Report of the first year of the project (2017WR7SHH)

"Multicriteria Data Structures and Algorithms: from compressed to learned indexes, and beyond"

The ever-growing need to efficiently store, retrieve and analyze large amounts of data, originated by very different sources, is currently made more complex by the different and, possibly, time-varying requirements posed by users and applications. The last thirty years have thus witnessed an upsurging interest towards the design of algorithms and data structures that could scale with this deluge of data. That has resulted in the development of four main research areas: External memory, Compressed, Cache-oblivious and Streaming. Thanks to those developments, we now have theoretically sound results to cope with Big Data, which are also implemented in widely used software libraries: SDSL-lite, WebGraph, Data Sketches Core library, etc..

An evident limitation of these results is that they offer a collection of different trade-offs in terms of I/Os, working space, time, etc., among which software engineers have (not easily, indeed) to choose the one that best fits the needs of their application. To make things even more difficult, these needs are often not exactly the ones addressed by theory-efficient results, especially since these needs may change with time, data distribution, and users.

To successfully meet these challenges, our PRIN project aims at studying, designing and implementing a new generation of "Multicriteria data structures and algorithms". The "multicriteria" feature refers to the fact that in our framework we seamlessly integrate, via a "principled" optimization approach, modern compressed data structures with new, revolutionary, data structures "learned" from the input data by using proper machine-learning tools. The goal of the optimization is to select, among a family of properly designed data structures, the one that "best fits" the multiple constraints imposed by its context of use, thus eventually "dominating" the multitude of trade-offs currently offered by known solutions.

The ultimate goal of the project is to lay down the theoretical and algorithmic-engineering foundations of this novel research area, which has the potential of providing a new generation of data structures and algorithms supporting innovative data-intensive applications. We plan to study how to optimally combine powerful techniques coming from Algorithmics (Compressed and Cache Oblivious data structures), Information Theory (Data Compression) and Artificial Intelligence (Machine Learning and especially Neural Networks) in a sound mathematical way, in order to design a "family of data structures" from which an application will draw, via proper optimization algorithms, the one which satisfies the multicriteria constraints imposed by the specific context of use. Following this approach, the performance of the selected data structure will be bounded above by the worst-case complexity of the known (classic) data structures and have the potential of significantly outperforming it, thanks to the optimization approach and the integration of novel techniques from ML.

A synopsis of the project is as follows:

(A) Formulate the theoretical foundations of Multicriteria data structures (and their algorithms).

(B) Study and integrate Learned indexes with Compressed and Cache-oblivious data structures in the framework of Multicriteria Indexing and Compression.

(C) Systematically design, engineer, and experiment some Multicriteria data structures on datasets and applications mainly coming from BioInformatics and Web/text search.

Task and deliverables of the first year

We report below the task outline and the deliverables as they were defined in the project submission. For each deliverable, we list the publications and/or the references to the software libraries that we have developed in order to address the issues of the corresponding tasks. Some of these results refer to the activity of the proponents started immediately after the communication by MUR of the acceptance of the project proposal, hence April 2019.

[T1] Classic (Compressed) Data Structures vs Purely Learned Indexes

The goal of this task is to theoretically and experimentally address the question of identifying under which conditions Learned Indexes outperform classic and compressed indexes.

[D1.1] *Characterization of the space-time-accuracy performance of ML models in terms of the distribution of the input data, and their comparison against the known (compressed) data structures.*

To achieve this goal we adopt all the theoretical machinery that is proper of Information Theory, Machine Learning Theory and Algorithmics, looking for the first time at the interplay among those scientific areas within a data-structure perspective by leveraging the vast amount of knowledge that has been developed in Compressed Data Structures, also by the proponents.

- Paolo Ferragina and Giorgio Vinciguerra. *"Learned Data Structures"*. Chapter in *Studies in Computational Intelligence*, vol. 896, Springer, pages 5-41, 2020. ISBN: 978-3-030-43883-8. DOI: 10.1007/978-3-030-43883-8_2**.** *Green open-access: preprint available on the project web page*

  ABSTRACT. Very recently, the unexpected combination of data structures and machine learning has led to the development of a new area of research, called learned data structures.

Their distinguishing trait is the ability to reveal and exploit patterns and trends in the input data for achieving more efficiency in time and space, compared to previously known data structures. The goal of this chapter is to provide the first comprehensive survey of these results and to stimulate further research in this promising area.

- Paolo Ferragina, Fabrizio Lillo, and Giorgio Vinciguerra. Why are learned indexes so effective? In: *Proc. 37th International Conference on Machine Learning (ICML)*. PMLR vol 119, 2020.

  ABSTRACT. A recent trend in algorithm design consists of augmenting classic data structures with machine learning models, which are better suited to reveal and exploit patterns and trends in the input data so as to achieve outstanding practical improvements in space occupancy and time efficiency. This is especially known in the context of indexing data structures where, despite few attempts in evaluating their asymptotic efficiency, theoretical results are yet missing in showing that learned indexes are provably better than classic indexes, such as B^+-trees and their variants. In this paper, we present the first mathematically-grounded answer to this open problem. We obtain this result by discovering and exploiting a link between the original problem and a mean exit time problem over a proper stochastic process which, we show, is related to the space and time occupancy of those learned indexes. Our general result is then specialised to five well-known distributions: Uniform, Lognormal, Pareto, Exponential, and Gamma; and it is corroborated in precision and robustness by a large set of experiments.

- Domenico Amato, Giosuè Lo Bosco, Raffaele Giancarlo. Learning from Data to Speed-up Sorted Table Search Procedures: Methodology and Practical Guidelines (2020).
  Available at arXiv:2007.10237.

  ABSTRACT. Sorted Table Search Procedures are the quintessential query-answering tool, with widespread usage that now includes also Web Applications, e.g, Search Engines (Google Chrome) and ad Bidding Systems (AppNexus). Speeding them up, at very little cost in space, is still a quite significant achievement. Here we study to what extent Machine Learning Techniques can contribute to obtain such a speed-up via a systematic experimental comparison of known efficient implementations of Sorted Table Search procedures, with different Data Layouts, and their Learned counterparts developed here. We characterize the scenarios in which those latter can be profitably used with respect to the former, accounting for both CPU and GPU computing. Our approach contributes also to the study of Learned Data Structures, a recent proposal to improve the time/space performance of fundamental Data Structures, e.g., B-trees, Hash Tables, Bloom Filters. Indeed, we also formalize an Algorithmic Paradigm of Learned Dichotomic Sorted Table Search procedures that naturally complements the Learned one proposed here and that characterizes most of the known Sorted Table Search Procedures as having a learning phase that approximates Simple Linear Regression.

[D1.2] *A collection of known and possibly new implementations of ML-based and compressed data structures, to be used in the next tasks T3 and T4 as "building blocks" for our multi-criteria framework.*

The list of software tools is based on the theoretical characterization offered by D1.1 and on the experiments performed on the well-tuned and robust libraries of compressed data structures, such as SDSL-lite, and of ML models, e.g. WEKA, Keras and OpenNN. Datasets will be synthetic and real, the latter borrowed from Web/text collections and BioInformatics.

- The code to reproduce the experiments of the ICML 2020 paper above is available at https://github.com/gvinciguerra/Learned-indexes-effectiveness.

- The code to experiment with and design novel Learned Sorted Table Search procedures of *arXiv:2007.10237* paper is available at https://github.com/raffaelegiancarlo/A-Software-LIbrary-to-Speed-up-Sorted-Table-Search-Procedures-via-Learning-from-Data.

[T2] Compressed ML models

At the submission of the project there were only a handful of papers dealing on a solid theoretic basis with the goal of obtaining space-conscious ML models. The (simple) existing approaches tend to a deterioration in the model accuracy, and/or to an increase of the training and execution time. In this task, we aim at overcoming such limitations.

[D2.1] *Development of compressed ML models.*

The goal is to design novel succinct ML models by investigating the trade-off between compressed space, (de)compressed time and their prediction accuracy.

- Raffaele Giancarlo, Simona E. Rombo, Filippo Utro: DNA combinatorial messages and Epigenomics - The case of chromatin organization and nucleosome occupancy in eukaryotic genomes. *Theoretical Computer Science,* 792: 117-130, 2019.
  DOI: 10.1016/j.tcs.2018.06.047.

  ABSTRACT. This contribution highlights how information measures such as empirical entropy and linguistic complexity can actually classify with great reliability nucleosome positions in Eukaryotic Genomes. Of interest here is an information theoretic model that uses only one register per-prediction for this type of task, as opposed to the large memory of ML models (exponential in some learning parameters). Our model is two orders of magnitude faster than existing ML models granting the same accuracy.

- Antonio Carta, Alessandro Sperduti, Davide Bacciu: Incremental Training of a Recurrent Neural Network Exploiting a Multi-Scale Dynamic Memory, *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD), 2020 (to appear)*.

  ABSTRACT. The effectiveness of recurrent neural networks can be largely influenced by their ability to store into their dynamical memory information extracted from input sequences at different frequencies and timescales. Such a feature can be introduced into a neural architecture by an appropriate modularization of the dynamic memory. In this paper we propose a novel incrementally trained recurrent architecture targeting explicitly multi-scale learning. First, we show how to extend the architecture of a simple RNN by separating its hidden state into different modules, each subsampling the network hidden activations at different frequencies. Then, we discuss a training algorithm where new modules are iteratively added to the model to learn progressively longer dependencies. Each new module works at a slower frequency than the previous ones and it is initialized to encode the subsampled sequence of hidden activations. Experimental results on synthetic and real-world datasets on speech recognition and handwritten characters show that the modular architecture and the incremental training algorithm improve the ability of recurrent neural networks to capture long-term dependencies.

- Domenico Amato, Giosuè Lo Bosco, and Riccardo Rizzo: CORENup - a combination of convolutional and recurrent deep neural networks for nucleosome positioning identification, *BMC Bioinformatics*, 2020 *(to appear)*.

  ABSTRACT. We propose CORENup, a deep learning model for nucleosome identification. CORENup processes a DNA sequence as input using one-hot representation and combines in a parallel fashion a fully convolutional neural network and a recurrent layer. These two parallel levels are devoted to catching both *non periodic* and *periodic* DNA string features. A dense layer is devoted to their combination to give a final classification. CORENup is a state of the art methodology for nucleosome positioning identification based on a Deep Neural Network architecture. The comparisons have been carried out using two groups of datasets, currently adopted by the best performing methods, and CORENup has shown top performance both in terms of classification metrics and elapsed computation time.

- Giosuè Cataldo Marinò, Gregorio Ghidoli, Marco Frasca, Dario Malchiodi. Compression strategies and space-conscious representations for deep neural networks. *Available at arXiv:2007.07967* (2020) *and submitted to the International Conference on Pattern Recognition, 2020.*

  ABSTRACT. Recent advances in deep learning have made available large, powerful convolutional neural networks (CNNs) with state-of-the-art performance in several real-world applications. Unfortunately, these large-sized models have millions of parameters, thus they might not be deployable on resource-limited platforms (e.g. where RAM is limited). Compression of CNNs thereby becomes a critical problem to achieve memory-efficient and

possibly computationally faster model representations. We have investigated the impact of lossy compression techniques for CNNs by weight pruning and quantization, and lossless weight matrix representations based on entropy coding. We leveraged the synergy of network compression and sparse network representations to drastically reduce the model occupancy. Experiments on four benchmark datasets for classification and regression problems, comparing the model accuracy, the space occupancy and the query time, have shown compression rates up to 165 times of the original uncompressed model, and preserved or improved performance. The query process instead is slightly slowered, since actually it does not exploit computer parallelism. To overcome this limitation, we are studying as ongoing development a parallel version of the dot product designed for our CNN compressed representation.

- Lavinia Egidi, Felipe Louza, Giovanni Manzini: Space-Efficient Merging of Succinct de Bruijn Graphs. *Proceedings 26th International Symposium on String Processing and Information Retrieval (SPIRE),* Lecture Notes in Computer Science 11811, *October 2019.*

  ABSTRACT. A well established approach in Bioinformatics is the use of Compressed Suffix Tree and Suffix Arrays as tools for the design of compressed ML models for Biosequences. To further reduce the model size one can replace them with de Bruijn graphs which have been already used as a lossy, and therefore much more compact, alternative for suffix trees/arrays. With the objective of pursuing that avenue, in this paper we show an efficient algorithm for building large succinct representations of de Bruijn graphs by merging the succinct representation of smaller graphs. The proposed solution is the merging algorithm with the smallest memory footprint and is the only one able to compute succinct representation of Variable Order de Bruijn graphs.

- Travis Gagie, Tomohiro I, Giovanni Manzini, Gonzalo Navarro, Hiroshi Sakamoto, and Yoshimasa Takabatake, Rpair: Rescaling REPair with Rsync. *Available at arXiv:1906.00809*. *Proceedings 26th International Symposium on String Processing and Information Retrieval (SPIRE),* Lecture Notes in Computer Science 11811, *October 2019.*

  ABSTRACT. Data compression is a powerful tool for managing massive but repetitive datasets, especially schemes such as grammar-based compression that support computation over the data without decompressing it. In the best case such a scheme takes a dataset so big that it must be stored on disk and shrinks it enough that it can be stored and processed in internal memory. Even then, however, the scheme is essentially useless unless it can be built on the original dataset reasonably quickly while keeping the dataset on disk. In this paper we show how we can preprocess such datasets with context-triggered piecewise hashing such that afterwards we can apply RePair and other grammar-based compressors more easily. We first give our algorithm, then show how a variant of it can be used to approximate the LZ77 parse, then leverage that to prove theoretical bounds on compression, and finally give experimental evidence that our approach is competitive in practice.

- Travis Gagie, Tomohiro I, Giovanni Manzini, Gonzalo Navarro, Hiroshi Sakamoto, Louisa Seelbach Benkner, and Yoshimasa Takabatake: Practical Random Access to SLP-Compressed Texts. *Available at arXiv:1910.07145. Proceedings 27th International Symposium on String Processing and Information Retrieval, (SPIRE 2020), October 2020 (to appear).*

  ABSTRACT. A promising approach for the compression of ML models and data is the use of grammar based compressors, and in particular of straight-line programs (SLPs) which are context-free grammars in Chomsky normal form that each generate exactly one string. Until recently the relatively poor time-space trade-offs during real-life construction of SLPs made them impractical for truly massive datasets. Recently we showed how simple pre-processing can dramatically improve those trade-offs, and in this paper we turn our attention to one of the features that make grammar-based compression so attractive: the possibility of supporting fast random access. We give a new encoding of grammars that is about as small as the practical state of the art but with significantly faster queries.

- Mariella Bonomo, Armando La Placa, Simona E. Rombo: Identifying the k Best Targets for an Advertisement Campaign via Online Social Networks. *Available at arXiv:2008.02108. Proceedings 12th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (KDIR), November 2020 (to appear).*

  ABSTRACT. It is proposed a Multi-criteria Model and Mining approach for the selection of the k best nodes in a social network to involve in an advertisement campaign. The approach uses a succinct representation of node profiles and their proximity. The optimal choice is done by properly choosing along the pareto of the objective function. The method works very well on large graphs.

- Umberto Ferraro Petrillo, Mara Sorella, Giuseppe Cattaneo, Raffaele Giancarlo, Simona E. Rombo: Analyzing big datasets of genomic sequences: fast and scalable collection of *k*-mer statistics. *BMC Bioinformatics* 20-S(4): 138:1-138:14 (2019).

  ABSTRACT. Distributed approaches based on the MapReduce programming paradigm have started to be proposed in the Bioinformatics domain, due to the large amount of data produced by the next-generation sequencing techniques. However, the use of MapReduce and related Big Data technologies and frameworks (e.g., Apache Hadoop and Spark) does not necessarily produce satisfactory results, in terms of both efficiency and effectiveness. We discuss how the development of distributed and Big Data management technologies has affected the analysis of large datasets of biological sequences. Moreover, we show how the choice of different parameter configurations and the careful engineering of the software with respect to the specific framework under consideration may be crucial in order to achieve good performance, especially on very large amounts of data. We choose *k*-mers counting as a case study for our analysis, and Spark as the framework to implement FastKmer, a novel approach for the extraction of *k*-mer statistics from large collections of biological sequences, with arbitrary values of *k*.

- Umberto Ferraro Petrillo, Francesco Palini , Giuseppe Cattaneo, Raffaele Giancarlo, An Extensible, Scalable Spark Platform for Alignment-free Genomic Analysis. *Available at arXiv:2005.00942*.

  ABSTRACT. Alignment-free classification and analysis of genomic sequences, a pillar of Bioinformatics, rely on subword dictionaries built from the set of sequences to process. That is, the dictionary is the model extracted from the data. Current sequencing technology no longer allows sequential or shared memory parallel processing for those tasks. Moreover, the dictionaries are extremely large, even for Hadoop and Spark. The two contributions here provide Spark solutions to Alignment-free classification and analysis of genomic sequences that use carefully designed succinct representations of dictionaries together with a load balancing mechanism that has the effect of allowing the processing of very large datasets. The effectiveness of the proposed solution is demonstrated by providing a novel, and very data intensive, benchmarking of Alignment-free methods, highlighting for the first time limitations regarding their reliability.

[D2.2] *A collection of software prototypes for succinct ML-models.*

We aim at engineering and implementing the algorithms and data structures studied and designed in deliverable D2.1 and experiment them on a few synthetic and real datasets, in order to draw theoretical and practical conclusions to be deployed in the next two tasks.

- The software and datasets for the experiments of the paper published in ECML '20 are available at https://github.com/AntonioCarta/mslmn.

- The source code and benchmark data for the work *arXiv:2007.07967* are available at https://github.com/giosumarin/ICPR2020_sHAM.

- BigRepair: a grammar compressor for huge datasets with many repetitions is available at https://gitlab.com/manzai/bigrepair.

- Source code and benchmark data for SLP compression of ML models are available at https://github.com/itomomoti/ShapedSlp.

- The Spark platforms for Alignment-free Genomic Analysis are available at https://bitbucket.org/maruscia/kmercounting and http://www.statistica.uniroma1.it/users/uferraro/experim/FADE/.

- CORENup software is available at https://github.com/DeepLearningForSequence/CORENup-A-Combination-of-Convolutional-and-Recurrent-Deep-Neural-Networks-for-NucleosomePositioning.

[T3] Multicriteria Data Indexing

The goal of this task is to benefit of the results achieved in tasks T1 and T2, and of some of our previous results in the context of compressed data structures, to achieve the following two deliverables.

[D3.1] *Methodology and framework for the design, analysis and experimentation of multicriteria indexes*.

We wish to deliver Multicriteria Indexes that are optimized to choose, among a set of well-studied and tuned "building blocks" (either ML models or compressed data structures), which one to adopt in the index according to the underlying data distribution and to the time or space bounds imposed by the application.

- Paolo Ferragina and Giorgio Vinciguerra. The PGM-index: a fully-dynamic compressed learned index with provable worst-case bounds. In: *PVLDB 13.8* (2020), pp. 1162–1175. ISSN: 2150-8097.
  DOI: 10.14778/3389133.3389135, *Open access via the VLDB site*

  ABSTRACT. We present the first learned index that supports predecessor, range queries and updates within provably efficient time and space bounds in the worst case. In the (static) context of just predecessor and range queries these bounds turn out to be optimal. We call this learned index the Piecewise Geometric Model index (PGM-index). Its flexible design allows us to introduce three variants which are novel in the context of learned data structures. The first variant of the PGM-index is able to adapt itself to the distribution of the query operations, thus resulting in the first known distribution-aware learned index to date. The second variant exploits the repetitiveness possibly present at the level of the learned models that compose the PGM-index to further compress its succinct space footprint. The third one is a multicriteria variant of the PGM-index that efficiently auto-tunes itself in a few seconds over hundreds of millions of keys to satisfy space-time constraints which evolve over time across users, devices and applications. These theoretical achievements are supported by a large set of experimental results on known datasets which show that the fully-dynamic PGM-index improves the space occupancy of existing traditional and learned indexes by up to three orders of magnitude, while still achieving their same or even better query and update time efficiency. As an example, in the static setting of predecessor and range queries, the PGM-index matches the query performance of a cache-optimised static B+ tree within two orders of magnitude (83×) less space; whereas in the fully-dynamic setting, where insertions and deletions are allowed, the PGM-index improves the query and update time performance of a B+ tree by up to 71% within three orders of magnitude (1140×) less space.

- Raffaele Giancarlo, Giovanni Manzini, Antonio Restivo, Giovanna Rosone, and Marinella Sciortino: A New Class of Searchable and Provably Highly Compressible String Transformations. *Proceedings 30th annual Symposium on Combinatorial Pattern Matching (CPM)*, LIPIcs 128, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, *2019.*
- Raffaele Giancarlo, Giovanni Manzini, Antonio Restivo, Giovanna Rosone, and Marinella Sciortino: The Alternating BWT: An algorithmic perspective. *Theoretical Computer Science, 812: 230–243, 2020.*
  DOI: 10.1016/j.tcs.2019.11.002, *Green open-access: postprint available on the personal web page of the authors*

  ABSTRACT. The Burrows-Wheeler Transform is a string transformation that plays a fundamental role for the design of self-indexing compressed data structures. Over the years, researchers have successfully extended this transformation outside the domains of strings. However, efforts to find non-trivial alternatives to the original Burrows-Wheeler string transformation have met limited success. In the above two papers we introduce and analyse a new family of transformations that have all the virtues of the BWT: they can be computed and inverted in linear time, they produce provably highly compressible strings, and they support linear time pattern search directly on the transformed string. Having at our disposal a wide class of string transformations with the same remarkable properties of the BWT is the first step for the design of multicriteria BWT-based data structures where individual users can select the one more suitable for their task

- Lavinia Egidi, Giovanni Manzini: Lightweight merging of compressed indices based on BWT variants. *Theoretical Computer Science, 812: 214–229 (2020).*
  DOI: 10.1016/j.tcs.2019.11.001, *Green open-access: postprint available on the personal web page of the authors*

  ABSTRACT. One of the criteria for establishing the appropriateness of an index to a particular problem is the possibility of updating it to incorporate new datasets. In the context of compressed indices it is desirable to do the update keeping all the information in compressed form. In this paper we introduce a general technique for merging different families of compressed indices based on the BWT without decompression and using a very small amount of working memory.

- Alan Kuhnle, Taher Mun, Christina Boucher, Travis Gagie, Ben Langmead, Giovanni Manzini: Efficient Construction of a Complete Index for Pan-Genomics Read Alignment. In: Cowen L. (eds), *Research in Computational Molecular Biology (RECOMB), Lecture Notes in Computer Science 11467*, Springer, 2019

- Alan Kuhnle, Taher Mun, Christina Boucher, Travis Gagie, Ben Langmead, Giovanni Manzini: Efficient Construction of a Complete Index for Pan-Genomics Read Alignment. *Journal of Computational Biology 27(4), 500–513 (2020).*
  DOI: 10.1089/cmb.2019.0309, *Open access*

ABSTRACT. A fundamental issue in the practical implementation of any index is the problem is how to construct it for very large datasets. In this paper we introduce a technique called prefix-free parsing and show its effectiveness for building indexes for large collections of similar genomes. We combine this construction technique with a recent compressed data structure for datasets with many repetitions and we show that for collections of human genomes our solution requires only about 2% of the time and 6% of the peak memory usage of traditional compressed data structures. We plan to apply the insight we got on datasets with many repetitions to the development of learned indices for genome collections.

- Mario Randazzo, Simona E. Rombo. A Big Data Approach for Sequences Indexing on the Cloud via Burrows Wheeler Transform. *Available at arXiv:2007.10095*. Proceedings of ECAI Workshop in Artificial Intelligence for Health, Personalized Medicine and Wellbeing, 2020 (to appear).

  ABSTRACT. Indexing sequence data is important in the context of Precision Medicine, where large amounts of ``omics'' data have to be daily collected and analyzed in order to categorize patients and identify the most effective therapies. Here we propose an algorithm for the computation of Burrows Wheeler transform relying on Big Data technologies, i.e., Apache Spark and Hadoop. Our approach is the first that distributes the index computation, and not only the input dataset, thus allowing us to fully benefit from the available cloud resources.

[D3.2] *Implementation of a collection of some multicriteria indexes.*

Adopt the methodology and results of D3.1 for the design/orchestration of multicriteria indexes, and the software developed in D1.2 and D2.2 as their “building blocks”. Datasets will be synthetic and real, the latter borrowed from Web/text collections and BioInformatics.

- The paper on PGM-index is accompanied by the website https://pgm.di.unipi.it; the software and the tested datasets have been also published on the most important platform: GitHub and The Python Package Index.

- The reference C++ implementation of the PGM-index is available under the Apache-2 license at https://github.com/gvinciguerra/PGM-index.

- A Python package implementing sorted containers powered by the PGM-index is available under the Apache-2 license at https://github.com/gvinciguerra/PyGM.

- The software used in the Theoretical Computer Science paper is now part of a larger project and is available at https://github.com/felipelouza/egap.

- The software and datasets used in the Journal of Comp. Biology paper are available at https://github.com/alshai/r-index.

[T4] Multicriteria Data Compression

The theoretical and experimental scenario characterizing the design of Multicriteria compressors leads to devising the following two deliverables.

[D4.1] *Study and design a novel class of ML-based data compressors.*

The goal is to study first whether Lempel-Ziv or BWT transformations, or newly devised transformations, produce character distributions that could be better approximated via properly designed and space-conscious ML-models, resulting from tasks T1 and T2. And then, design a bi/multi-criteria optimization scheme that effectively (i.e. in de/compression time and space occupancy) combines those modules to achieve better lossless compression.

- Umberto Ferraro Petrillo, Francesco Palini, Giuseppe Cattaneo, Raffaele Giancarlo: FASTA/Q Data Compressors for MapReduce-Hadoop Genomics: Space and Time Savings Made Easy - Version 1. *Available at* arXiv:2007.13673 (2020).

  ABSTRACT. The contribution proposes meta-methods for data compression with the end result that one can easily import specialised genomic data compressors into a Big Data Platform such as Hadoop, making them splittable. This is a big advance in Big Data Technologies for Bioinformatics, thanks to the savings in time and space that can be achieved. Due to the design of our meta-methods, that at this time allow multi-choice data compression, future multi criteria data compressors developed within the project can be readily deployed in the Big Data Scenario.

[D4.2] *Implementation of a collection of some multi-criteria compressors.*

Here we adopt the methodology and results of D4.1 for the design of multi-criteria compressors, and the software developed in D1.2 and D2.2 as their “building blocks”. Datasets will be synthetic and real, those latter borrowed from Web/text collections and BioInformatics.

- A platform implementing the meta-data compressors for Hadoop and Spark is available at http://www.statistica.uniroma1.it/users/uferraro/experim/FASTdoopC/.

International collaborations

The research activities of the project have developed a series of international collaborations which prove their impact and potentialities:

- Research group of Prof. Idreos Stratos, Harvard University, on studies concerning the design and implementation of learned data structures and their uses in modern DBs;
- Researchers of the Dalhousie University (Canada), Kyushu Institute of Technology (Giappone) and the University of Chile on studies concerning the use of grammar compressors for ML models;
- Researchers of the University of Florida and Johns Hopkins University on studies concerning the compressed indexing of bio-sequences.

Invited talks of the first year

Some of the talks refer to the activity of the proponents started immediately after the communication by MUR of the acceptance of the project proposal, hence April 2019.

Italian Conference on Theoretical Computer Science (ICTCS). September 14, 2020. Online. "The future of data structures: data-aware and self-designing". Invited speaker: P. Ferragina.

IEEE MELECON Conference. June 16-18, 2020. Online. Special meeting on innovative startups and entrepreneurs: "Services of Big Data Analytics and Artificial Intelligence for Precision Medicine". Invited speaker: S. E. Rombo.

Dipartimento di Scienze Della Vita, UNIMORE. November 26, 2019. Modena. "Life sciences and algorithmic design: speed and accuracy in small space". Invited speaker: R. Giancarlo.

Dipartimento di Informatica (Dip. Eccellenza), Università di Roma "La Sapienza". October 29, 2019. Roma. "Algorithms, Theoretical Computer Science and Epigenomics: Mining Mathematical Laws for Predicting Chromatin Organization and Nucleosome Occupancy in Eukaryotic Genomes". Invited speaker: R. Giancarlo.

Italian Bioinformatics Conference. June 26-28, 2019. Palermo. "Combinatorial Messages and Epigenomics: The case of Chromatin Organization in Eukaryotic Genomes". Invited speaker: R. Giancarlo.

INNS Big Data and Deep Learning Conference. April 18-19, 2019. Sestri Levante. "Hybrid data structures and beyond". Invited speaker: P. Ferragina.

Spotify. April 2, 2019. London. "The evolution of searching data structures". Invited speaker: P. Ferragina.

Conference talks & seminars of the first year

46th Intl. Conference on Very Large Data Bases (VLDB). August 31-September 4, 2020. Online. “The PGM-index: a fully-dynamic compressed learned index with provable worst-case bounds”. Speaker: G. Vinciguerra.

37th Intl. Conference on Machine Learning (ICML). July 12-18, 2020. Online. “Why are learned indexes so effective?”. Speaker: G. Vinciguerra.

1st Intl. Workshop on Artificial Intelligence for Health (AIxHealth 2019) held in conjunction with the 18th Intl. Conference of the Italian Association for Artificial Intelligence (AIxIA 2019). November 19-22, 2019. “Deep Neural Networks' Architectural Issues and Data Representation Paradigms for Classification of DNA Sequences”. Speaker: G. Lo Bosco.

26th Intl. Symposium on String Processing and Information Retrieval (SPIRE). October 7-9, 2019. Segovia. “Space-Efficient Merging of Succinct de Bruijn Graphs”. Speaker: G. Manzini.

Analysing Big Omics Data Workshop. June 26, 2019. Palermo. “A Spark Algorithmic Paradigm For Spaced Words Alignment-Free Classification, with Focus on Phylogeny”. Speaker: R. Giancarlo.

Italian Bioinformatics Conference. June 26-28, 2019. Palermo. “Tripartite graph clustering for the prediction of lncRNA-disease associations”. Speaker: S. E. Rombo.

30th Annual Symposium on Combinatorial Pattern Matching (CPM). June 18-20, 2019. Pisa. “A New Class of Searchable and Provably Highly Compressible String Transformations”. Speaker: G. Manzini.

European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML PKDD). September 14-18, 2020. “Incremental Training of a Recurrent Neural Network Exploiting a Multi-Scale Dynamic Memory”. Speaker: A. Carta.