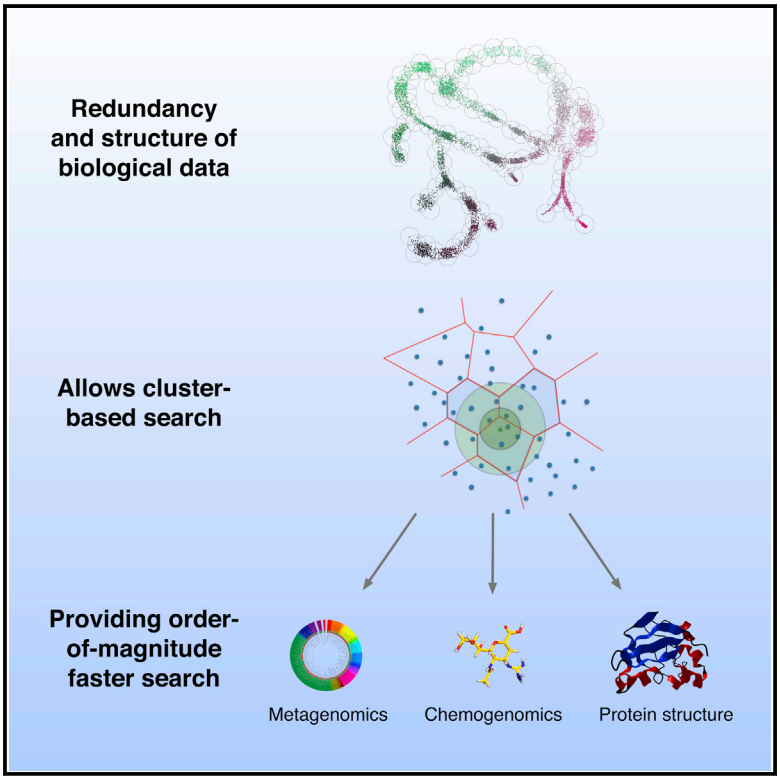
Article

Cell Systems

Entropy-Scaling Search of Massive Biological Data

Graphical Abstract



Authors

Y. William Yu, Noah M. Daniels, David Christian Danko, Bonnie Berger

Correspondence

bab@mit.edu

In Brief

Yu, Daniels et al. describe a general framework for efficiently searching massive datasets having certain properties common in biology.

Highlights

- We describe entropy-scaling search for finding approximate matches in a database
- Search complexity is bounded in time and space by the entropy of the database
- We make tools that enable search of three largely intractable real-world databases
- The tools dramatically accelerate metagenomic, chemical, and protein structure search

Yu et al., 2015, Cell Systems *1*, 130–140
August 26, 2015
http://dx.doi.org/10.1016/j.cels.2015.08.004

CellPress

CellPress

Cell Systems
Article

Entropy-Scaling Search of Massive Biological Data

Y. William Yu,[1,2,3] Noah M. Daniels,[1,2,3] David Christian Danko,[2] and Bonnie Berger[1,2,*]
[1]Department of Mathematics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA
[2]Computer Science and Artificial Intelligence Laboratory, Massachusetts Institute of Technology, Cambridge, MA 02139, USA
[3]Co-first author
*Correspondence: bab@mit.edu
http://dx.doi.org/10.1016/j.cels.2015.08.004

SUMMARY

Many datasets exhibit a well-defined structure that can be exploited to design faster search tools, but it is not always clear when such acceleration is possible. Here, we introduce a framework for similarity search based on characterizing a dataset's entropy and fractal dimension. We prove that searching scales in time with metric entropy (number of covering hyperspheres), if the fractal dimension of the dataset is low, and scales in space with the sum of metric entropy and information-theoretic entropy (randomness of the data). Using these ideas, we present accelerated versions of standard tools, with no loss in specificity and little loss in sensitivity, for use in three domains—high-throughput drug screening (Ammolite, 150× speedup), metagenomics (MICA, 3.5× speedup of DIAMOND [3,700× BLASTX]), and protein structure search (esFragBag, 10× speedup of FragBag). Our framework can be used to achieve "'compressive omics," and the general theory can be readily applied to data science problems outside of biology (source code: http://gems.csail.mit.edu).

INTRODUCTION

Throughout all areas of data science, researchers are confronted with increasingly large volumes of data. In many fields, this increase is exponential in nature, outpacing Moore's and Kryder's laws on the respective doublings of transistors on a chip and long-term data storage density (Kahn, 2011). As such, the challenges posed by the massive influx of data cannot be solved by waiting for faster and larger capacity computers but, instead, require the development of data structures and representations that exploit the structure of the dataset.

Here, we focus on similarity search, where the task at hand is to find all entries in a database that are "similar," or approximate matches, to a query item. Similarity search is a fundamental operation in data science and lies at the heart of many other problems, much like how sorting is a primitive operation in computer science. Traditionally, approximate matching has been studied primarily in the context of strings under edit distance metrics (Box 1) (e.g., for a spell-checker to suggest the most similar words to a misspelled word) (Ukkonen, 1985). Several approaches, such as the compressed suffix array and the FM-index (Grossi and Vitter, 2005; Ferragina and Manzini, 2000), have been developed to accelerate approximate matching of strings. However, it has been demonstrated that similarity search is also important in problem domains where biological data are not necessarily represented as strings, including computational screening of chemical graphs (Schaeffer, 2007) and searching protein structures (Budowski-Tal et al., 2010). Therefore, approaches that apply to more general conditions are needed.

As available data grow exponentially (Berger et al., 2013; Yu et al., 2015) (e.g., genomic data in Figure S1), algorithms that scale linearly (Box 1) with the amount of data no longer suffice. The primary ways in which the literature addresses this problem—locality sensitive hashing (Indyk and Motwani, 1998), vector approximation (Ferhatosmanoglu et al., 2000), and space partitioning (Weber et al., 1998)—involve the construction of data structures that support more efficient search operations. However, we note that, as biological data increase, not only does the redundancy present in the data also increase (Loh et al., 2012) but also internal structure (such as the fact that not all conceivable configurations, e.g., all possible protein sequences, actually exist) also becomes apparent. Existing general-purpose methods such as compressed data structures (Grossi and Vitter, 2005) do not explicitly exploit the particular properties of biological data to accelerate search (see the Theory section in the Supplemental Experimental Procedures).

Previously, our group demonstrated how redundancy in genomic data could be used to accelerate local sequence alignment. Using an approach that we called "compressive genomics," we accelerated BLAST and BLAT (Kent, 2002) by taking advantage of high redundancy between related genomes using link pointers and edit scripts to a database of unique sequences (Loh et al., 2012). We have used similar strategies to obtain equally encouraging results for local alignment in proteomics (Daniels et al., 2013). Empirically, this compressive acceleration appears to scale almost linearly in the entropy of the database, often resulting in orders of magnitude better performance; however, these previous studies neither proved complexity bounds nor established a theory to explain these empirical speedups.

Here, we generalize and formalize this approach by introducing a framework for similarity search of omics data. We prove that search performance primarily depends on a measure of the novelty of new data, also known as entropy. This framework, which we call entropy-scaling search, supports the creation of a data structure that provably scales linearly in both time and space with the entropy of the database, and thus sublinearly with the entire database.