Data Management Engineering & Analysis Research Lab.

M. Oğuzhan Külekci

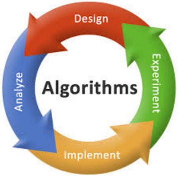
kulekci@itu.edu.tr

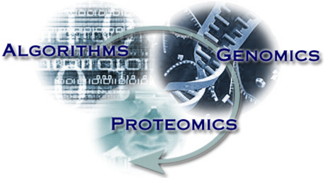
İSTANBUL TECHNICAL UNIVERSITY
INFORMATICS INSTITUTE

2015

Design, analysis, and engineering of discrete algorithms with applications on massive data management

FOUNDATIONS

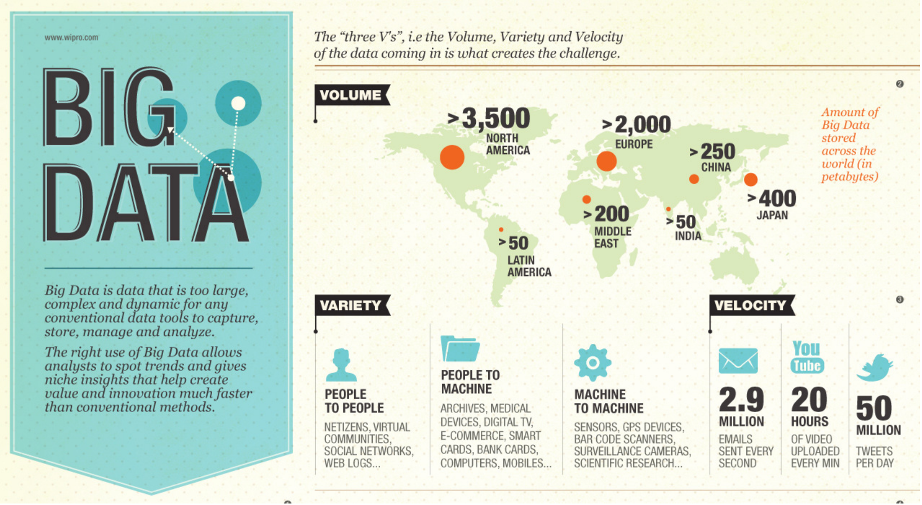




BIOINFORMATICS

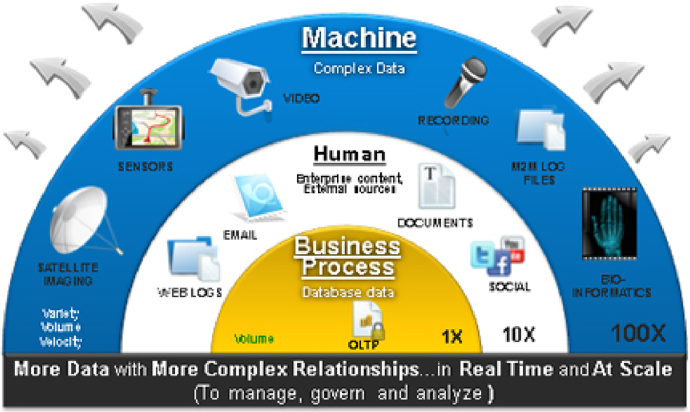
SEARCH TECHNOLOGIES

Motivation



Intel Big Data 101
The Economist: Big Data

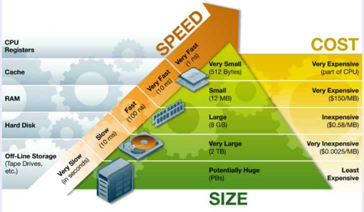
Sources of Big Data Growth



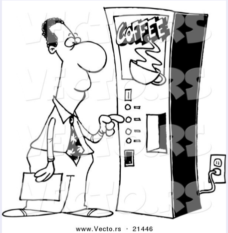
How to Sail in Big–Data Tsunami?

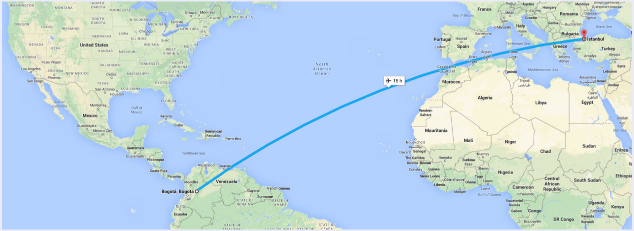
Do our best to

- perform minimal I/O
- fit data in memory as close as possible to CPU



Internal versus External Memory Access





FOUNDATIONS

Big Data $\Rightarrow$ Big Storage

When it comes to storage, there is never seems to be enough space.

Remove the redundancy, and squeeze the data down to its entropy, which is analogous to a vacuum storage bag.

We work on Lossless Data Compression

Many algorithms have been devised in basically three main families:

- Dictionary based (LZ family)
- Statistical (PPM and its variants)
- Transformation based (Burrows–Wheeler transform and extensions)

Lossless Data Compression: Selected Topics of Interest

Modeling

Trying to model how the target data is generated. Define the structure of the data, identify the redundancies to be removed.

M. Oğuzhan Külekci, Compressed context modeling for text compression , IEEE Data Compression Conference (DCC), 2011
M. Oğuzhan Külekci, A memory versus compression ratio trade-off in PPM via compressed context modeling, CoRR abs/1211.2636,2012 (*working paper*)

Coding

Actual coding of the data by removing redundancies according to the model. Huffman coding, arithmetic coding, fixed-to-variable (e.g.,Elias) and variable-to-fixed codes (e.g.,Tunstall), ...

B. Adas, E. Bayraktar, M. Oğuzhan Külekci, Huffman Codes versus Augmented Non-Prefix-Free Codes, under review, 2015
M. Oğuzhan Külekci, S. Thankachan, Range selection queries in data aware space and time, IEEE Data Compression Conference, 2015
M. Oğuzhan Külekci, Enhanced Variable-Length Codes: Direct Access with Improved Compression, IEEE Data Compression Conference, 2014
M. Oğuzhan Külekci, Uniquely decodable and directly accessible non-prefix-free codes via wavelet trees, IEEE International Symposium on Information Theory (ISIT), 2013
M. Oğuzhan Külekci, On enumeration of DNA sequences, ACM Conference on Bioinformatics, Computational Biology, and Biomedicine, Orlando, Florida, USA, 7-10 October, 2012
M. Oğuzhan Külekci, Enumeration of sequences with large alphabets. CoRR abs/1211.2926 (2012) (working paper)

Compression is nice, however...

What if you need to extract just one item from the zipped vacuum bag or to add a new one?

Inflate and deflate again?

Seems not very efficient...

Find ways to work directly on compressed data !

Compressed Data Structures

Compressed Data Structures

The Aim

Represent the data structure in space as small as possible, without a loss in its functionality.

G. Jacobson, Succinct Static Data Structures, PhD thesis, Carnegie Mellon University, 1989.
D. Clark: Compact Pat Trees, PhD thesis, University of Waterloo, Canada, 1996

- Compressed arrays, lists, trees, ...
- Very active area in the last decade especially in data management and information retrieval.

See the keynote speech delivered by Jeff Vitter at CIKM'12

Compressed Data Structures with Relevance

Compact integer representations

Given a list of integers, keep them compressed, while providing efficient random access.

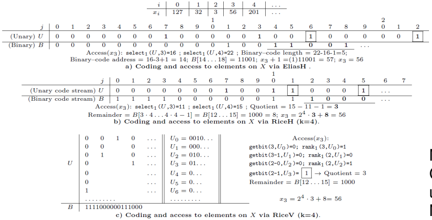


Figure 1: Coding and access to elements on X via EliasH, RiceH, and RiceV coding.

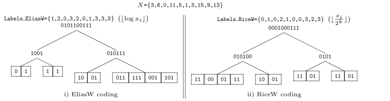
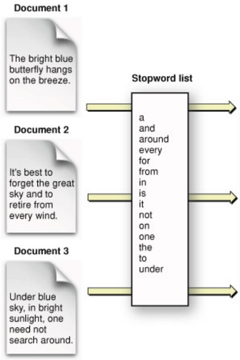


Figure 2: The i) *EliasW* and ii) *RiceW* (with $k = 2$) coding of the sample sequence X.

M. Oğuzhan Külekci, Enhanced Variable-Length Codes: Direct Access with Improved Compression, under review (*working paper*)

M. Oğuzhan Külekci, Uniquely decodable and directly accessible non-prefix-free codes via wavelet trees, IEEE International Symposium on Information Theory (ISIT), 2013

Compressed Data Structures: Selected Topics of Interest



Inverted index

ID	Term	Document
1	best	2
2	blue	1, 3
3	bright	1, 3
4	butterfly	1
5	breeze	1
6	forget	2
7	great	2
8	hangs	1
9	need	3
10	retire	2
11	search	3
12	sky	2, 3
13	wind	2

4 2 1 3 31 33 28 5 7 4

1, 5, 7, 8, 11, 42, 75, 103, 108, 115, 119, ...

Document ID lists are so long when you have billions of docs, and thus, stored in sorted order by saving the differences between the consecutive items.

Inverted-index compression

How to keep that differences array compact and provide capability to access any particular item.

Compressed prefix sum ! (working paper)

Compressed Data Structures: Selected Topics of Interest

s	$CRS_s(T)$
1	mississippi$
2	ississippi$m
3	ssissippi$mi
4	sissippi$mis
5	issippi$miss
6	ssippi$missi
7	sippi$missis
8	ippi$mississ
9	ppi$mississi
10	pi$mississip
11	i$mississipp
12	$mississippi

a Cyclic–right–shifts

s	$CRS_s(T)$
12	$mississippi
11	i$mississipp
8	ippi$mississ
5	issippi$miss
2	ississippi$m
1	mississippi$
10	pi$mississip
9	ppi$mississi
7	sippi$missis
4	sissippi$mis
6	ssippi$missi
3	ssissippi$mi

b Sorted CRS

i	F		L	i
1	$	mississipp	i	1
2	i	$mississip	p	2
3	i	ppi$missis	s	3
4	i	ssippi$mis	s	4
5	i	ssissippi$	m	5
6	m	ississippi	$	6
7	p	i$mississi	p	7
8	p	pi$mississ	i	8
9	s	ippi$missi	s	9
10	s	issippi$mi	s	10
11	s	sippi$miss	i	11
12	s	sissippi$m	i	12

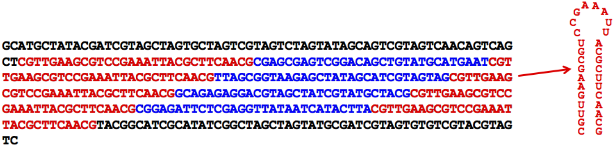
c $BWT(T) = L$

Compressed Text Indexing

- An index whose size is proportional to the entropy compressed size of the target text (compressed suffix arrays/trees)
- Self-index: No need to keep the original data as the index replaces the original text, e.g., FM-Index
- Alternative to inverted–index (a la Google) in much less space with *full–text* support.

New compressed text indexing schemes and/or improvements over the existing ones?

Compressed Data Structures: Selected Topics of Interest



Finding structures over data

Detecting structures such as maximal repeats, shortest unique strings, and other transformations, helps to understand the knowledge inside the data.

- Benefit from compression and compressed data structures to achieve such tasks efficient on massive data.
- Sometimes memory is not enough to load whole data, so the external memory approaches should not be missed.

M. Oğuzhan Külekci, Jeffrey Scott Vitter, Bojian Xu, Efficient Maximal Repeat Finding Using the Burrows–Wheeler Transform and Wavelet Tree, IEEE/ACM Transactions on Computional Biology and Bioinformatics, 9(2), 421-429, 2012

M. Oğuzhan Külekci, Jeffrey Scott Vitter, Bojian Xu, Time and space efficient maximal repeat finding using Burrows – Wheeler transform and wavelet trees, IEEE International Conference on Bioinformatics and Biomedicine, Hong Kong, December, 2010

Shortest unique substring and maximal repeat finding (working paper)

Compression is nice, however...

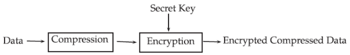
What if you don't want others to see what is inside the zipped bag?

Use a non-transparent bag?

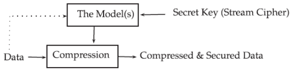
If so, how would you do look for some items in the non-transparent bags?

Security and privacy of the compressed data!

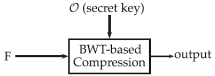
Privacy–Preserving Compression and Text Indexing



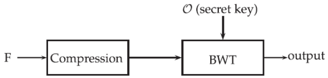
Approach 1 : Compress–then–encrypt



Approach 2: Unifying compression and security



Scrambled-BWT (sBWT)



Compress–then–sBWT

- Privacy-preserving text indexing and compression is a key challenge in cloud computing.
- Keep your data compressed on the cloud while pertaining the efficient search capability.
- We need to develop privacy-preserving compressed data structures and device new methods that would really work in practice.

M. Oğuzhan Külekci, On Scrambling the Burrows–Wheeler Transform to Provide Privacy in Lossless Compression, Computers & Security, 31(1), 26-32, 2012

M. Oğuzhan Külekci, A method to ensure the confidentiality of the compressed data, IEEE Conference on Data Compression, Communication, and Processing, Palinuro, Italy, 2011

BIOINFORMATICS

Wet Lab.

Biological question,
hypothesis generation and testing

BIOINFORMATICS

COMBINATORIAL CHALLENGES

Alignment, assembly, search, sequence compression, privacy, archival,structural variations....

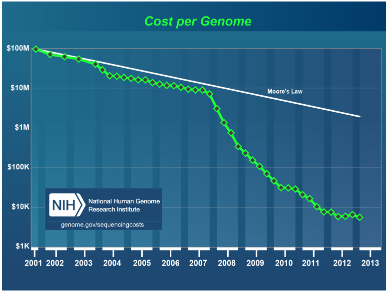
BIG DATA

MACHINE LEARNING

Artificial intelligence, data mining, biostatistics, Monte Carlo methods, HMM/bayesian inference, prediction...

We are mostly interested in the design and implementation of the tools targeting combinatorial challenges.

The Processing of Sequencing Data

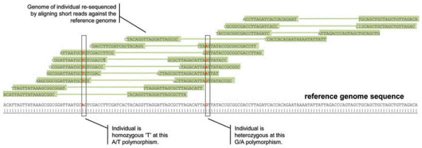


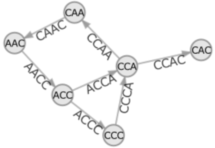
Genomputation: Computations over genomic data

Cost of sequencing rapidly drops and high-throughput machines produce terabytes of data.

- Alignment/assembly of the sequencing data
- Compression for efficient archival and transmission of sequence data
- Sequence database design
- Privacy/security of the biological sequences

Bioinformatics: Selected Topics of Interest





Alignment ⇐ Reference genome is available

- Map **billions** of reads onto the reference.

Assembly ⇐ De Novo (without reference)

- Solving a puzzle with billions of pieces.

- Alignment and assembly are hard problems due to
  i) massive set size **ii)** errors/mutations on the reads.
- Further processing required to extract information, such as SNP, CNV, structural variations, repeats, and etc.

M. O. Külekci, W. Hon, R. Shah, J. S. Vitter, B. Xu, PSI-RA: A parallel sparse index for genomic read alignment, BMC Genomics 12(S2),S7, 2011

M. O. Külekci, W. Hon, R. Shah, J. S. Vitter, B. Xu, PSI-RA: A parallel sparse index for read alignment on genomes, IEEE International Conference on Bioinformatics & Biomedicine, Hong Kong, December, 2010

M. O. Külekci, J. S. Vitter, B. Xu, Efficient Maximal Repeat Finding Using the Burrows-Wheeler Transform and Wavelet Tree, IEEE/ACM Transactions on Computional Biology and Bioinformatics, 9(2), 421-429, 2012

Bioinformatics: Selected Topics of Interest

Design and implementation of sequence databases

- Very soon, all of us will have our genomes sequenced as a daily practice in the hospitals.
- Traditional databases are not sufficient to store millions of people's genomes efficiently.
- Problem is not limited to storage, but efficient processing to turn that huge data into knowledge.
- How about privacy?

Genomic data compression for archival and/or transmission

- Sequence data is huge, but is also highly redundant.
- There are limitations in the classical methods to cope with huge data.
- Device new algorithms to archive/transmit sequencing data efficiently.
- Compression is also a powerful tool to seek a needle (information) in the haystack (big data).

SEARCH TECHNOLOGIES

One of the fundamental problems in computer science and engineering.
The core component of search engines and information retrieval applications.

Many algorithms on various instances including, but not limited to

- On–line (scanning) versus off-line (indexing)
- Exact vs. approximate
- Single–pattern vs. multiple–patterns
- Sequential versus parallel

string matching.

. . . and still new problems/opportunities emerge due to advances in technology, e.g., social media, computational biology, multi–cores, GPUs, ...

Search Technologies: Selected Topics of Interest

New string matching algorithms

We mainly focus on

- natural language searching
- biological sequence scanning
- financial data analysis
- intrusion/anti–virus/malware detection

to solve some bottlenecks and/or improve the performance in space/time.

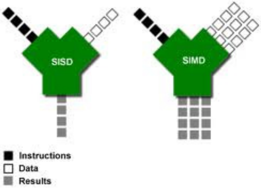
Inside the Search Algorithm

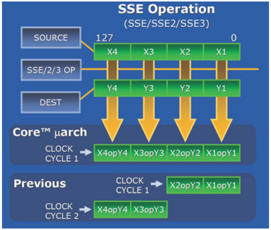
M. Oğuzhan Külekci, TARA: An algorithm for searching multiple patterns on text files, 22nd International Symposium on Computer and Information Sciences (ISCIS), Ankara, Turkey, November, 2007, (**Best paper award**)
M. Oğuzhan Külekci, A method to overcome computer word size limitation in bit-parallel pattern matching, 19th International Symposium on Algorithms and Computation (ISAAC), LNCS (5369), 496–506, Gold Coast, Australia, December, 2008
M. Oğuzhan Külekci, BLIM: A New Bit-Parallel Pattern Matching Algorithm Overcoming Computer Word Size Limitation, Mathematics in Computer Science 3(4), 407-420, 2010
M. O. Külekci, J. S. Vitter, B. Xu, Boosting pattern matching performance via k-bit filtering, 25th International Symposium on Computer and Information Sciences, LNEE(62), 27–33, London, UK, September, 2010
M. O. Külekci, J. S. Vitter, B. Xu, Fast Pattern-Matching via k-bit Filtering Based Text Decomposition, Computer Journal 55(1), 62-68, 2012

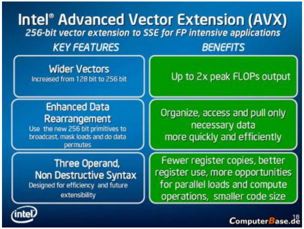
Search Technologies: Selected Topics of Interest

Algorithm Engineering: From theory to practice

- Design and implementation of the existing or new algorithms according to recent architectural advances of the processors.
- Currently, CPUs include special 128/256 bits long registers, e.g., Intel's SSE and AVX technologies.
- As oppose to classical single–instruction–single–data paradigm, now single–instruction–multiple–data parallelism is available.

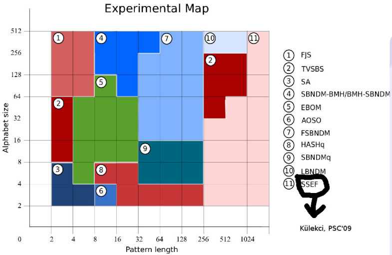






We work on improving text algorithms via SIMD parallelizations.

Search Technologies: Selected Topics of Interest



The Exact Online String Matching Problem: a Review of the Most Recent Results ACM Computing Surveys, Vol. 45(2) by Simone Faro and Thierry Lecroq

Packed String Matching

- Instead of processing symbols one–by–one, packed string matching operates on blocks of symbols.
- A new trend appeared to optimize string matching with SIMD parallelization.
- Pattern matching algorithms designed and implemented with SIMD outperforms the competitors very significantly in most cases.

M. O. Külekci, Filter based fast matching of long patterns by using SIMD instructions, Prague Stringology Conference (PSC), p. 118–129, Prague, Czech Republic, August, 2009

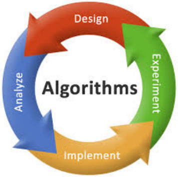
S. Faro, M. O. Külekci, Fast multiple string matching using streaming SIMD technology, 19th International Symposium on String Processing and Information Retrieval (SPIRE), Cartagena, Columbia, October 20-25, 2012, LNCS (7608), 217-229

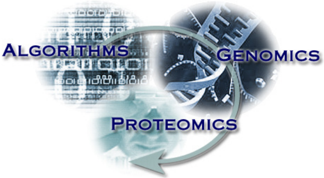
S. Faro, M. O. Külekci, Towards a Very Fast Multiple String Matching Algorithm for Short Patterns, Prague Stringology Conference, Prague, Czech Republic, September 1-4, 2013

S. Faro, M. O. Külekci, Fast packed string matching for short patterns, Meeting on Algorithm Engineering and Experiments (ALENEX), New Orleans, Louisiana, USA, January 6-8, 2013

Design, analysis, and engineering of discrete algorithms with applications on massive data management

FOUNDATIONS





BIOINFORMATICS

SEARCH TECHNOLOGIES