Motivation

The Problem
- Cloud-based systems are crucial for processing and analyzing large data.
- Similarity Joins (SJ) are key data processing and analysis tools.
- Very little work on Similarity Joins has been done for cloud systems.

Our Contribution
- We propose MRSimJoin, a MapReduce-based algorithm to efficiently solve the SJ problem.
- The algorithm is general enough to be used with data that lies in any metric space.
- We have implemented MRSimJoin in Hadoop.

MRSimJoin Round

- MRSimJoin iteratively partitions the data into smaller partitions until each partition is small enough to be efficiently processed by a single-node SJ routine.
- The process is divided into a sequence of rounds.
- The initial round partitions the input data while any subsequent round partitions a previously generated partition.

Multiple Rounds

- Each round corresponds to a MapReduce job.
- The output of a round includes:
  1. Result links for the small partitions that were processed in a single-node.
  2. Intermediate data for partitions that require further partitioning.

Partitioning in a MRSimJoin Round

- Data partitioning is performed using a set of K pivots (conceptually similar to QuickJoin), which are a subset of the records to be partitioned.
- The process generates two types of partitions: base partitions and window-pair partitions.
  1. A base partition contains all the records that are closer to a given pivot than to any other pivot.
  2. A window-pair partition contains the records in the boundary between two base partitions.

Performance Evaluation

Tests run over 5 million (SF1) 9D records.

Increasing Scale Factor

- ColorData, 9D, Eps: 1.5%

- Output Size
- Execution Time (seconds)

Increasing Epsilon

- ColorData, 9D, SF: 1