

**Space Filling Approach for Distributed Processing of Top-k Dominating Queries**

**Abstract:**

A top-k dominating query returns *k* data objects that dominate the highest number of data objects in a given dataset. This query provides us with a set of intuitively preferred data, thus can support a wide variety of multi-criteria decision-making applications, e.g., e-commerce and web search. Due to the growth of data centers and cloud computing infrastructures, the above applications are increasingly being operated in distributed environments. These motivate us to address the problem of distributed top-k dominating query processing. We propose an efficient decentralized algorithm that exploits virtual points and returns the exact answer. The virtual points are utilized to focus on the data space to be preferentially searched and also to limit the search space to prune unnecessary computation and data forwarding. We also propose two other algorithms, which return an approximate answer set while further reducing query processing time. Extensive experiments on both real and synthetic data demonstrate the efficiency and scalability of our algorithms.

**Existing System:**

Top-k dominating queries, which have attracted much research attention recently have the advantages of both top-k and skyline queries, without their drawbacks. That is, top-k dominating queries hold the following advantages: (i) the result size is controllable, (ii) the result is scale invariant, and (iii) users are not required to specify scoring functions. Given a *d*-dimensional dataset and *k*, top-k dominating queries retrieve the *k* data objects that dominate the highest

number of data objects. Fig. 1 shows a concrete example of a top-k dominating query, which illustrates our assumed distributed environment that consists of a coordinator and multiple distributed sites.

**Proposed system:**

To accelerate the performance of top-k dominating query processing, we need to limit the search space and reduce unnecessary computation and data forwarding. This is particularly important for processing top-k dominating queries efficiently, because computing dominance-relationships is CPU-intensive.

Thus, an important key to boosting query processing efficiency is to reduce the number of data objects (i) that are processed by the coordinator and (ii) whose scores are computed. Achieving this idea, however, faces a challenge in distributed environments; though we do not hold a global view in advance, how do we know the data spaces where the top-k dominating data do not exist? As the state-of-the-art [3] retrieves the top-k dominating data *from the entire data space*, this challenge is not trivial. We make the following contributions to overcome the challenge.