: The Metric Space Approach*. Vol. 32. Springer, 2010.

Zheng, K., Wang, H., Qi, Z., Li, J., and Gao, H. A survey of query result diversification. *Knowledge and Information Systems* 51 (1): 1–36, 2017.