The article discusses two common paradigms for identifying preference records in a multi-objective setting: dominance and utility function. However, each of them has their own drawbacks, and the authors propose three requirements for practical decision support: personalization, controllable output size, and flexibility in preference specification. The authors then combine elements from both paradigms to propose two new operators, ORD and ORU, and evaluate their performance against previous work.

This text discusses the challenges of decision-making in a world where people have access to a plethora of options and proposes a solution that combines two paradigms, dominance-based and ranking by utility, to address the drawbacks of each approach. The dominance-based paradigm considers a record to be of interest if it is not dominated by any other record. On the other hand, ranking by utility associates a score with each record based on its attributes and a user-specific function. The proposed solution combines these two paradigms to create operators that satisfy the requirements of being output-size specified, personalized, and flexible. Personalization is achieved using linear scoring, but the input preferences are treated as estimates, which allows for some flexibility in the specified preferences. The proposed solution gradually expands preferences to include alternative preferences while strictly controlling output size, which is crucial for design considerations, such as display size, device capabilities, and connection speed. The text discusses existing literature on skyline and regret-minimizing sets, which attempt to produce competitive or representative skyline records in a general sense but lack personalization. The proposed solution attempts to address the challenges of decision-making in a world with numerous options while combining the strengths of both paradigms and avoiding their drawbacks.

The passage discusses various approaches to solving the top-k query problem with personalization, including skyline-based techniques, regret-minimizing sets (RMS), interactive regret minimization (IRM), and fixed-region techniques.

Skyline-based techniques involve selecting representative records from the skyline that minimize a distance metric or maximize some objective function. Some techniques consider subspace skylines or attribute importance, while others consider user-specific ratings. However, most skyline-based techniques do not take into account a user's personal preferences.

Regret-minimizing sets (RMS) aim to produce an 𝑚-sized subset 𝑆 ⊂ 𝐷 that minimizes the maximum regret ratio for any possible user. There are various variants of RMS, including 𝑘-RMS and average regret ratio, but they are not concerned with personalization.

Interactive regret minimization (IRM) involves the user in the search process, presenting them with a number of records and asking them to choose the best. As the user provides feedback, IRM learns their preference vector and eventually identifies the one record with maximum utility. IRM assumes a different query processing model and requires active user involvement.

Fixed-region techniques, such as 𝑅-dominance and 𝑅-skyline, define a convex preference polytope 𝑅 and select records that are not 𝑅-dominated by any other. These techniques can be integrated into skyline algorithms or used to identify potentially optimal records.

Overall, the passage highlights the strengths and weaknesses of various techniques for solving the top-k query problem with personalization, with each technique having its own set of assumptions and limitations.

This is a research paper on preference-based record shortlisting, a multi-objective querying technique. The paper proposes two operators, ORD and ORU, that require no precomputation other than a general-purpose spatial index on the dataset. The operators utilize a preference vector to calculate the utility score of records and select records that are 𝜌-dominated by fewer than 𝑘 others, for the minimum 𝜌 that produces exactly 𝑚 records in the output.

The paper also defines the preference domain as the unit (𝑑 − 1)-simplex in a space whose 𝑑 axes correspond to the 𝑤𝑖 values. For 𝑑 = 3, the preference domain is an equilateral triangle, and for 𝑑 = 4, the preference domain is a tetrahedron. The paper focuses on low-dimensional settings since multi-objective querying generally loses its meaning in high dimensions. The paper also notes that although it positions its work within preference-based record shortlisting for a human user, its techniques apply to general multi-objective scenarios where the suitability of available options is defined by a linear function over the options’ attributes.

The text describes two operators for the 𝑘-skyband algorithm: ORD (Optimized Range-Diversity) and ORU (Optimized Range-Utility).

ORD aims to find a diverse set of 𝑚 records from the 𝑘-skyband that maximize the distance between them, using a progressive 𝑘-skyband retrieval process. This process fetches the 𝑘-skyband members one by one in decreasing score order for 𝑤, and it stops when the candidate set reaches size (𝑚 + 1). Then, the candidate with the largest inflection radius is discarded, and the process continues fetching 𝜌¯-skyband members (in decreasing order of score for 𝑤) until the candidate set is finalized. The finalized candidate set corresponds to the 𝜌-skyband for 𝜌 equal to the maximum inflection radius across its members.

ORU, on the other hand, aims to find a set of 𝑚 records from the top-𝑘 for at least one preference vector within a certain range of a seed 𝑤. This range is the minimum 𝜌 that produces exactly 𝑚 records. ORU achieves this by using a modified version of the BBS algorithm that visits index nodes and records in decreasing order of (upper bound of) score for 𝑤, using a max-heap. The algorithm performs 𝜌-dominance tests instead of regular dominance tests after the (𝑚 + 1)-th record is fetched. The operator stops when it reaches 𝜌, which produces exactly 𝑚 records.

Both operators are efficient solutions that address several performance issues, such as avoiding computing the entire 𝑘-skyband in the beginning of the process and limiting the number of considered candidates to as tight a superset of the output as possible.

This section describes the fundamentals of the ORU methodology, which uses the convex hull to perform preference queries on a dataset. The convex hull of a dataset is the smallest convex polytope that encloses all its records, and it comprises facets, each defined by d extreme vertices (records) in general position. A vector is normal to a hyper-plane when its direction is perpendicular to the hyper-plane. The norm of a facet on the hull is the normal vector to that facet whose sum of coordinates is 1, and is directed towards the exterior of the hull. The top record for a preference vector v is the one met first by a hyper-plane normal to v that sweeps the data space from the top corner to the origin. The upper hull is the part that corresponds to facets with non-negative norms. Given a preference vector v whose top record in layer L\_i is r, if we start shifting v towards any direction in the preference domain, the first record in L\_i to outscore r is always in A(r), i.e., among the records adjacent to r. The adjacent set A(r) denotes the records adjacent to r. Lemma 1 states that each of the records in A(r) is the first outscoring record for some shifting direction of v.

This section discusses an algorithmic basis for processing ordered range queries (ORU) and proves a theorem to support it. The algorithm processes the top-k results for any possible preference vector within radius 𝜌 from the seed w, to form the ORU output. The section provides a detailed description of how to determine the top-2nd record for any possible preference vector and partition the top regions based on a given preference vector. It also presents Theorem 1, which generalizes the observation that the top-2nd record anywhere in a preference region must be in the union of two sets: (i) the adjacent records to any member of the known top-i result in its respective layer, and (ii) the records in the (t+1)-th layer whose top-region overlaps the preference region. The section also provides an example to illustrate how the algorithm processes the top-k results for a given seed w and minimum radius 𝜌.

The text discusses a methodology for selecting the top-k results in a multi-dimensional dataset, called the ORU (Order-Respecting Utility) operator. ORU operator follows a divide-and-conquer approach, where a tree of preference regions is created for each point in the dataset. The preference regions are ordered based on their distance from the query point, and the regions are explored in this order until the top-k results are obtained. The text also describes an incremental 𝜌-skyband module that is required for the ORU operator. The module incrementally retrieves the 𝑘-skyband members without computing the entire 𝑘-skyband.

The section describes experiments carried out using both real and synthetic datasets. The authors use four datasets named HOTEL, HOUSE, NBA, and ANTI, COR, IND. They used Table 2 to list the problem parameters and their tested and default values. In each experiment, one parameter was varied while others were fixed to their defaults. The qualitative results were used to differentiate the operators from previous ones. A case study was performed on the NBA 2018-19 season statistics for 708 players. They used TripAdvisor data and reviews to get weight vectors for 137,563 users. The fixed-region methods required a preference polytope to be specified as input, but it was not feasible to estimate the size of the polytope required to produce m records.

The text describes the performance evaluation of two algorithms - ORD and ORU. For ORD, two comparison algorithms are used - a fixed-region R-skyband technique and a baseline ORD-BSL. The results demonstrate that ORD is significantly faster than both versions of RSB and ORD-BSL, even for high dimensional data. The superiority of ORD is established based on its scalability, ability to handle large datasets, and superior performance in terms of running time. For ORU, a fixed-region JAA algorithm and a baseline ORU-BSL are used for comparison. ORU outperforms ORU-BSL by 2 to 4 orders of magnitude and JAA-10% and JAA-5% by 12 to 134 times. ORU is also scalable and can handle large datasets. However, it is more complex than ORD due to the nature of its definition. The results demonstrate the vital role of gradual expansion for ORU. Finally, the running time of both algorithms is evaluated for different synthetic and real data distributions.

The paper discusses the weaknesses of standard skyline and top-k queries and proposes two new operators (ORD and ORU) that satisfy the requirements for practical decision support in multi-objective settings. ORU requires parallelization to achieve sub-second responses, and materializing the Onion technique or caching and reusing partial upper hulls can improve performance. The paper suggests future work to explore algorithmic redesign for order-insensitivity or to switch to ORD processing for small regions under Case 1. The experiments demonstrate practical and scalable performance. Future work could explore the proposed directions or apply ORD/ORU to highly skewed or sparse datasets in higher dimensions.