Resumen “Fundamental Data Structures for Managing Large Datasets”
Gonzalo Navarro

The amount of digital information available worldwide has been growing at a daunting pace in almost every domain, together with the need of performing more and more sophisticated queries in order to extract useful information from an ever-growing mass of data. An increasingly popular decision to achieve high query throughputs is to maintain all the data in main memory, if necessary aggregating it over clusters of machines. A limited main memory is also the choice in mobile computing, whose increasing popularity is beyond any doubt.

Compact data structures are an excellent tool to boost the performance of main-memory data structures. They allow handling larger datasets within the same space, reducing the I/O traffic, and on aggregated clusters, reducing the total number of computers required, which impacts on communication, hardware, and energy costs.

From a research point of view, compact data structures combine challenges from two different areas: algorithms and data structures on the one hand, and compression and information theory on the other. After a slow start over the 90’s, they have grown and pervaded many application areas, where they have achieved important theoretical and practical results. In this project we plan to research on fundamental aspects of compact data structures, with particular focus on three application areas where their impact is expected be relevant in the next years: Information Retrieval. We encompass here various related areas: (1) Natural language text collections; (2) general sequence collections; (3) XML collections; (4) RDF databases. In (1), where inverted indexes are the dominant solution, we plan to improve their representation so that the ranked retrieval algorithms can be carried out faster and/or using less space. In (2), where most of the research has been on (compressed or not) text indexes able to find all the occurrences of pattern strings, we plan to develop improved document retrieval techniques, which give functionalities similar to the ranked retrieval that is common on (1), and possibly also apply those results in the scope of (1). In (3), we plan to enhance the existing results with support for more complex queries, including the important XPath semijoin operation, and possibly ranked retrieval. In (4), we will pursue on novel representations of the set of triples in order to support simple pattern queries and joins, the basics of SPARQL.

Bioinformatics. We plan to work on the efficient representation of sequence collections in order to support the complex searches required in this field. One important focus will be on repetitive collections, which arise when the genomes of many individuals of the same or similar species are sequenced. This type of biological database is likely to sharply grow in the next years. We plan to exploit repetitiveness to reduce space, while at the same time offering new search capabilities related to retrieving relevant units of information more meaningful than just occurrences, in particular the pieces (e.g., genes or genomes) where the patterns occur, or those where the patterns occur most often.

Versioned Collections. There are many applications where the data evolves over time and it is necessary to query about any point in the past, or about a certain time range. These include, for example, versioned document collections (like Wikipedia), software repositories, geographic information systems (GIS) where the data evolves over time, and others. Their storage is challenging. In principle, data structures that exploit repetitiveness are suitable to handle this kind of data, however the nature of the queries might be different, because the versions
may have a linear or tree structure on which new types of queries may be posed. Further, the data itself may be not only text (as in documents) but also trees or graphs (software repositories), grids (GIS), and so on.

We plan to address these issues with a toolbox of structures and techniques we have been developing in recent years: (a) Repetition-aware text indexes, based on Lempel-Ziv or grammar compression, useful to exploit repetitions among biological sequences and among similar document versions, and possibly among trees (XML, software repositories) and grids (GIS); (b) Geometric representations that adapt to the entropy of the data, like $k^2$-trees and wavelet trees, which are useful to represent binary relations (RDF) and geographic information (GIS), for example; (c) Structures handling multi-attribute orderings, like treaps and (again) wavelet trees, which are useful to retrieve prioritized answers from one- and two-dimensional data; this is useful for ranked retrieval in inverted indexes and in general sequence collections, as well as for weighted geometric data.

We plan to work both on fundamental aspects, developing new algorithms and compact data structures, and on their application areas, obtaining theoretical and practical results of both basic and applied nature.