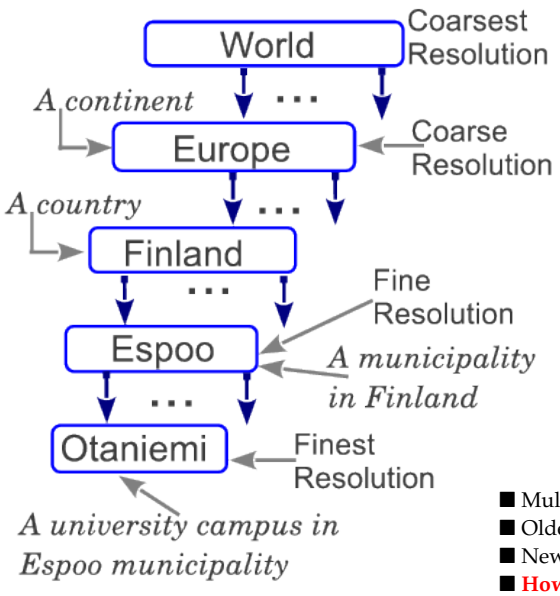
MIXTURE MODELS FROM MULTIRESOLUTION 0-1 DATA

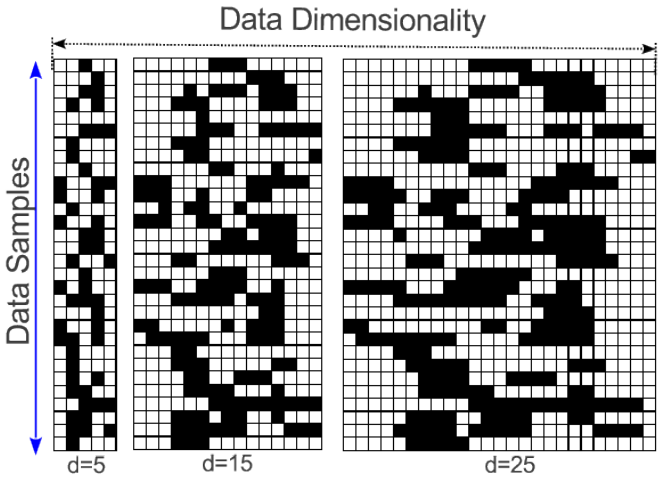
Prem Raj Adhikari[1,2] and Jaakko Hollmén[1,2] , {prem.adhikari, jaakko.hollmen}@aalto.fi

[1]Aalto University School of Science, and [2]Helsinki Institute for Information Technology,
Department of Information and Computer Science, PO Box 15400, FI-00076 Aalto, Espoo, Finland

PART-OF-HIERARCHY: EXAMPLE

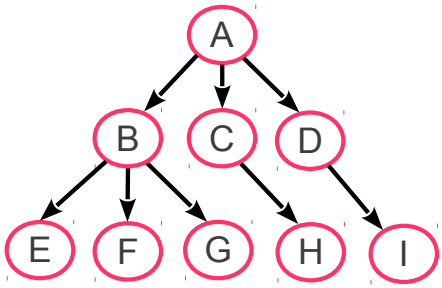


MULTIRESOLUTION DATA



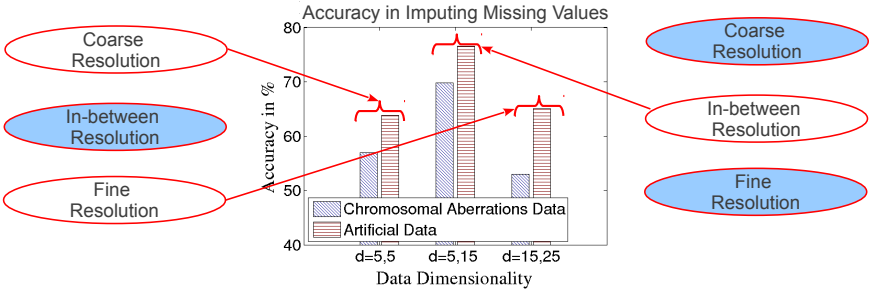
- Multiresolution data is everywhere: biology, computer vision, telecommunications ...
- Older Generation Technology ⇒ Data in Coarse Resolution
- Newer Generation Technology ⇒ Data in Fine Resolution
- **How to analyze data in multiple resolutions in a single analysis?**

BAYESIAN NETWORK FROM MULTIPLE RESOLUTIONS



A ~ Europe; **B** ~ Finland; **C** ~ Sweden;
D ~ Denmark; **E** ~ Espoo; **F** ~ Tampere;
G ~ Turku; **H** ~ Stockholm; **I** ~ Copenhagen;

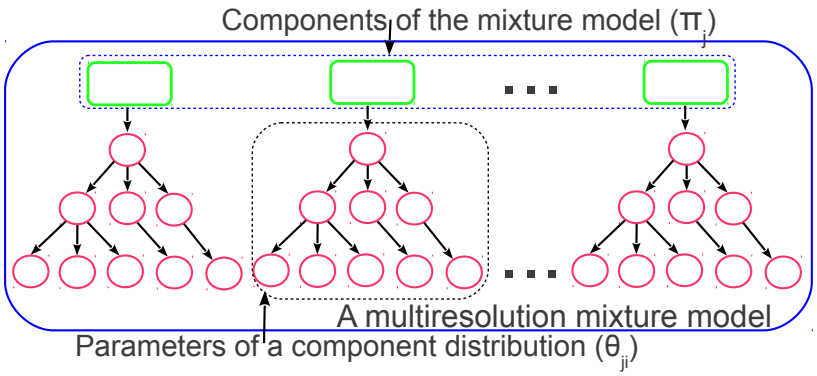
IMPUTING MISSING RESOLUTIONS USING BAYESIAN NETWORKS



For a joint distribution P(A,B,C) and an evidence B=true, marginal inference calculation is:
$P(A \mid B = true) \propto \sum_C P(A, B = true, C)$.

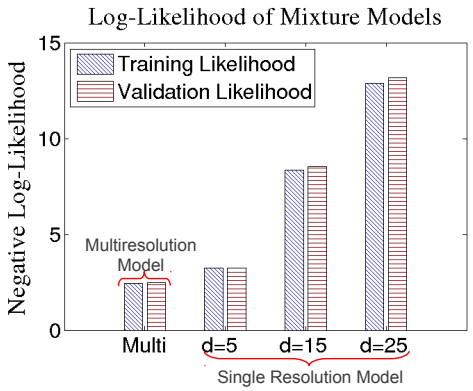
To impute missing values, we draw samples under given evidence from consistent junction tree using BRMLToolbox. Comma in labels in X-axis separates dimensions of two datasets.

MIXTURE MODEL WITH MULTIRESOLUTION COMPONENTS



The components of mixture model are Bayesian networks themselves. We use EM algorithm in a 10-fold cross-validation setting to learn parameters of the mixture model.

MIXTURE MODELLING RESULTS



The Y-axis shows the negative log likelihood, therefore, the shorter the bar, better the result

The work is funded by Helsinki Doctoral Programme in Computer Science - Advanced Computing and Intelligent Systems (**Hecse**)

REFERENCES

P. R. Adhikari and J. Hollmén. **Multiresolution Mixture Modeling using Merging of Mixture Components.** *Proceedings of the 4th ACML*, volume 25 of *ACML'12, Singapore*, pages 17–32. JMLR, 2012.

A. S. Willsky. **Multiresolution Markov models for signal and image processing**. *Proceedings of the IEEE*, 90(8):1396–1458, 2002.

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

PREDICTING THE HARDNESS OF LEARNING BAYESIAN NETWORKS

Brandon Malone, Kustaa Kangas, Matti Järvisalo, Mikko Koivisto, Petri Myllymäki

Motivation: There are various algorithms for finding a Bayesian network structure that is optimal with respect to a given scoring function. Due to the chaotic nature of the running times of such algorithms, it is *a priori* not clear which algorithm will solve a given problem instance fastest. **Results:** 1) We can train models that predict the running time of an algorithm on a given instance with reasonable accuracy based on *features* of the instance. 2) Even very simple features admit an efficient hybrid algorithm, or *portfolio*, that runs the algorithm predicted to be fastest.

INTRODUCTION

BAYESIAN NETWORKS

A Bayesian network is a graphical model on random variables $X_1, \ldots, X_n$.

The *structure* of a Bayesian network is a directed **acyclic** graph (DAG) G.

A *scoring function* s measures how well G fits observed data on the variables. Typical scoring functions decompose into a sum

$$s(G) = \sum_{i=1}^{n} s_i(G_i) ,$$

where G_i is the set of parents of X_i in G.

Common s: penalized likelihood, minimum description length, BDeu, etc.

STRUCTURE LEARNING PROBLEM

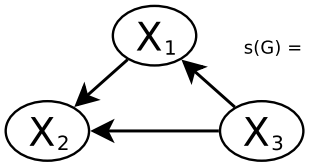
Input: A set $\mathcal{G}_i$ of candidate parent sets for each variable X_i and the local scores $s_i(G_i)$ for all $G_i \in \mathcal{G}_i$.

Task: Find a DAG G such that $G_i \in \mathcal{G}_i$ and the score $s(G)$ is maximized. (NP-hard)

G_1	$s_1(G_1)$
{X_2, X_3}	7
{X_2}	4
{X_3}	6
Ø	3

G_2	$s_2(G_2)$
{X_1, X_3}	8
{X_1}	3
{X_3}	2
Ø	1

G_3	$s_3(G_3)$
{X_1, X_2}	4
{X_1}	3
{X_2}	3
Ø	2



s(G) = 6 + 8 + 2 = 16

ALGORITHMS

Various exact algorithms are guaranteed to find an optimal G while avoiding exhaustive search in the space of all DAGs:

Dynamic programming over variable subsets finds an optimal ordering of variables that is compatible with an optimal DAG.

A* search formulates the DP approach as a shortest-path problem, uses admissible best-first heuristics to prune the search space.

Integer linear programming searches a convex polytope where each vertex is a feasible solution. Cutting planes are added during search to enforce acyclicity.

Branch and bound searches a relaxed space of cyclic graphs and breaks cycles by branching on arcs to remove in best-first order.

MODEL TRAINING

1. Select a set of training instances.
2. Select a set of instance *features*.
3. Compute the features of each instance.
4. Run all algorithms on all instances and record their running times.
5. Using the data, learn for each algorithm a model that maps a feature vector to a running time prediction.

FEATURES

We consider 74 features of various types:

1. Number of variables n, number of candidate parent sets $m = \sum_{i=1}^{n} |\mathcal{G}_i|$ (typically $|\mathcal{G}_i| \ll 2^{n-1}$ due to pruning).
2. Sizes of $G_i \in \mathcal{G}_i$: mean, variance, etc.
3. Properties of cyclic upper bound graphs: average degree, number of leaves, etc.
4. Probing: Properties extracted by running one algorithm for a few seconds: best network found, lower bounds, etc.

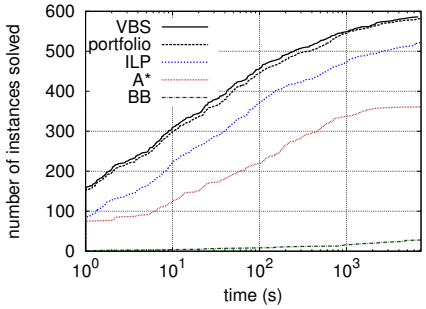
PREDICTORS

We use REP trees to train two predictors:

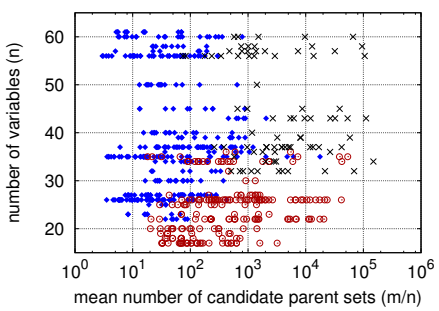
Predictor A: Uses the features n and m.
Predictor B: Uses all features.

PORTFOLIO

Given a new instance, a simple portfolio runs the algorithm predicted to be fastest by predictor A. Comparison to individual algorithms and the Virtual Best Solver that makes perfect predictions:

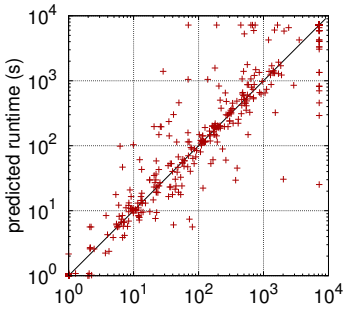


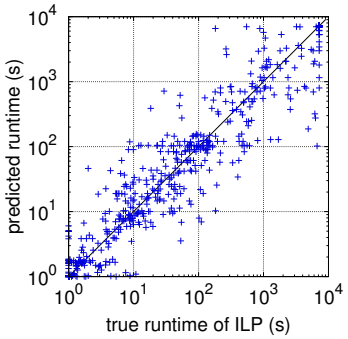
Orthogonality between dominant solvers w.r.t. n and m. Blue instances were solved faster by ILP, red ones by A*:



PREDICTION

Although the simple predictor A already admits an efficient portfolio algorithm, predictor B makes more accurate predictions:





HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

LEMPEL-ZIV FACTORIZATION IN EXTERNAL MEMORY

Juha Kärkkäinen
Dominik Kempa
Simon J. Puglisi

For over three decades, the Lempel-Ziv factorization (or LZ77 parsing) has been a fundamental tool for data compression (e.g. in 7-zip). More recently it has become the basis for several compressed text indexes which are particularly effective for massive, highly repetitive data sets. When computing the parsing for such large data sets, the space requirements of algorithms can become a problem. We escape the limitations of RAM by describing the first external memory LZ77 parsing algorithms and present their experimental comparison.

LEMPEL–ZIV FACTORIZATION

Lempel-Ziv factorization $LZ(T)$ of string T is a greedy partition of T into *longest previous factors* (LPFs). LPF at position i is the longest factor $T[i..i+\ell)$ that also occurs at some position $j < i$.

Example:

i	0	1	2	3	4	5	6	7	8	9
T[i]	A	B	A	B	B	A	B	B	A	B

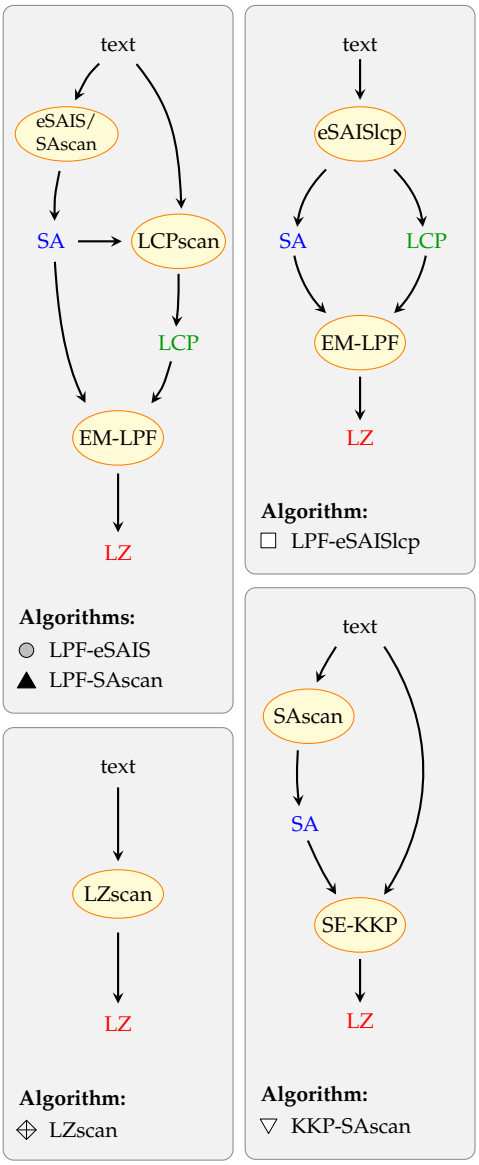
LPF[2] = AB (j = 0)
LPF[5] = ABBAB (j = 2)

LZ(T): | A | B | A B | B A B B A B |

BASIC ALGORITHMS

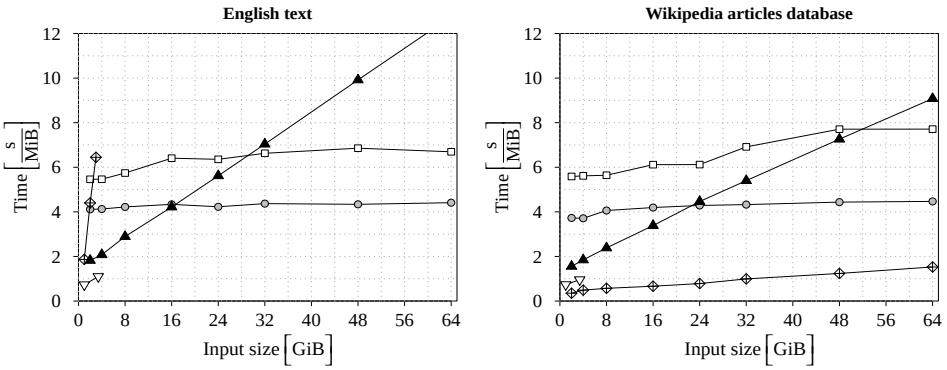
Name	I/O complexity	Space
SAscan [4]	$\mathcal{O}\left(\frac{n}{B}\left(1+\frac{n\log\sigma}{M\log n}\right)\right)$	$6.5n$
eSAIS [1]	$\mathcal{O}\left(\frac{n}{B}\log_{\frac{M}{B}}\frac{n}{B}\right)$	$28n$
eSAISlcp [1]	$\mathcal{O}\left(\frac{n}{B}\log_{\frac{M}{B}}\frac{n}{B}\right)$	$54n$
LCPscan [3]	$\mathcal{O}\left(\frac{n}{B}\left(1+\frac{n\log^2\sigma}{M\log^2 n}\right)\right)$	$16n$
EM-LPF [5, 2]	$\mathcal{O}\left(\frac{n}{B}\log_{\frac{M}{B}}\frac{n}{B}\right)$	$26n$
LZscan [5]	$\mathcal{O}\left(\frac{n}{B}\cdot\frac{n\log\sigma}{M\log n}\right)$	$1.5n$
SE-KKP [5]	$\mathcal{O}\left(\frac{n}{B}\right)$	$21n$

LZ FACTORIZATION ALGORITHMS



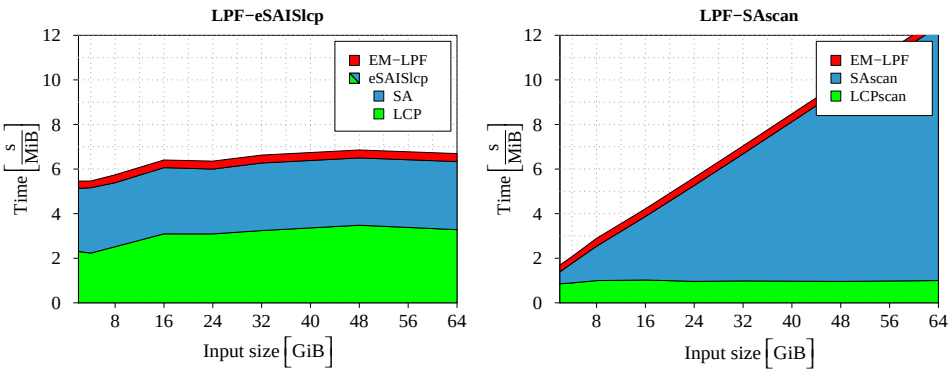
EXPERIMENTAL COMPARISON

We implemented and compared all LZ factorization algorithms depicted on the right. The algorithms were executed on varying size prefixes of two testfiles: a large data set containing English text (left) and a database of Wikipedia articles containing many versions of the same articles (right). All algorithms were allowed to use 3.5GiB of internal memory. The results depend on the amount of repetitions in the input text. LZscan dominates all other algorithms for highly repetitive input but performs poorly when the data is less repetitive, such as the English test file. The fastest algorithm for such data is determined by the ratio of input size to available RAM.



DETAILED RUNTIME BREAKDOWN

Below we present a detailed runtime breakdown of LPF-eSAISlcp and LPF-SAscan executed on English text. The graphs reveal that most of the time is spent during the computation of supporting data structures (SA and LCP). The LCP array construction is significantly accelerated with the use of our new algorithm (LCPscan) which makes SA construction the slowest phase of the factorization. The main challenge in efficient and scalable LZ factorization is therefore developing new methods for suffix sorting, possibly using parallel or distributed computation.



REFERENCES

[1] T. Bingmann, J. Fischer, and V. Osipov. Inducing suffix and LCP arrays in external memory. In *Proc. ALENEX*, pages 88–102, 2013.

[2] M. Crochemore, L. Ilie, and W. F. Smyth. A simple algorithm for computing the Lempel-Ziv factorization. In *Proc. DCC*, pages 482–488, 2008.

[3] J. Kärkkäinen and D. Kempa. LCP array construction in external memory. Accepted to SEA 2014.

[4] J. Kärkkäinen and D. Kempa. Engineering a lightweight external memory suffix array construction algorithm. In *Proc. ICABD*, 2014. To appear.

[5] J. Kärkkäinen, D. Kempa, and S. J. Puglisi. Lempel-Ziv parsing in external memory. In *Proc. DCC*, 2014. To appear.

Discovering Significant Episodes

Nikolaj.Tatti@aalto.fi
Aalto University, Helsinki Institute of Information Technology

Aalto University

HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

Sequential patterns

What event sets occur close to each other?

Episodes:

- a set of events
- constraints on the order (repsesented by DAG)

- a should occur first
- then b and c, or c and b
- finally d

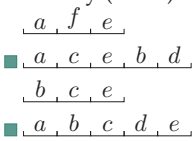
Occurrence and support

When an episode appears in the sequence?

- all events should appear in the sequence
- DAG should be respected
- gaps are allowed

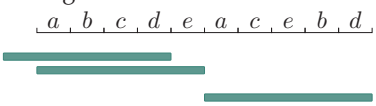
How often episode occur?

Input: Many (short) sequences

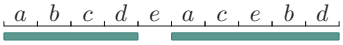


Input: One long sequence

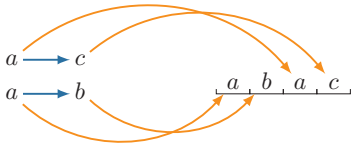
- sliding windows of fixed size

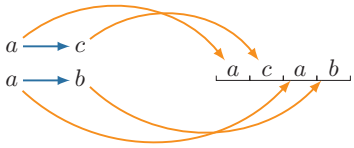


- minimal windows



Testing occurrence is **NP**-hard
...not sure how to map nodes to a sequence





Not a problem in practice:

- episodes are small: do full enumeration
- polynomial delay
- use subclasses
  - parallel / serial / strict

Frequent Episode Mining

Find *all* episodes that have high support.

Support is monotonic.
We can generate episodes efficiently by

- adding nodes
- adding edges
- stop if the episode is not frequent

Pattern explosion

High support:

- discovered patterns are trivial

Low support:

- too many patterns
- mostly redundant

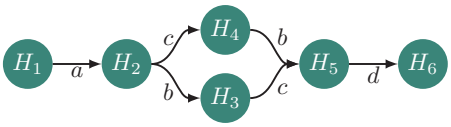
Pattern ranking

- compute the support using some statistical model
- compare the observed support with the expectation
- patterns with large devation are interesting

Independence model

What is the expected support w.r.t. the independence model?

Construct a finite state machine from the episode:



G appears in the sequence if and only if we can reach H_6 from H_1

Compute the probabilities iteratively

$$P(H_i \mid k) = \sum_{H_j \in par(H_i)} p(lab) p(H_j \mid k-1) + q p(H_i \mid k-1),$$

where

$$q = 1 - \sum_{H_j \in par(H_i)} p(lab) \quad .$$

Example:

$$p(H_6 \mid k) = p(d) p(H_5 \mid k-1) + (1 - p(d)) p(H_6 \mid k-1) \quad .$$

$$p(H_5 \mid k) = p(b) p(H_4 \mid k-1) + p(c) p(H_3 \mid k-1) + (1 - p(b) - p(c)) p(H_5 \mid k-1) \, .$$

Partition model

If b occurs often aftern a, then both

and

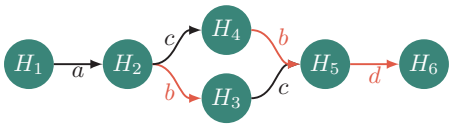
are significant.

How to report only G_1 and downrank G?

Reduce redundancy using a partition model.

- divide episode into two subepisodes
- model how soon we can discover each subepisode and assume independence between them
- use the new model to compute more accurate expected support

Model how soon b occurs after a and how soon d occurs after c.

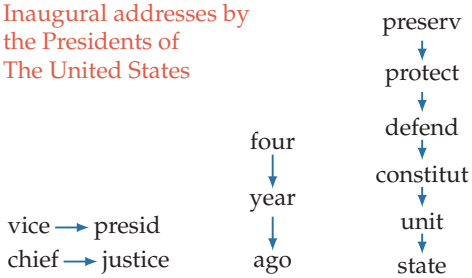


Use the new model to compute the expected support:

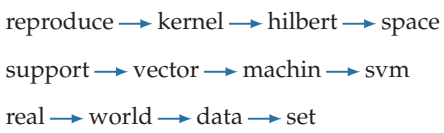
- b appears soon after a
- transition probabilities from H_2 to H_3 and from H_4 to H_5 are large
- expected support is increased
- observed support is closer to the expected support
- G is not highly ranked anymore

Examples from text data

Inaugural addresses by the Presidents of The United States



Abstracts from Journal of Machine Learning (JMLR)



> Size matters <

Finding the most informative set of window lengths

Jefrey Lijffijt [1], Panagiotis Papapetrou [2] and Kai Puolamäki [3]

[1] Aalto University, [2] Stockholm University, and [3] Finnish Institute for Occupational Health

> Problem setting

When looking for local patterns in sequential data, it is often difficult to choose the right granularity (window length) for analysis.

> Solution

Select the most informative window length by mapping the problem to a regression problem.

Even better, select a small set of window lengths.

> Problem statement

Let $S = \langle s_i, \ldots, s_n \rangle$ denote an event sequence and $S_{i,m} = \langle s_i, \ldots, s_{i+m-1} \rangle$ a subsequence of length m.

Given an algorithm that takes as input a subsequence of any length m and as output gives a real number $f(S_{i,m})$, a set of possibly interesting window lengths W, the set of k most informative window lengths W^* is given by

$$\underset{W^* \subseteq W:\, |W^*|=k}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n'} \sum_{w \in W} \min_{w^* \in W^*} C\left(f(S_{i,w}), f(S_{i,w^*})\right) \right\}.$$

> Example

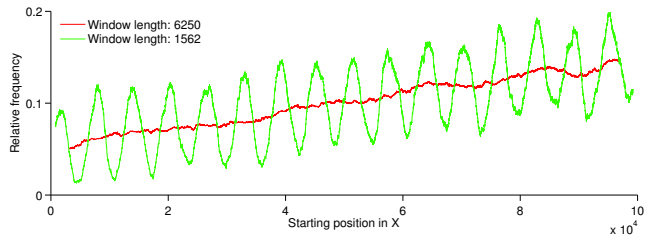


Figure 1: Generative processes may have multiple components. Such structure can only be uncovered by studying multiple window lengths concurrently.

> Burstiness of words

We computed optimal sets of window lengths for several bursty and non-bursty words in Jane Austen's Pride & Prejudice.

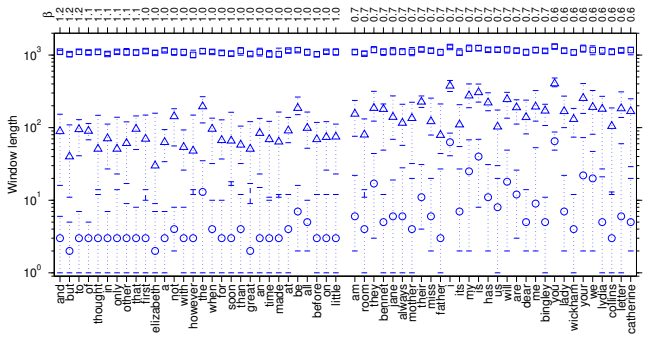


Figure 2: Bursty words give longer window lengths, because the scale structure is less gradual then for uniformly distributed words.

> Detailed solution

If the cost function is the squared error: $C(x,y) = |x-y|^2$, the optimization problem is an instance of the k-medoids clustering problem.

The optimization problem is NP-hard, but can be approximated efficiently using the Clustering LARge Applications (Clara) algorithm [1].

We modify the algorithm by using the smart seeding from the k-means++ algorithm [2], providing an approximation guarantee. We call this algorithm Clara++.

> Publication

A preliminary version has appeared as Lijffijt, Papapetrou & Puolamäki. Size matters: Finding the most informative set of window lengths. In *Proc. of ECML-PKDD*, 2012.

The full paper is currently under review.

> Sensor measurements

We applied our method to data from a strain sensor on De Hollandse Brug, a bridge in the Netherlands.

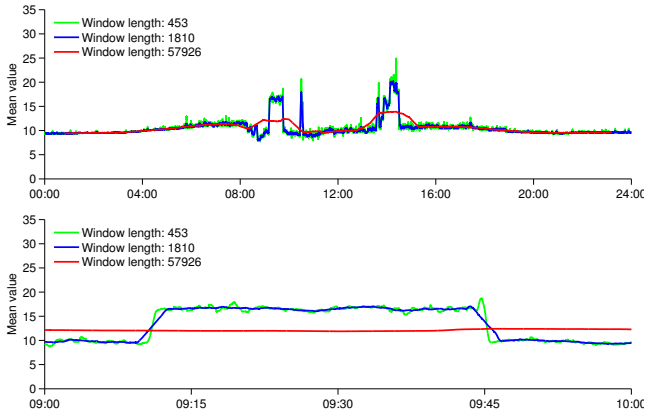


Figure 3: By using multiple window lengths, we can see different patterns in the data. The three lines show trends of the amount of traffic on the bridge at different time scales. The blue and green lines are fairly similar, suggesting that two window lengths suffice.

> References

[1] Kaufman & Rousseeuw. *Finding Groups in Data: An Introduction to Cluster Analysis*. John Wiley & Sons, 1990.

[2] Arthur & Vassilvitskii. k-means++: the advantages of careful seeding. In *Proc. of SODA*, 2007.

> Acknowledgements

This work has been funded by the Finnish Centre of Excellence in Algorithmic Data Analysis (ALGODAN) and the Finnish Doctoral Programme in Computational Sciences (FICS). We thank Heikki Mannila for useful feedback and discussions.

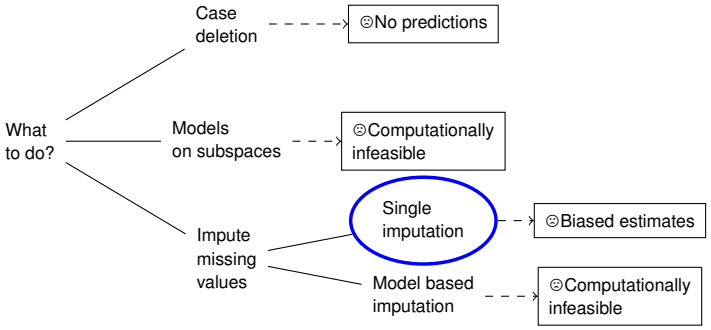
Regression Models for Data Streams with Missing Values

Indrė Žliobaitė and Jaakko Hollmén
Aalto University, Dept. of Information and Computer Science and Helsinki Institute for Information Technology, Finland
{indre.zliobaite, jaakko.hollmen}@aalto.fi

Problem setting and assumptions

- Predictive modeling for streaming data
  - data arrives and needs to be analyzed in real time
  - data distribution may change over time
  - predictive model adapts during operation
- **Problem**: massively missing input data (∼50% of records), while predictions are needed non-stop
- **Our approach**: make predictive models robust to missing data
- **Focus**: linear regression models
- **Benefits**: very fast imputation, computationally light and online updatable models

Possible solutions



Expected prediction error

The expected error of a linear prediction model with mean imputation is

$$E[MSE_p] = (1-p)E[MSE_0] + p - p(1-p)\beta^T(\mathbf{\Sigma}-\mathbf{I})\beta,$$

p - probability of a missing value in an observation vector, β - regression coefficients, $\mathbf{\Sigma}$ - covariance matrix of the input data, MSE_0 - the error when no data is missing. Assumption: variables are missing independently with the uniform prior probability.

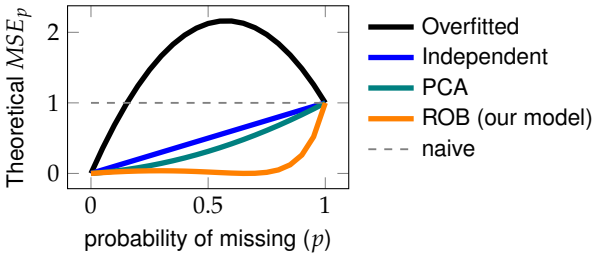
Deterioration index: $D = \beta^T(\mathbf{\Sigma}-\mathbf{I})\beta$.

Implications

- If $\mathbf{\Sigma} = \mathbf{I}$ (independence), then $D = 0$ and MSE increases linearly in p.
- If $\mathbf{\Sigma} \neq \mathbf{I}$ and $D < 0$ (overfitting) then MSE increases quadratically.
- If $\mathbf{\Sigma} \neq \mathbf{I}$ and $D > 0$ then MSE increases **only** sub-linearly.

Example

Data: $x_1 = x_2 = x_3 = x_4 = y$, $x_1 \sim \mathcal{N}(0,1)$.



Four regression models (for all $MSE_0 = 0$):

- **Independent**: $\hat{y} = x_1$;
- **PCA**: $\hat{y} = 0.25x_1 + 0.25x_2 + 0.25x_3 + 0.25x_4$;
- **Overfitted**: $\hat{y} = 2x_1 - 1.5x_2 + x_3 - 0.5x_4$;
- **ROB regression**: different model for each value of p.

How to build robust regression models?

Minimize $E[MSE_p]$. Theoretically optimal solution is

$$\hat{\beta}_{ROB} = \left((1-p)\mathbf{X}^T\mathbf{X} + pn\mathbf{I}\right)^{-1}\mathbf{X}^T\mathbf{y},$$

p - probability of a missing value in an observation vector, $\mathbf{X}$ - training data inputs, $\mathbf{y}$ - training targets, n - training set size, $\mathbf{I}$ - identity matrix.

ROB is similar to the Ridge regression $\hat{\beta}_{RR} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{y}$.

Online updates with streaming data

For each new observation $\mathbf{x}$, predict $\hat{y} = \mathbf{x}\hat{\beta}$.

When the true target y arrives, update the model:

- $\mathbf{S} \leftarrow \mathbf{S} + \mathbf{x}\mathbf{x}^T - p(\mathbf{x}\mathbf{x}^T - \mathbf{I})$
- $\hat{\beta} \leftarrow \hat{\beta} + \mathbf{S}^{-1}\mathbf{x}(y - \mathbf{x}^T\hat{\beta}) - \mathbf{S}^{-1}p(\mathbf{x}\mathbf{x}^T - \mathbf{I})\hat{\beta}$

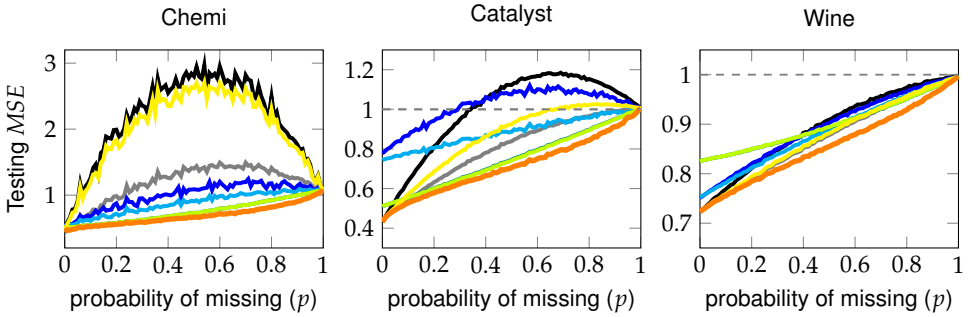
$\mathbf{S}$ is the covariance estimate, offline for centered data $\mathbf{S} = \mathbf{X}^T\mathbf{X}/n$.

Performance

Compared models: **ROB** and

Inputs	Optimization	
	OLS	RR
all r	**ALL**	rALL
selected k	**SEL**	rSEL
PCA k	**PCA**	rPCA
PLS k	**PLS**	

OLS - Ordinary least squares, RR - Ridge regression



The proposed ROB regression consistently achieves the best performance.

References: Žliobaitė, I., Hollmén, J. (2013). **Fault tolerant regression for sensor data.** *Proc. of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECMLPKDD'13)*, Springer LNAI 8188, p. 449-464.
Žliobaitė, I., Hollmén, J. **Optimizing regression models for data streams with missing values**. Journal paper under review (MLJ).

Acknowledgement

This work has been supported by the Academy of Finland grant 118653.

Mining the near infrared sky: star formation and embedded clusters

Otto Solin[1,2], Lauri Haikala, Esko Ukkonen[1]

[1] University of Helsinki, Department of Computer Science, [2] University of Helsinki, Department of Physics

otto.solin@helsinki.fi

Major part of star formation, be it low- or high-mass stars, takes place in clusters. The clusters are not bound and will eventually disrupt e.g. because of the Galactic differential rotation. The stellar clusters trace therefore the recent Galactic star formation. The younger the clusters are the more compact they are and the more closely they are associated with the interstellar gas and dust clouds they formed in. Detailed study of young clusters still associated with their parent cloud will provide information on the star formation process and the stellar initial mass function.

At the moment some 2000 Galactic stellar clusters are known. This is only a small fraction of the estimated total population of which a major part is obscured by interstellar dust to us and can not be observed in optical wavelengths. However, the extinction decreases at longer wavelengths and in the *K*-band (2.2 microns in the near infrared) the extinction is only 11 percent of that in the optical *V*-band (0.55 microns).

The aim of this research is to develop methods to locate previously unknown stellar clusters from two near infrared surveys: the UKIDSS Galactic Plane Survey (GPS; Lucas et al. 2008) mapping the northern plane of the Milky Way, and the VISTA variables in the Vía Láctea (VVV; Minniti et al. 2010) survey mapping the Galactic bulge and the southern disk. These new surveys don't cover the whole sky but they are many times deeper than their predecessor, the Two-Micron All-Sky Survey (2MASS; Skrutskie et al. 2006) covering the whole Milky Way.

The search method takes pre-filtered catalogue data, divided into overlapping bins, and performs a maximum likelihood fitting of a mixture of a Gaussian density and a uniform background. On each bin the fitting is done using the standard Expectation Maximization (EM) algorithm. The real clusters and locations of star formation are selected by visually inspecting the images of the cluster candidate areas suggested by the automated search of the catalogue data. In addition to the UKIDSS and VVV catalogues, stars brighter than 10^m in *K* from the 2MASS survey are used, because the brighter stars saturate in UKIDSS and VVV, and moreover tend to produce false positives around them.

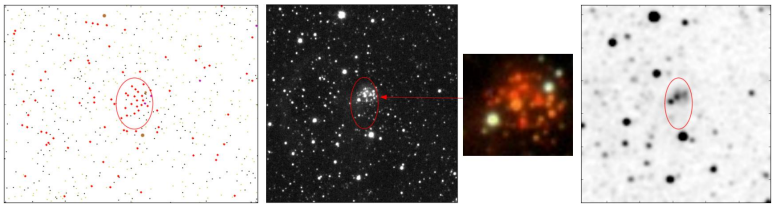
Scrutiny of the data base and the survey images reveals that the survey pipeline source detection algorithm tends to classify most of the objects within regions of variable surface brightness as non-stellar (parameter mergedClass=+1), whereas objects with intensity profiles similar to the cameras point spread function are classified as star-like (mergedClass=-1). Clustering non-stellar sources directs the search to stellar clusters either embedded in or near molecular/dust clouds. Besides stellar clusters, the search targets also the locations of non-clustered star formation and single embedded stars with associated nebulosities. The surface brightness, either due to outflow activity or reflection, will produce "cluster" detections.

For UKIDSS as expected most of the detected new clusters (137) and sites of star formation (30) are tightly concentrated on the Galactic plane. Relatively few new clusters were detected in the direction of the northern Galactic plane because this is in the direction of the Galactic anticentre where the absolute number of clusters is much lower than that in the inner galaxy. Likewise for VVV most candidates (88 clusters and 39 sites of star formation) are in the Galactic plane outside the bulge area where the contamination from the field stars is overwhelming and our method is not able to trap the clusters.

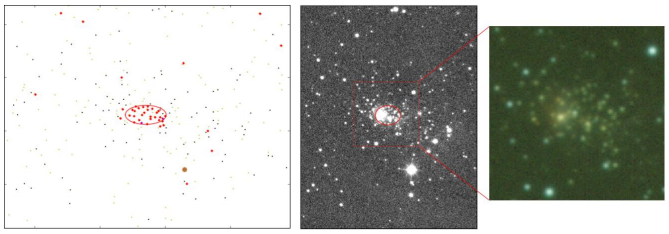
Most images of the new cluster candidate areas show clear signs of reflected light in particular in in the *K*-band thus indicating embedded clusters or sites of star formation.

The next step in this research is to locate clusters using the measured colours of the stars.

New cluster candidate identified previously as an infrared point source



In the leftmost panel are the UKIDSS catalogue entries in the cluster area. The red points are UKIDSS non-stellar sources brighter than 17^m in *K*, black points other sources brighter than 17^m in *K*, yellow points sources fainter than 17^m in *K*, and brown points sources listed in 2MASS but not in UKIDSS GPS. The red confidence ellipse is the cluster area given by the EM-algorithm. In the two middle panels are the *K*-band and *JHK* false colour images of the cluster area. In the 2MASS image (the rightmost panel) of the same area no cluster can be seen.



Besides an IRAS point an MSX source, an HII region and a submillimetre source are detected in the direction of this candidate.

The number of indicators (IRAS, MSX, (sub)mm sources,masers, and HII regions) seen in the direction of many candidates gives confidence the new clusters or embedded star formation locations are real entities and not produced by chance nor are due to catalogue artefacts. In general radio surveys find circumstellar dust envelopes and disks, and cold cores of molecular clouds. In areas where a radio telescope sees only a point source or signs of e.g. an ultracompact HII region, the UKIDSS and VVV images show structures of surface brightness and single stars thus verifying the results of the millimetre/submillimetre radio surveys of suspected star forming regions.

Specifically, many candidates are associated with infrared dark clouds. This is not surprising as these clouds are assumed to be the forming sites of massive clusters.

Zone of avoidance galaxies (ZOAGs) have been identified in the direction of four of the UKIDSS cluster candidates. So instead of being extragalactic sources these are Galactic clusters. On the other hand the cluster search using the VVV survey resulted in four new ZOAGs.

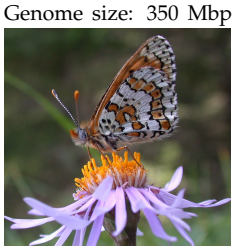
The results for both surveys have been published in the journal Astronomy & Astrophysics (DOI: 10.1051/0004-6361/201118531 and 10.1051/0004-6361/201322890).

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

ALGORITHMS FOR GENOME ASSEMBLY

Leena Salmela, Veli Mäkinen, Niko Välimäki, and Esko Ukkonen

Genome size: 350 Mbp



Melitaea cinxia
Photo: Niclas Fritzén

Reads

	454	SOLiD	Illumina	PacBio
Reads	12 million	210 million	349 million	2.7 million
Read length	400–800 bp	50 bp	75–150 bp	Up to 23.5 kbp
Errors	Indels	Mismatches	Mismatches	Indels
Paired end	-	-	600 bp, 800 bp	-
Mate pairs	7 kbp, 16 kbp	2–5 kbp	1 kbp, 2–4 kbp	-
Other	Also single	Color coding	-	High error rate

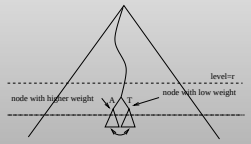
Total input data size: 50000 Mbp

Error Correction

Remove sequencing errors by aligning the reads with each other

Hybrid SHREC

- Based on SHREC by Schröder et al.
- Build a suffix trie of the read set.
- Correct low weight nodes in the trie by comparing to siblings
- Support for simultaneous correction of color coded and base coded reads



L. Salmela: Correction of sequencing errors in a mixed set of reads. *Bioinformatics* 26:10(1284–1290), 2010. (Award for best paper submitted to HiTSeq 2010).

Coral

- Build multiple alignments of reads that share k-mers
- Correct reads based on these multiple alignments
- Sequencing error model can be specified by setting gap penalty and mismatch penalty for multiple alignments

```
GTAA-GTTGAACCCTTA
  AA A GTTGAACCCTTACC
      GTTGAACC-TTACCCGG
         GA C CCTTACCCGGTTCA
```

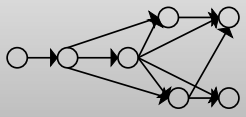
L. Salmela and J. Schröder: Correcting errors in short reads by multiple alignments. *Bioinformatics* 27(11):1455–1461, 2011. (Also in HiTSeq 2011).

Overlap Computation

Find suffix-prefix overlaps between reads. Represent the overlaps in an overlap graph.

Overlap Tool

- Supports mismatches and indels in the overlaps
- Based on *Burrows-Wheeler transform, backward backtracking* (Lam et al. 2008) and *suffix filters* (Kärkkäinen et al. 2008)
- Easy to parallelize
- Scales up to millions of reads



N. Välimäki, S. Ladra, and V. Mäkinen: Approximate all-pairs suffix/prefix overlaps. *Information & Computation* 213:49–58, 2012 (CPM 2010 Special Issue).

Contig Assembly

Report paths in the overlap graph as contigs, i.e. contiguous sequences.

MIP Scaffolder

- Partitioning the problem into smaller subproblems of *restricted* size
- Solving each subproblem as a mixed integer program (MIP)

L. Salmela, V. Mäkinen, N. Välimäki, J. Ylinen, and E. Ukkonen: Fast scaffolding with small independent mixed integer programs. *Bioinformatics* 27:23(3259–3265), 2011.

Scaffolding

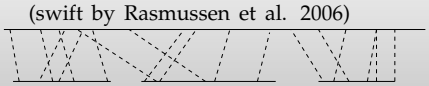
Mate pairs give links between contigs. Remove minimum number of mate pairs so that the remaining ones are consistent.

Superscaffolder

- Break chimeric scaffolds (assignment to several chromosomes)
- Find paths based on mate pair links between scaffolds in the same chromosome
- Remove ambiguous connections (manually or automatically)

Validation with ESTs

- Align ESTs against scaffolds:
  - Find local maximal approximate matches (swift by Rasmussen et al. 2006)



  - Produce maximal colinear chains of the above matches (Abouelhoda 2007)
- Compute the coverage of ESTs

V. Mäkinen, L. Salmela, and J. Ylinen: Normalized N50 assembly metric using gap-restricted co-linear chaining. *BMC Bioinformatics* 13:255, 2012.

Gap Closing

Use paired end reads to fill the gaps between contigs.

Genetic map (Chromosome assignment for some scaffolds)

(Error corrected) Mate pairs

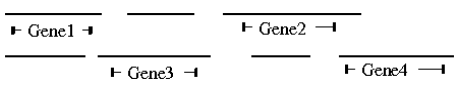
Superscaffolding

Use mate pairs and genetic map as a guide to connect scaffolds in the same chromosome.

Validation

Genetic map, Map ESTs to scaffolds,...

Annotation



Acknowledgements

Jan Schröder, Simon J. Puglisi, Ilkka Hanski, Rainer Lehtonen, Virpi Ahola, Mikko Frilander, Lars Paulin, Petri Auvinen, Panu Somervuo, Liisa Holm, Patrik Koskinen, Jussi Nokso-Koivisto, Pasi Rastas

PREDICTING QUANTITATIVE BINDING INTERACTIONS BETWEEN DRUG COMPOUNDS AND PROTEIN KINASES

Anna Cichonska, Jing Tang, Tero Aittokallio, Juho Rousu

FIMM
Institute for Molecular Medicine Finland
Nordic EMBL Partnership for Molecular Medicine

HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

Protein kinases constitute key regulators of cancer survival pathways. Effective inhibitors of these proteins are being designed. However, determining interactions between drug compounds and their molecular targets experimentally is time consuming and expensive. Various computational methods have been developed to facilitate this process. Among them, similarity-based machine learning methods are considered as state-of-the-art approaches. The assumption is that similar compounds are likely to interact with similar targets. Similarities between drugs are typically being computed based on their chemical structures and similarities between targets are being obtained by amino acid sequence alignments. In many applications, it is important to focus on predicting quantitative binding affinities rather than binary values since molecular interactions are not simple on-off relationships.

AIMS

1) To establish an approach for predicting missing drug-target interaction affinities in Metz et al. data set.
2) To determine the most meaningful metric for evaluating quantitative drug-kinase interaction predictions.

DATA SETS

Two data sets from the large-scale studies of selectivity profiles for kinase inhibitors were used. Inhibition constants, K_i, were measured in the Metz et al. (2011) survey, while dissociation constants, K_D, were used in the Davis et al. (2011) analysis. Both measurements reflect how tightly a compound binds to a target: a low value indicates an interaction. The overlap between the two studies comprises 24 compounds x 155 targets. Metz et al. data are very sparse – 47% of possible drug-target interaction affinities are missing (Fig. 1-3).

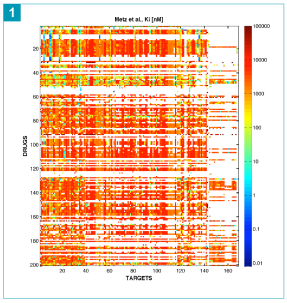


FIG. 1: Heat map representing Metz et al. data. White points indicate **missing K_i values**.

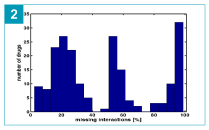


FIG. 2: Histogram showing the number of unmeasured interactions per drug in Metz et al. data.

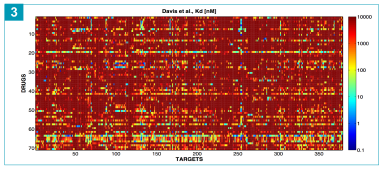


FIG. 3: Heat map representing Davis et al. data. The threshold for an interaction was set separately for each drug: 50 x min(K_D).

METHODS

Kronecker RLS – machine learning algorithm based on regularized least squares regression with Kronecker kernels (Tapio Pahikkala, Antti Airola). The method uses a product of drug Φ_D and target Φ_T kernels. Matrices are combined into a larger kernel that directly relates D-T pairs:

$$K((d_j,t_j),(d_k,t_l)) = \phi_D(d_j,d_k)\phi_T(t_j,t_l) = \phi_D\phi_T$$

	DRUGS (D)	TARGETS (T)
Features	**Tanimoto** kernel based on the size of common 2D/3D substructures similarities	• Normalized **Smith-Waterman** score (SW) • **GTG** (Global Trace Graph) features – conserved amino acids
Features representation	Kernel similarity matrix ***D-K*/*T-K***: given molecules VS given molecules	
	D-PubChem: given D vs 4 000 D (PubChem) S -> SS^T	***T-UniProt***: given T vs ~20 000 T (UniProt) S -> SS^T
Read-outs	$\log_{10}(K_i)$ or $\log_{10}(K_D)$	

Drug-kinase interaction prediction evaluation metrics

- Relative Absolute Error (RAE); the lower RAE, the better the prediction.
- F1 Score – harmonic mean of Sensitivity and Positive Predictive Values; the higher F1 score, the better the prediction.
- City Block (CB) distance from the point [0,1] on the Sensitivity vs 1-Specificity plot; the lower the distance, the better the prediction.

$$RAE = \frac{1}{\#\text{imputed values}} \sum \frac{|y-\hat{y}|}{f(y)}$$

$$f(y) = \begin{cases} |y|, & \text{if } |y| > \varepsilon \\ \varepsilon, & \text{if } |y| \le \varepsilon \end{cases}$$

y - original interaction value
$\hat{y}$ - predicted interaction value
ε - threshold for an interaction

RESULTS: determining the best features for drugs and targets, Davis et al. data

Davis et al. data were used in order to test the predictive performance of the Kronecker RLS algorithm on the quantitative data set and choose the best features for drugs and targets that will be used in the Metz et al. missing data imputation.

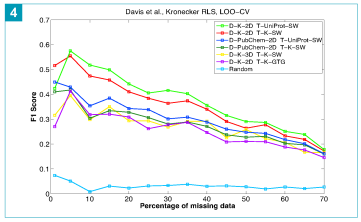


FIG. 4: F1 score, results of LOO-CV. Different percentages of the data were artificially removed from the observed K_D matrix. **The more missing data, the worse the prediction.**

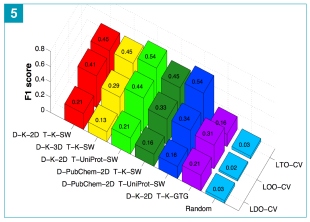
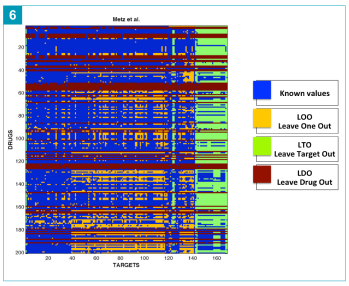


FIG. 5: F1 score, comparison of the results of LOO-CV, LTO-CV and LDO-CV. **Predicting targets for a new drug (LDO) is the hardest experimental setting in comparison to LOO and LTO.**

1	LOO-CV 20% missing data			RANK			
	CB dist	F1	RAE	CB	F1	RAE	Σ
D-K-2D T-K-SW	0.61	0.41	1.99	3	2	1	6
D-K-3D T-K-SW	0.71	0.29	3.63	5	6	6	17
D-K-2D T-UniProt-SW	0.58	0.44	2.02	1	1	2	4
D-PubChem-2D T-K-SW	0.61	0.33	3.32	3	4	5	12
D-PubChem-2D T-UniProt-SW	0.59	0.34	3.22	2	3	3	8
D-K-2D T-K-GTG	0.70	0.31	3.30	4	5	4	13
Random	1.01	0.02	13.74	6	7	7	20

TABLE 1: LOO-CV, ranking the values of each evaluation metric. The best features are considered as the ones having the lowest sum of ranks Σ. **Using ranking approach is more robust than relying on a single metric (aim #2).**

RESULTS: imputing missing D-T interactions in Metz et al. data



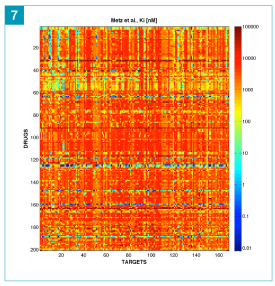


FIG. 6: Metz et al. data, imputation settings. LDO – more than 70% of the interactions affinities missing for a drug, LTO – more than 70% of the data missing for a target.

FIG. 7: Metz et al. data, heat map representing **original and imputed K_i values, aim #1** (Kronecker RLS, the best features were used: *D-K-2D*, *T-UniProt-SW*).

CONCLUSIONS

- Effective in silico drug-target interactions prediction methods are needed to support experimental analysis.
- Kronecker RLS algorithm allowed us to impute the missing data present in the Metz et al. study.
- Applying $\log_{10}$ transformation on the drug-target affinity values (K_i, K_D) works well because of the data scaling.
- Defining similarities between targets based on extended targets' profiles (*T-UniProt-SW*) and using them as features helped to achieve the best prediction performance.
- **Future directions:** applying other machine learning algorithms e.g. Kernel-Mapping Recommender system; utilizing 3D structural information for targets.

FIMM - Institute for Molecular Medicine Finland
P.O. Box 20, FI-00014 University of Helsinki, Finland
Biomedicum Helsinki 2U, Tukholmankatu 8, 00290 Helsinki, Finland

HIIT - Helsinki Institute for Information Technology
PO Box 15600, 00076 Aalto, Finland

Supervised and unsupervised biological network inference from multiple 'omics data

Aalto University
School of Science

Jana Kludas, Fitsum Tamene, Juho Rousu
{jana.kludas, fitsum.tamene, juho.rousu}@aalto.fi
Helsinki Institute for Information Technology, Aalto University, Finland

HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

INTRODUCTION

- **Protein-protein interactions (PPI)**: important for system-level understanding of biological processes
- **BIOLEDGE** project: BIO knowLEDGe Extractor and Modeler for Protein Production, focus on **secretion proteins**
- target species: **Saccharomyces cerevisiae**, Pichia pastoris, Trichoderma reesei

RESEARCH GOALS

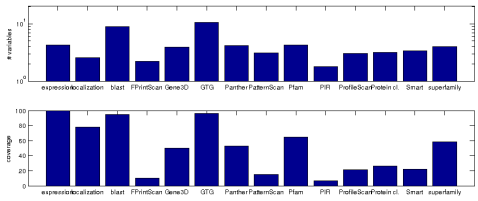
- (I) investigate the descriptive power of different features extracted from the protein sequences and genes
- (II) test 3 methods for graph inference: classification based on local modeling, classification based on global modeling, unsupervised graph inference based on expression data
- (III) overlay resulting networks

DATA SETS

Secretion Model by *Feizi, Nielson et al. Genome-scale modeling of Protein Secretory Machinery in Yeast, PLOS (2013)*

- Network of the components of the yeast (*S. cerevisiae*) secretory pathway
- 161 Proteins
- 50363 variables
- represents an undirected graph

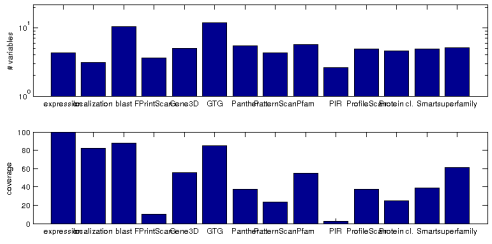
Figure 1: Number of variables per data source and their coverage.



KEGG Pathway

- Genome scale metabolic network in *S. cerevisiae* from KEGG
- 1335 Proteins
- 200317 variables
- represents a directed graph

Figure 2: Number of variables per data source and their coverage.



- **Sparse, High-Dimensional, Few Instances** -

PROTEIN-PROTEIN-INTERACTION (PPI) PREDICTION

Given a set of proteins $V = (v_1, ..., v_n)$,
a set of feature vectors $\Phi(v_1), ..., \Phi(v_n) \in \Re_p$,
a set of known interactions $S = ((e_1, y_1), ..., (e_m, y_m))$
as pairs of vertices: $e_i \in V \times V$ with $y_i = [1; -1]$.

INFERENCE WITH LOCAL MODELS

1. choose a seed vertex $v_{seed} \in V$
2. create local training set
3. feature selection based on confidence of feature-label pairs
4. train SVM on the local training set
5. predict label of any vertex that has no label
6. repeat step 1.-6. for each vertex $v_{seed} \in V$
7. combine the predicted edges

INFERENCE WITH GLOBAL MODELS

1. define representation of the protein's attributes for pairs of proteins
2. binary classification problem over pairs of vertices

* Kernels: direct Sum, direct Product, tensor product pairwise kernel, maximum tensor product pairwise kernel (MaxK), metric learning pairwise kernel (MLPK)

(both methods from *Vert J.P.: Reconstruction of Biological Networks by Supervised Machine Learning Approaches. Wiley, pp. 163-188 (2010)*)

UNSUPERVISED INFERENCE

* based on estimating the inverse covariance matrix of microarray data ie. *yeast2* dataset
* methods: partial correlations (qp-graph *de la Fuente et al: Discovery of meaningful associations in genomic data using partial correlation coefficients. Bioinformatics Vol. 20 no. 18 (2004)*), context likelihood of relatedness (CLR) algorithm (*Faith et al. Large-Scale Mapping and Validation of Escherichia coli Transcriptional Regulation from a Compendium of Expression Profiles, PLOS, 2007*)

PPI PREDICTION RESULTS FOR DIFFERENT 'OMICS DATA

Figure 3: Local modeling of Secretion data

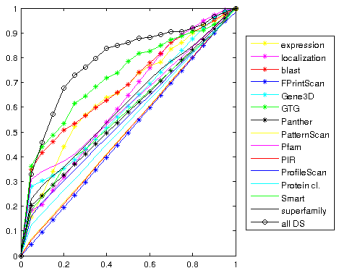


Figure 4: Global modeling of Secretion data

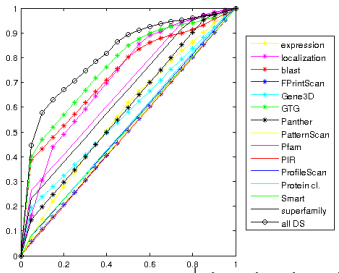


Figure 5: Local modeling of KEGG pathway data

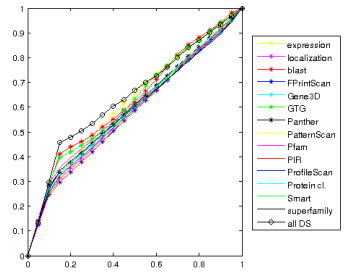
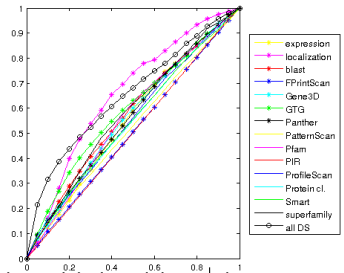


Figure 6: Global modeling of KEGG pathway data



	best local model	best global model	best unsupervised
Secretion Model	linear SVM $AUC = 0.784$	MaxK $AUC = \mathbf{0.812}$	CLR $AUC = 0.638$
KEGG pathway	FS, linear SVM $AUC = \mathbf{0.685}$	MLPK $AUC = 0.654$	QP-15 $AUC = 0.649$

OVERLAYING RESULT NETWORKS

Figure 7: Secretion data

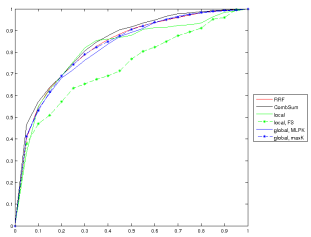
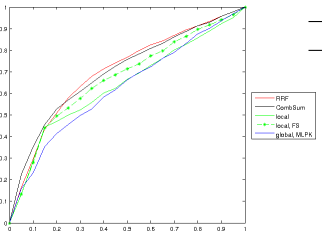


Figure 8: KEGG pathway data



best overlay:	CombSum
Secretion Model	$AUC = \mathbf{0.843}$
KEGG pathway	$AUC = \mathbf{0.703}$

- integration of multiple data sources and overlaying of different result networks increases the prediction accuracy

CONCLUSIONS

Drawbacks of current approaches:

- local modeling does not scale well - running time linear in number of proteins
- global modeling has large memory requirements - quadratic in number of proteins
- sequence features like Blast and GTG are more informative than genomic features such as expression data for biological network prediction
- undirected graphs are easier to predict than directed ones

ACKNOWLEDGEMENTS

The work was financially supported by the BIOLEDGE project (FP7-KBBE-289126), Academy of Finland grant 118653 (ALGODAN), and in part by the IST Programme of the European Community under the PASCAL2 Network of Excellence, ICT-2007-216886.

BIOLEDGE

Molecular Fingerprint Prediction *with* Multiple Kernel Learning

Aalto University
School of Science

HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

Huibin Shen[1,2], **Kai Dührkop**[3], **Sebastian Böcker**[3] **and Juho Rousu**[1,2]
[1,2] Aalto University and HIIT, Finland, [3]Friedrich Schiller University Jena, Germany

1 Introduction

Metabolite identification from tandem mass spectrometric measurements (MS/MS or MS^2) is a major problem encountered in several real-life applications. This task has been approached independently through machine learning [1] and fragmentation tree method [2]. Here we present work that combines the two research veins through Multiple Kernel Learning (MKL) as shown in Figure 1.

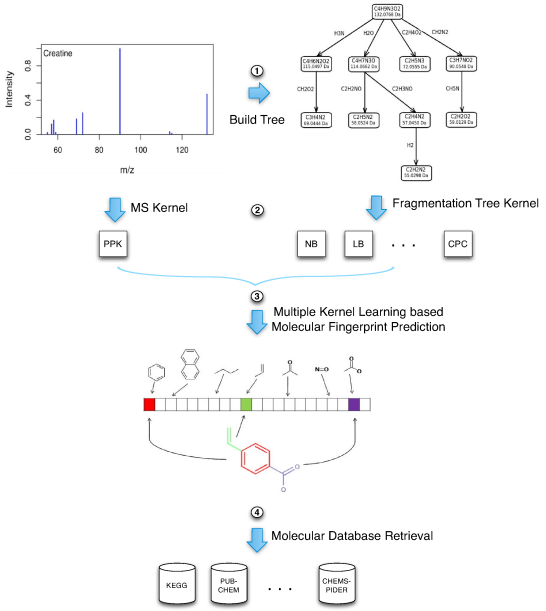


Figure 1: The machine learning framework for metabolite identification.

2 Kernels

Probability product kernel (PPK) The probability product kernel [3] approach models the spectrum $x = (x_1, \ldots, x_\ell)$ as a mixture $p_x = \frac{1}{\ell}\sum_{k=1}^{\ell} p_{x_k}$ of Gaussians p_{x_k} with mean at the observed mass and intensity values (μ_k, ι_k) and standard deviations σ_μ and σ_ι are estimated from the training data, same for all spectra. The probability product kernel between two spectra is then given by $K_{PPK}(x,x') = \frac{1}{\ell_x \ell_{x'}}\sum_{k,k'}\int_{\mu,\iota} p_{x_k}(\mu,\iota)\, p_{x_{k'}}(\mu,\iota)\, d\mu d\iota$.

Fragmentation tree kernels Define a fragmentation tree as $T = \{V, E\}$ with node set V, edge set E, root R and pseudo-edge (root to non-root node) set as $\mathscr{E}$. $\iota(v,V)$, $\iota(e,E)$, and $\iota(\varepsilon,\mathscr{E})$ are the intensities of nodes, terminal nodes of edges and terminal nodes of the pseudo-edges, respectively. $N(e,E)$ denotes the count of edge e in E. Given a mass spectrum x, $T(x) = \{V(x), E(x), \mathscr{E}(x)\}$ denotes the most likely fragmentation tree for that spectrum.

Linear kernels are computed over the features in Table 1:

Table 1: Features defined on fragmentation tree.

Features	LB	LC	LI	RLB	RLI	NB	NI
Definition	$\mathbf{1}_{\{e\in E(x)\}}$	$N(e,E(x))$	$\iota(e,E(x))$	$\mathbf{1}_{\{\varepsilon\in\mathscr{E}(x)\}}$	$\iota(\varepsilon,\mathscr{E}(x))$	$\mathbf{1}_{\{v\in V(x)\}}$	$\iota(v,V(x))$

Kernels count the substructures of fragmentation trees can be computed efficiently by dynamic programming. They include counting common path (CPC), path of length 2 (CP2), path with nodes intensities (CPI) and subtree (CSC).

3 Multiple kernel learning

Multiple kernel learning seeks a linear, convex or even non-linear combination of the kernels. A set of kernels $\boldsymbol{K}_N = \{\boldsymbol{K}_i | i = 1, \ldots, n\}$, $\boldsymbol{K}_i \in \mathbb{R}^{m\times m}$ is defined above and $\boldsymbol{K}_Y(i,i') = y_i y_i'$ is the target kernel.

UNIMKL: As a baseline MKL approach, the uniform combination of kernels (UNIMKL) is used: $\boldsymbol{K}_{\text{UNIMKL}}(x,x') = \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{K}_i(x,x')$.

ALIGN: This method [4] uses the centered kernel-target alignment score to weight the base kernels by

$$\mu_i = \frac{\langle \tilde{\boldsymbol{K}}_i, \tilde{\boldsymbol{K}}_Y\rangle_F}{\|\tilde{\boldsymbol{K}}_i\|_F \|\tilde{\boldsymbol{K}}_Y\|_F}, \quad \forall i = 1 \ldots n$$

where $\tilde{\boldsymbol{K}}$ is the kernel after centering operation.

ALIGNF: This approach [4] seeks a convex combination of the kernels to maximize the alignment score.

QCMKL: This method [5] extends the kernel space by taking elementwise product of the kernels and searches the best convex combination via semidefinite programming (Lanckriet, 2004).

ℓ_p-MKL: ℓ_p-norm MKL [6] regularizes the kernel weights by general ℓ_p-norms.

4 Results and discussion

Five fold cross validation was performed on a data set with 998 compounds. The Table 2 shows the NB kernel achieves the best performance among all the individual kernels. The ℓ_3-MKL is two percent better than uniform combination and 4 percent better than the NB kernel in accuracy. The weights (Figure 2, left) learned by these MKL algorithms do not agree in general but consistent in some cases. The improvement in fingerprint prediction can be transferred to metabolite identification directly (Figure 2, right).

Table 2: Cross validation performance for each kernel and the MKL algorithms.

	LB	LC	LI	RLB	RLI	NB	NI	CPC	CP2	CPI	CSC	PPK
Acc	78.8	78.4	77.0	80.9	76.8	**81.2**	79.8	79.4	78.0	72.1	73.2	75.6
F1	50.8	47.1	46.3	55.3	44.5	**57.6**	54.1	50.0	49.6	25.7	29.6	31.7
	UNI	ALIGN	ALIGNF	QCMKL	ℓ_2MKL	ℓ_3MKL	ℓ_4MKL	ℓ_5MKL				
Acc	83.2	83.4	83.8	84.3	84.5	**85.2**	85.1	85.0				
F1	59.5	60.6	63.5	63.0	64.1	67.9	**68.1**	**68.1**				

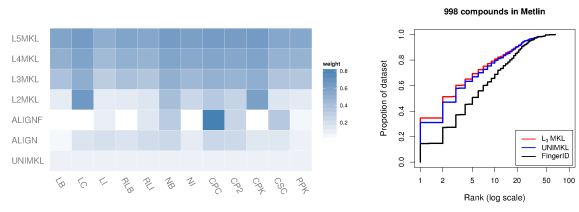


Figure 2: Weights for kernels (left) and metabolite identification result (right).

References

[1] Heinonen, M., Shen, H., Zamboni, N., Rousu, J.: "Metabolite identification and molecular fingerprint prediction through machine learning", *Bioinfomatics*, 28(18), 2333-2341, 2012

[2] Böcker, S., Rasche, F.: "Towards de novo identification of metabolites by analyzing tandem mass spectra", *Bioinformatics*, 24(16), 49–55, 2008

[3] Jebara, T., Kondor, R., Howard, A.: "Probability product kernels", *Journal of Machine Learning Research*, 5, 819–844, 2004

[4] Cortes, C., Mohri, M., Rostamizadeh, A.: "Algorithms for learning kernels based on centered alignment", *Journal of Machine Learning Research*, 13(1), 795–828, 2012

[5] Li, J., Sun, S.: "Nonlinear combination of multiple kernels for support vector machines", *International Conference on Pattern Recognition*, 2889–2892, 2010

[6] Kloft, M., Brefeld, U., Sonnenburg, S., Zien, A.: "ℓ_p-norm multiple kernel learning", *Journal of Machine Learning Research*, 12, 953–997, 2011

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

SYSCOL SYSTEMS BIOLOGY OF COLORECTAL CANCER

ALGODAN Algorithmic Data Analysis

MODELING AND PREDICTING REGULATORY AREAS

Jarkko Toivonen and Esko Ukkonen, Department of Computer Science

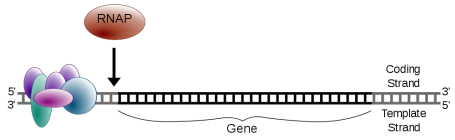
REGULATION OF GENES

The basic question is why gene expression differs between cells of single organism even though the cells contain the same DNA.
What affects the gene expression of a cell?

- Environmental condition
- Cell type
- The stage of development of an organism

What mechanism regulates the expression of genes?

- A *promoter* is an area in DNA close to the beginning of a gene. Transcription of a gene starts here.
- Certain proteins, called *transcription factors* (TF), can regulate the transcription of the gene by binding to its promoter area.



A MODEL FOR A BINDING SITE

Binding sites of transcription factors

- In order to understand how the regulatory system works, it is important to be able to describe and predict the binding sites of transcription factors in the genome
- A model that describes the binding sites where the TF prefers to bind is called *motif*, which can be represented, for example, by
  - A *consensus sequence* of a TF is the DNA sequence with the highest binding affinity to the TF
  - Regular expression (like ACG[GC]TT)
  - *Position Weight Matrix* (PWM) and its *sequence logo*, shown on left

DATA

The SELEX procedure (*Systematic evolution of ligands by exponential enrichment*) is a high-throughput *in vitro* method for selecting DNA sequences according to the binding affinity of the TF to the sequence.
From this dataset we can learn a motif model for the transcription factor in question.
Why use SELEX?

- Lots of TF bound sequences are produced which enable high precision motifs
- Fast and relatively inexpensive

LEARNING A PWM FROM SELEX DATA

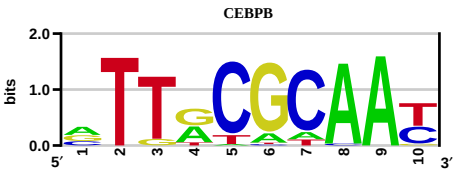
Using the SELEX data

- The SELEX procedure results in a set of fixed length sequences that were bound by the transcription factor
- Sequences are fed to a motif finding program which produces an alignment of the binding sites
- An example of counts from the aligment of the SELEX experiment with the ERG transcription factor

	1	2	3	4	5	6	7	8	9
A	164	22	23	0	0	164	164	98	6
C	10	164	164	0	0	1	1	9	42
G	37	23	0	164	164	0	1	164	21
T	31	3	0	0	1	1	40	2	164

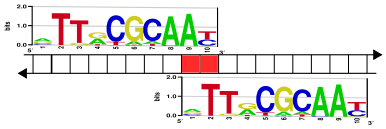
- These counts are then normalized column-wise, resulting in a multinomial distribution in each of the columns. This matrix can be visualised as a sequence logo.

An example of a PWM logo for the CEBPB factor:

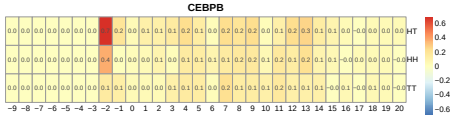


CO-OPERATIVE BINDING (COB) MODEL

Distances between closely bound transcription factors and their relative orientation can affect the strength of *co-operative binding* (COB) of two TFs. The case Head-to-tail (HT) orientation and distance −2 is illustrated below for factor CEBPB



Visualisation of COB model:



The value of co-operative binding in each cell is computed using the count observed in the SELEX data and the expected count in similar but random background of the case corresponding to the cell:

$$\log_2 \frac{\text{observed count}}{\text{expected count}}$$

MODEL FOR REGULATORY AREAS

We have created a model for predicting putative regulatory clusters called *scanner*. The model comprises of the following parts:

- PWMs describing the binding sites of transcription factors
- COB models describing the interaction between transcription factor pairs
- A dinucleotide model that describes the affinity of a nucleosome to DNA
  - Nucleosomes pack DNA and therfore affect the availability of the underlying DNA for TF binding

The clusters are found using dynamic programming that searches chains of TF binding sites

- The validity of the model can be tested with *in vivo* data, like ChIP-seq
- For n TFs we need n PWM and n^2 COB models

APPLICATION IN CANCER RESEARCH

Even though understanding of the regulatory system is important in itself, still the main objective is cancer research.

- Oncogenes promote cell growth and reproduction
- Tumor suppressor genes inhibit cell division and survival
- Mutations in the DNA can affect the expression of these genes. This can result in unrestricted growth, i.e. cancer
- The scanner can be used to predict the effect the mutations have on expression

[1] A. Jolma, T. Kivioja, J. Toivonen, et al. Multiplexed massively parallel SELEX for characterization of human transcription factor binding specificities. *Genome Res.*, 20(6):861–873, Jun 2010.

[2] A. Jolma, J. Yan, T. Whitington, J. Toivonen, et al. DNA-binding specificities of human transcription factors. *Cell*, 152(1–2):327–339, 2013.

This is joint work with Arttu Jolma, Teemu Kivioja, Pasi Rastas, Mikko Sillanpää, Jussi Taipale and Esko Ukkonen.

GEOSPATIAL DATA ANALYSIS AND PROCESSING

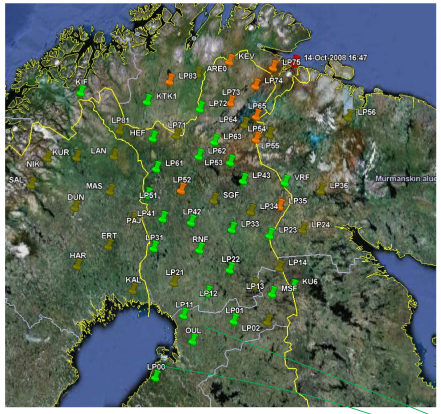
Mikko Nikkilä, Valentin Polishchuk, Topi Talvitie

Supported by: Academy of Finland Centre of Excellence ALGODAN, Research Funds of University of Helsinki grant 490092 "Computational Geometry group support",
Academy of Finland grant 1138520 "Algorithmic studies in applied geometry", Academy of Finland grant 261019 "Towards automatic (pre-)processing of seismological data"

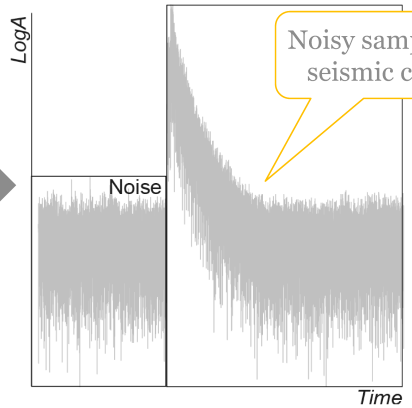
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

New shape reconstruction tool robust to outliers: k-order α-shape = k-hull + α-shape

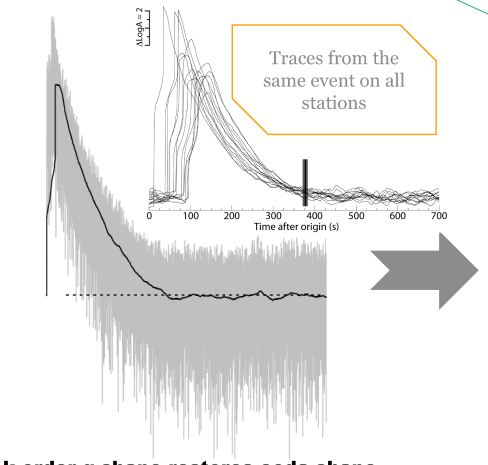
Application 1: Local magnitude scale for seismic activity in Fennoscandia



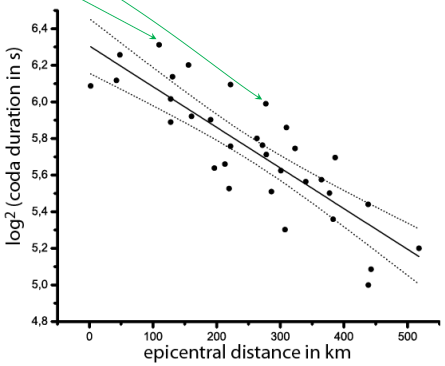
Events recorded at array of stations



Trace on a station = seismic noise + coda



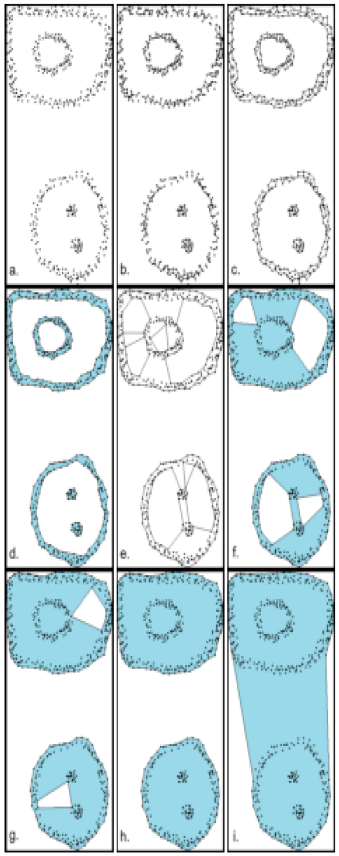
k-order α-shape restores coda shape



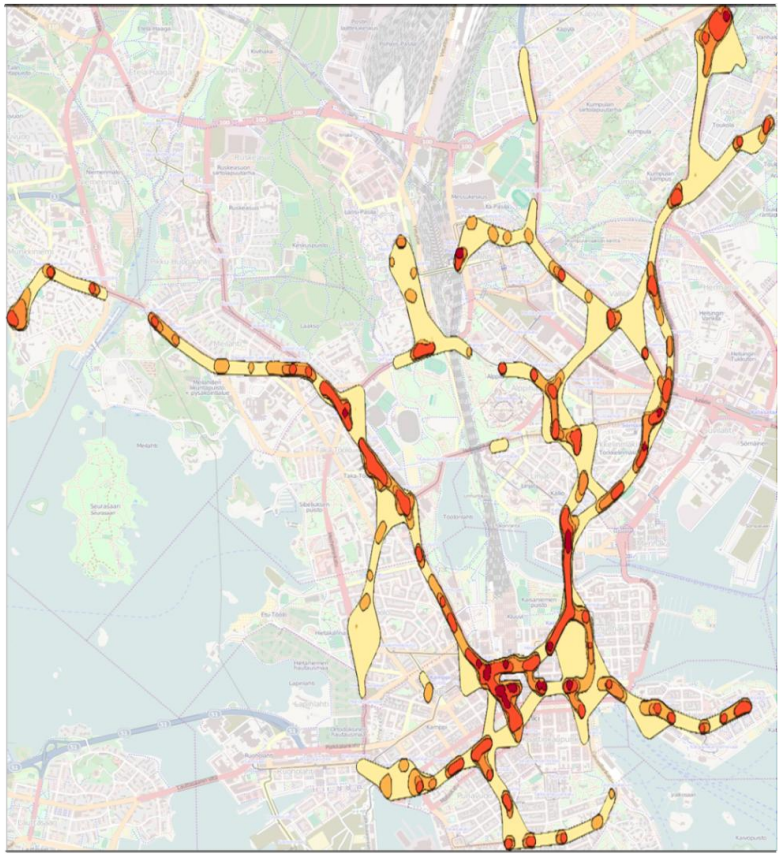
Local magnitude $M_L = a*\Delta + b*\log^2_{10}\tau + c$
Proportional to Richter Magnitude Scale

Collaboration with Institute of Dynamics of Geospheres, Russian Academy of Sciences. Published as:
Mikko Nikkilä, Valentin Polishchuk, Dmitry Krasnoshchekov.
Robust estimation of seismic coda shape.
Geophysical Journal International, online first February 3, 2014

Application 2: Patterns in (unplanned) stops of Helsinki public transport



Clustering with k-order α-shape: "meaningful" shapes (cycles) are shaded



Real-world data instantiation: where do buses have most frequent unplanned stops?

Collaboration with IBM Research Haifa Lab. Published as:
P. Bak, E. Packer, H. Ship, M. Nikkilä, V. Polishchuk.
Visual Analytics for Spatial Clustering: Using a Heuristic Approach for Guided Exploration.
IEEE Transactions on Visualization and Computer Graphics, 19(12):2179-2188, 2013

Multilabel Classification through Random Graph Ensemble

Aalto University
School of Science

Hongyu Su, Juho Rousu
{hongyu.su, juho.rousu}@aalto.fi
Helsinki Institute for Information Technology, Aalto University, Finland

HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

ABSTRACT

We present new methods for multilabel classification, relying on ensemble learning on a collection of random output graphs imposed on the multilabel and a kernel-based structured output learner as the base classifier. Diversity of base classifiers arises from the different random output structures, a different approach from boosting or bagging. In our experiments, the random graph ensembles are very competitive and robust, ranking first or second on most of the datasets.

ENSEMBLE ANALYSIS

We study the theoretical property of MAM ensemble by analyzing reconstruction error of compatibility score. Compatibility score for a fixed pair $(x, \mathbf{y})$ is

$$\psi(x,\mathbf{y}) = \sum_{e\in E} \psi_e(x,\mathbf{y}_e) = \sum_{j\in V} \psi_j(x,y_j).$$

Denote the $\psi^*(x,\mathbf{y})$ optimal compatibility score. Reconstruction error is given by the squared distance:

$$\Delta^R_{\mathsf{MAM}}(x,\mathbf{y}) = \left(\psi^*(x,\mathbf{y}) - \psi^{\mathsf{MAM}}(x,\mathbf{y})\right)^2$$
$$\Delta^R_I(x,\mathbf{y}) = \frac{1}{T}\sum_t \left(\psi^*(x,\mathbf{y}) - \psi^{(t)}(x,\mathbf{y})\right)^2.$$

THEOREM The reconstruction error of compatibility score distribution given by MAM ensemble $\Delta^R_{\mathsf{MAM}}(x,\mathbf{y})$ is guaranteed to be no greater than the average reconstruction error given by individual base learners $\Delta^R_I(x,\mathbf{y})$. In addition, the gap can be estimated as

$$\Delta^R_I(x,\mathbf{y}) - \Delta^R_{\mathsf{MAM}}(x,\mathbf{y}) = \mathbf{Var}_t\Big(\sum_{j\in V}\Psi_j(x,y_j)\Big) \geq 0.$$

The variance can be further expanded as

$$\mathbf{Var}\Big(\sum_{j\in V}\Psi_j(x,y_j)\Big) = \underbrace{\sum_{j\in V}\mathbf{Var}(\Psi_j(x,y_j))}_{diversity} + \underbrace{\sum_{\substack{p,q\in V,\\ p\neq q}}\mathbf{Cov}(\Psi_p(x,y_p),\Psi_q(x,y_q))}_{coherence}.$$

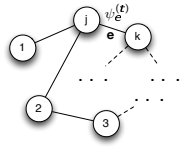
CONCLUSION

We have put forward new methods for multilabel classification, relying on ensemble learning on random output graphs. In our experiments, models thus created have favourable predictive performances on a heterogeneous collection of multilabel datasets. The theoretical analysis of the MAM ensemble highlights the covariance of the compatibility scores between the inputs and microlabels learned by the base learners as the quantity explaining the advantage of the ensemble prediction over the base learners. Our results indicate that structured output prediction methods can be successfully applied to problems where no prior known output structure exists, and thus widen the applicability of the structured output prediction.

MODELS

BASE LEARNER (MMCRF)

Can be seen to decompose into a set of "potential functions" $\Psi_E^{(t)}(x) = (\psi_e^{(t)}(x,\mathbf{u}_e))_{e\in E^{(t)},\mathbf{u}_e\in\mathcal{Y}_e}$



$\{\psi_e^{(t)}(x,++), \psi_e^{(t)}(x,+-), \psi_e^{(t)}(x,-+), \psi_e^{(t)}(x,--)\}$
Prediction is by $\hat{\mathbf{y}}(x) = \mathbf{argmax}_{\mathbf{y}\in\mathcal{Y}}\sum_e \psi_e(x,\mathbf{y}_e)$.

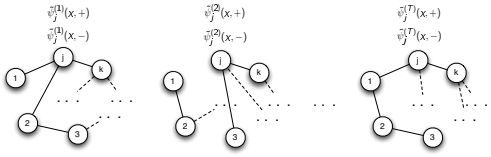
MAJORITY VOTING ENSEMBLE (MVE)

In MVE, the ensemble prediction or each microlabel is the most frequently appearing prediction among the base classifiers

$$F_j^{\mathsf{MVE}}(x) = \mathbf{argmax}_{y_j\in\mathcal{Y}_j}\left(\frac{1}{T}\sum_{i=1}^T \mathbf{1}_{\{F_j^{(t)}(x)=y_j\}}\right),$$

where $F^{(t)}(x) = (F_j^{(t)}(x))_{j=1}^k$ is the predicted multilabel in t'th base classifier.

AVERAGE OF MAX-MARGINALS (AMM)



Our goal is to infer for each microlabel u of each node j its *max-marginal*, that is, the maximum score of a multilabel that is consistent with $y_j = u_j$, $u_j \in \{+,-\}$

$$\tilde{\psi}_j(x,u_j) = \max_{\{\mathbf{y}\in\mathcal{Y}: y_j=u_j\}}\sum_e \psi_e(x,\mathbf{y}_e).$$

The ensemble prediction for each target is obtained by averaging the max-marginals of the base models and choosing the maximizing microlabel for the node:

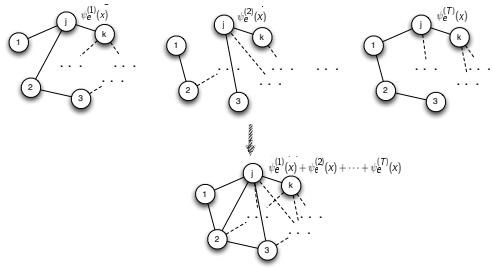
$$F_j^{\mathsf{AMM}}(x) = \mathbf{argmax}_{u_j\in\mathcal{Y}_j}\frac{1}{|T|}\sum_{t=1}^T \tilde{\psi}_{j,u_j}^{(t)}(x),$$

and the predicted multilabel is composed from the predicted microlabels

$$F^{\mathsf{AMM}}(x) = \left(F_j^{\mathsf{AMM}}(x)\right)_{j\in V}.$$

MAXIMUM AVERAGE MARGINALS (MAM)

Generate the **union graph** of the trees underlying the base models, with average edge labeling scores $\frac{1}{|T_e|}\sum_{t\in T(e)}\psi_{e,u}^{(t)}(x)$ (normalized by how many times an edge appears)



Inference on the union graph:

$$F^{\mathsf{MAM}}(x) = \mathbf{argmax}_{\mathbf{y}\in\mathcal{Y}}\sum_{e\in\cup_t E_t}\frac{1}{T}\sum_{t=1}^T \psi_e^{(t)}(x,\mathbf{y}_e)$$

Interpretation: ensemble prediction is the multilabel maximizing the average score over the base models.

EXPERIMENTAL RESULTS

Figure 1: Ensemble learning curve (microlabel accuracy) plotted as the size of ensemble. Average performance of base learner with random tree as output graph structure is denoted as horizontal dash line.

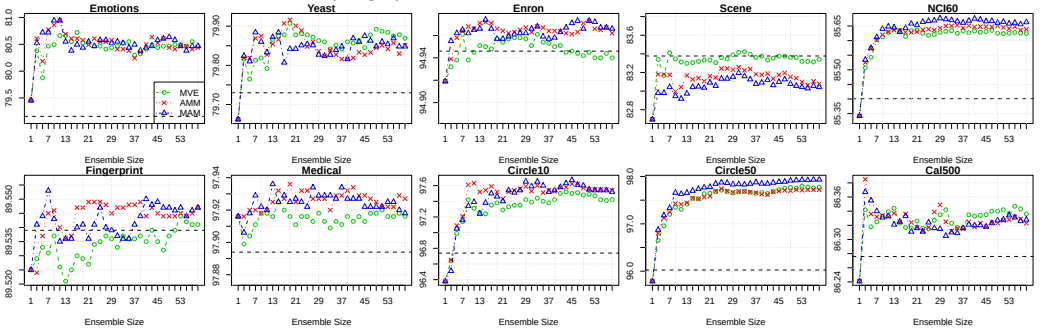


Table 1: Prediction performance by microlabel accuracy.

Dataset	Microlabel Accuracy					
	SVM	Bagging	AdaBoost	MTL	MMCRF	MAM
Emotions	77.3±1.9	74.1±1.8	76.8±1.6	*79.8±1.8*	79.2±0.9	**80.5±1.4**
Yeast	**80.0±0.6**	78.4±0.7	74.8±0.3	79.3±0.2	79.7±0.3	*79.9±0.4*
Scene	**90.2±0.3**	87.8±0.8	84.3±0.4	*88.4±0.6*	83.4±0.2	83.0±0.2
Enron	93.6±0.2	93.7±0.1	86.2±0.2	93.5±0.1	*94.9±0.1*	**95.0±0.2**
Cal500	**86.3±0.3**	86.0±0.2	74.9±0.4	86.2±0.2	**86.3±0.2**	**86.3±0.3**
FP	**89.7±0.2**	85.0±0.7	84.1±0.5	82.7±0.3	*89.5±0.3*	*89.5±0.8*
NCI60	84.7±0.7	79.5±0.8	79.3±1.0	84.0±1.1	*85.4±0.9*	**85.7±0.7**
Medical	97.4±0.1	97.4±0.1	91.4±0.3	97.4±0.1	**97.9±0.1**	**97.9±0.1**
Circle10	94.8±0.9	92.9±0.9	**98.0±0.4**	93.7±1.4	96.7±0.7	*97.5±0.3*
Circle50	94.1±0.3	91.7±0.3	*96.6±0.2*	93.8±0.7	96.0±0.1	**97.9±0.2**
@Top2	4	0	2	2	*5*	**9**

Table 2: Prediction performance by multilabel accuracy.

Dataset	Multilabel Accuracy					
	SVM	Bagging	AdaBoost	MTL	MMCRF	MAM
Emotions	21.2±3.4	20.9±2.6	23.8±2.3	25.5±3.5	*26.5±3.1*	**30.4±4.2**
Yeast	*14.0±1.8*	13.1±1.2	7.5±1.3	11.3±2.8	13.8±1.5	**14.0±0.6**
Scene	**52.8±1.0**	*46.5±2.5*	34.7±1.8	44.8±3.0	12.6±0.7	5.4±0.5
Enron	0.4±0.1	0.1±0.2	0.0±0.0	0.4±0.3	*11.7±1.2*	**12.1±1.0**
Cal500	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0	0.0±0.0
FP	**1.0±1.0**	0.0±0.0	0.0±0.0	0.0±0.0	*0.4±0.9*	*0.4±0.5*
NCI60	*43.1±1.3*	21.1±1.3	2.5±0.6	**47.0±1.4**	36.9±0.8	40.0±1.0
Medical	8.2±2.3	8.2±1.6	5.1±1.0	8.2±1.2	*35.9±2.1*	**36.9±4.6**
Circle10	69.1±4.0	64.8±3.2	**86.0±2.0**	66.8±3.4	75.2±5.6	*82.3±2.2*
Circle50	29.7±2.5	21.7±2.6	28.9±3.6	27.7±3.4	*30.8±1.9*	**53.8±2.2**
@Top2	5	2	2	2	*6*	**8**

ACKNOWLEDGEMENTS

The work was financially supported by Helsinki Doctoral Programme in Computer Science (Hecse), Academy of Finland grant 118653 (ALGODAN), and in part by the IST Programme of the European Community under the PASCAL2 Network of Excellence, ICT-2007-216886. This work only reflects the authors' views.

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

TREEDY: A HEURISTIC FOR COUNTING AND SAMPLING SUBSETS

Teppo Niinimäki, Mikko Koivisto

Consider a collection of weighted subsets of a ground set N. We present a tree-based greedy heuristic, Treedy, that for a given query subset Q of N and a tolerance d approximates the weighted sum over all subsets of Q within relative error d. It also enables approximate sampling of subset of Q proportionally to the weights within total variation distance d. Experimental results show that approximations yield dramatic savings in running time compared to exact computation, and that Treedy typically outperforms a previously proposed sorting-based heuristic.

INTRODUCTION

PROBLEM DEFINITION

Input: A downward closed collection $\mathcal{C}$ of subsets of a ground set N. Weights $w(S) \geq 0$ for $S \in \mathcal{C}$ and $w(S) = 0$ otherwise.

Query: Query set $Q \subseteq N$. Tolerance $d \geq 0$.

Counting problem:
Approximate

$$W(Q) = \sum_{S \subseteq Q} w(S)$$

within relative error d.

Sampling problem:
Draw a random subset $S \subseteq Q$ from a distribution within total variation distance d from $\Pr(S) = w(S)/W(Q)$.

APPLICATION: BAYESIAN NETWORK LEARNING

Order-MCMC [1] is a method for learning the structure of a Bayesian network. It

- samples node orderings $v_1 v_2 \cdots v_n$ from the posterior distribution

$$\Pr(v_1 v_2 \cdots v_n) = \prod_{i=1}^{n} W_i(\{v_1, \ldots, v_{i-1}\})$$

where $W_i(Q) = \sum_{S \subseteq Q} w_i(S)$ is a sum over possible parent sets of v_i

$\Rightarrow n$ subset counting queries

- optionally samples DAGs from orderings

$\Rightarrow n$ subset sampling queries

ALGORITHMS

"Collector algorithm" approach: Visit the subsets of Q that are in $\mathcal{C}$ (called *relevant* sets) and add up their weights until the sum is guaranteed to be within tolerance d.

ALGORITHM: EXACT

A baseline method that visits all relevant sets. Computes the sum exactly.

ALGORITHM: IDEAL

An idealized method that visits the minimum number of heaviest relevant sets to reach tolerance d. (Simulated in the experiments.)

ALGORITHM: SORTED

An improved version of the heuristic of Friedman and Koller for order-MCMC [1].

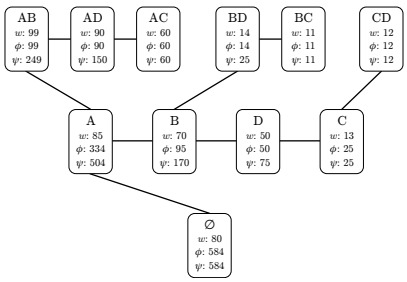
Preprocessing: Sorts the sets in $\mathcal{C}$ by weight starting from the heaviest set.

Per query: Traverses the sorted $\mathcal{C}$ until the weight of the remaining sets is small enough compared to accumulated weight.

ALGORITHM: TREEDY

A novel heuristic based on tree traversal.

Preprocessing: Builds a "greedy tree" with the sets in $\mathcal{C}$ as nodes. Computes weight potentials ϕ and aggregate potentials ψ.



Per query: Traverses the tree greedily w.r.t. ψ. Ignores irrelevant branches. Stops once the weight of remaining branches is small enough compared to accumulated weight.

FROM COUNTING TO SAMPLING

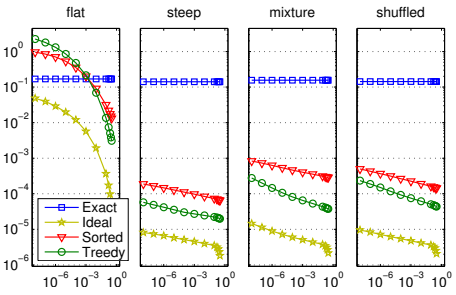
Sampling within total variation distance d to the exact distribution is possible by first visiting relevant sets up to tolerance d and then drawing samples from visited sets.

EXPERIMENTS

We measured the time (s) per subset counting query as a function of approximation tolerance d. Parameter $k \in \{4, 5\}$ was used to restrict the size of the subsets in $\mathcal{C}$.

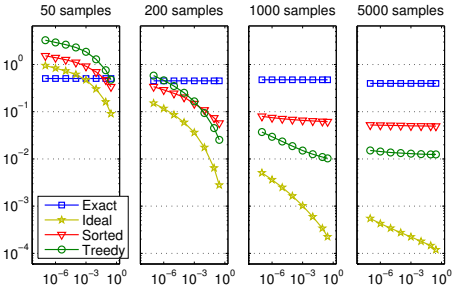
ARTIFICIAL INSTANCES

Runtimes for different types of artificial weight functions ($n = 60$, $k = 5$):

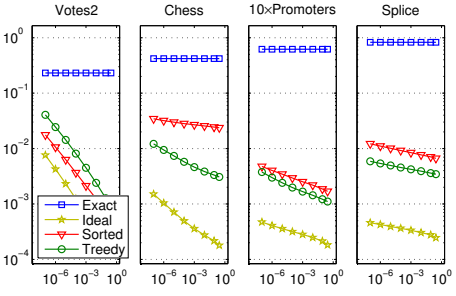


BAYESIAN NETWORK LEARNING

Runtime of order-MCMC for data from ALARM-network ($n = 37$, $k = 5$):



Runtime of order-MCMC for datasets from the UCI repository ($n \in [34, 61]$, $k \in \{4, 5\}$):



[1] N. Friedman and D. Koller. Being Bayesian about network structure: A Bayesian approach to structure discovery in Bayesian networks. Machine Learning, 50:95–125, 2003.

HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

Analysis of Environmental Proxies and Dendrochronological Series

HELSINKI INSTITUTE FOR INFORMATION TECHNOLOGY

Mikko Korpela and Jaakko Hollmén
{mikko.korpela, jaakko.hollmen}@aalto.fi
Aalto University School of Science, Department of Information and Computer Science
Helsinki Institute for Information Technology HIIT

Introduction

Direct temperature measurements are only available from the past few hundred years. Therefore, proxy measurements must be used. We study the use of different environmental proxy variables for temperature reconstruction. Differences in both the time coverage of the proxies (Fig. 1) and the temperature signal present in them pose a challenge to the recovery of reliable temperature records (Fig. 2).

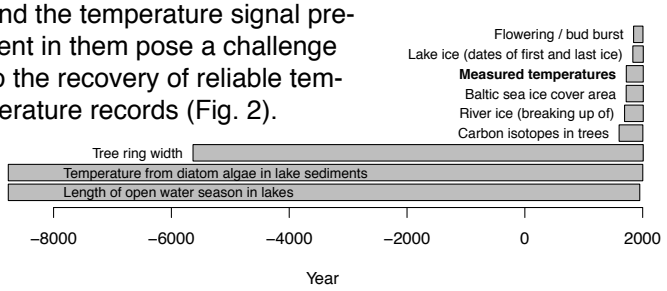


Fig. 1: Rough availability of different proxy measurements

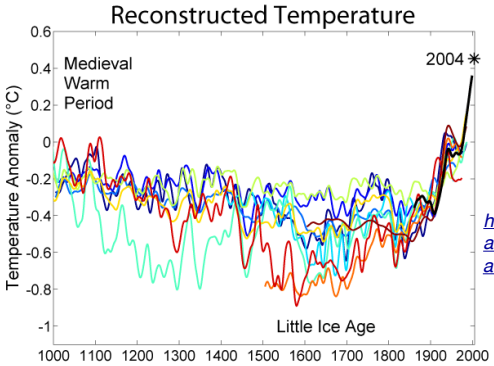


Fig. 2: Different temperature reconstructions. Image created by Robert A. Rohde / Global Warming Art. http://www.globalwarmingart.com/wiki/File:1000_Year_Temperature_Comparison.png

Environmental Proxy Selection Problem

• Identify the most informative proxy variables for reconstruction of temperature in Finland (Fig. 3)

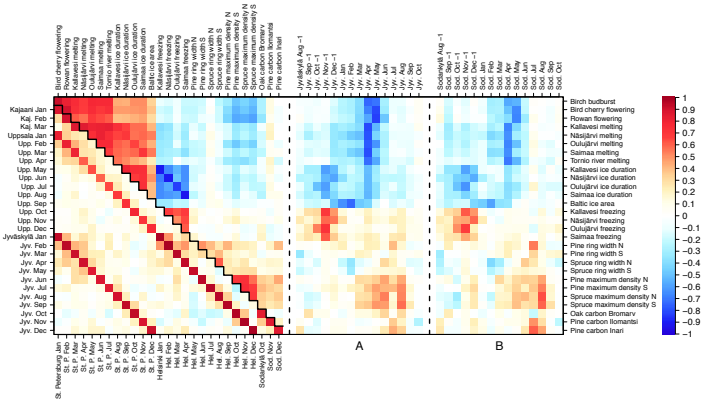


Fig. 3: Correlation of various temperature and proxy variables

• Different time of year or different geographic location ⇒ alternative set of good proxies

• With respect to finding solutions to input selection problems, we are working on an R version of the backward selection type algorithm SISAL [4].

- Extend [4] by exploring more states by branching
- Software package is ready
- Experiments still needed

The dplR package for R

The dendrochronology program library in R (dplR) [1] is an add-on package for the R Project for Statistical Computing. These are open source software.

We use the package for preprocessing of tree ring measurements and do active development to make it better suit our and the users' needs. Some of our contributions include:

- Improved performance
- Support for additional data formats (e.g. TRiDaS)
- Other new or borrowed functionality (Fig. 4, 5)

Contributing to dplR can also open possibilities for research collaboration [2].

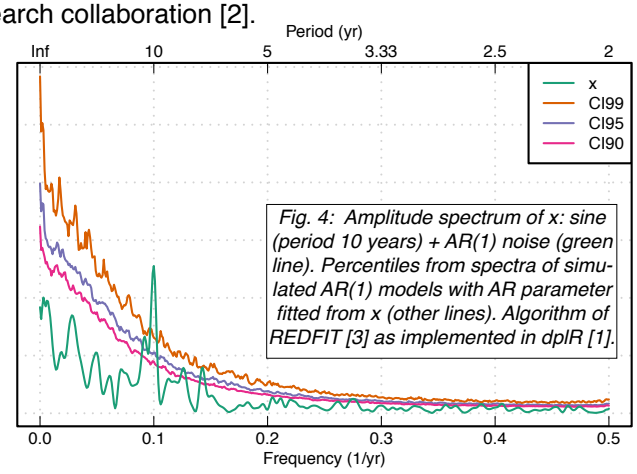


Fig. 4: Amplitude spectrum of x: sine (period 10 years) + AR(1) noise (green line). Percentiles from spectra of simulated AR(1) models with AR parameter fitted from x (other lines). Algorithm of REDFIT [3] as implemented in dplR [1].

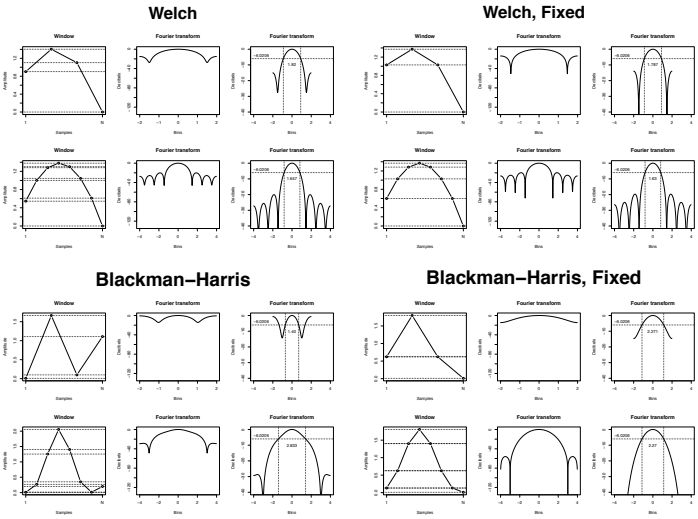


Fig. 5: Two types of sampling windows, small window sizes, with frequency response. Left: As used in REDFIT [3]. Right: Fixed (FFT symmetric) windows used in our implementation of the same algorithm, available in dplR [1].

References

[1] A. G. Bunn. A dendrochronology program library in R (dplR). *Dendrochronologia,* 26(2):115–124, 2008.

[2] A. G. Bunn, E. Jansma, M. Korpela, R. D. Westfall, J. Baldwin. Using simulations and data to evaluate mean sensitivity as a useful statistic in dendrochronology. *Dendrochronologia,* 31(3):250–254, 2013.

[3] M. Schulz, M. Mudelsee. REDFIT: estimating red-noise spectra directly from unevenly spaced paleoclimatic time series. *Computers & Geosciences,* 28(3):421–426, 2002.

[4] J. Tikka, J. Hollmén. Sequential Input Selection Algorithm for Long-term Prediction of Time Series. *Neurocomputing,* 71(13–15):2604–2615, 2008.

EXPANDABLE STRING REPRESENTATION FOR MUSIC FEATURES USING LIMITED-SIZE ALPHABETS

Simo Linkola, Lari Rasku, Teppo E. Ahonen

{slinkola,rasku,teahonen}@cs.helsinki.fi

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI

MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

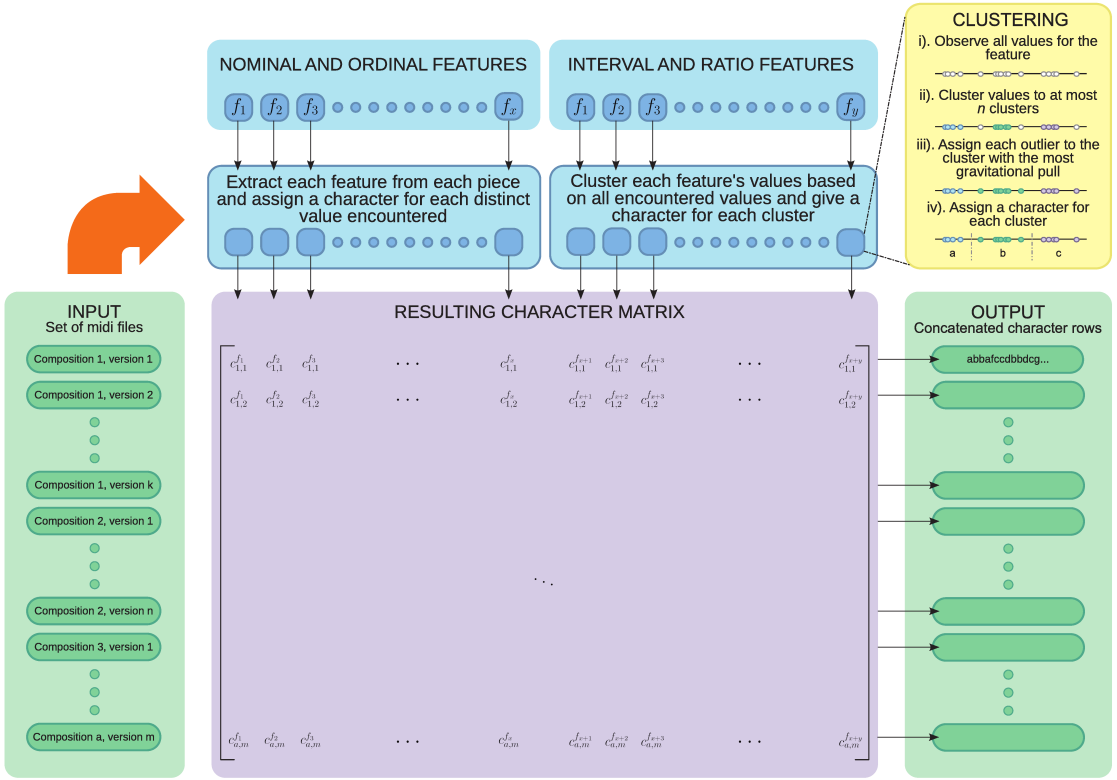
WE PROPOSE

an extandable string representation for symbolic polyphonic music, and apply it for measuring tonal similarity.

First, the data is processed into tempo- and key-invariant form. Then, time-invariant high-level features are extracted from the dataset, and the features are clustered adaptively to obtain an alphabet for each feature. Finally, the string for each piece of music is composed by concatenating the instances of the clustered features.

Measuring similarity between the string representations can be done with any similarity metric. We experimented with Hamming distance.

PROCESS FLOW



TO EVALUATE

our approach, we performed a retrieval experiment with a dataset of classical variations. The dataset consists of 17 sets of classical themes and their variations, with a total of 95 pieces of music.

	PREC	PREC@1
SKYLINE FEATURES	0.170	0.232
JSYMBOLIC FEATURES	0.525	0.674
ALL FEATURES	0.519	0.695
ALL FEATURES, KEY INVARIANCE	0.449	0.611

The evaluation suggests that the skyline features might not be usable alone, but in conjunction with jSymbolic features they provide slightly better results. Also, the features that are not dependent on key seem to contain more distinguishing power.

The work of Simo Linkola was funded by the Academy of Finland (the Finnish Centre of Excellence in Computational Inference Research COIN). Lari Rasku and Teppo E. Ahonen were supported by the Academy of Finland under grant number 118653 (CoE ALGODAN).

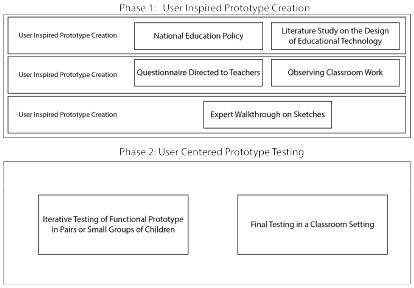
HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

POETRY ENGINE DESIGNING AN INTERACTIVE TOOL FOR POETRY CO-CREATION AT SCHOOL

Anna Kantosalo
Department of Computer Science
Helsinki Institute of Information Technology
Jukka M. Toivanen
Department of Computer Science
Helsinki Institute of Information Technology
Hannu Toivonen
Department of Computer Science
Helsinki Institute of Information Technology

The goal of the "Poetry Engine"-project is to develop a software tool based on computational poetry to help kids learn and practice creative language use. Ideally the design should empower pupils from different age groups to explore and compose poetic language in an interactive setting with a computational poetry engine. The tool is designed for real educational contexts in the Finnish comprehensive school. It is build on existing poetry engines and developed with user centered design methods.

METHODOLOGY



The design process uses user centered design methods, which are an established set of techniques for developing software to the needs and capabilities of it's users as well as the special properties of the context the software is used in. User centered methodology has previously been successfully applied in the development of serious applications for computational creativity by Richie et al. [1], who built a joke generating tool for children with special needs.

The final design will be powered by existing computational poetry engines developed by the Discovery Research Group at the University of Helsinki. These include the P.O.Eticus engines based on two different approaches: Generating poetry from a corpus of patterns via replacement [3] and with constraints [2].

The corpus based approach already allows for the user to define a topic for the poem. This topic is then used for finding semantically related content in a word association network. The syntax and the morphology of the poem are fetched from another corpus. The selection of the corpora offers some further possibility for user involvement in the initial stages of poetry creation.

The constraint based approach is based on the corpus approach and has two components, the specifier and the explorer. The specifier determines the syntactical and aesthetical rules of poem creation and the explorer composes poems according to these rules. The specifier component can take input from the user, or use other inspirational material for automatizing the input generation for the explorer.

To achieve optimal levels of user and machine co-creation, we will possibly need to develop fast ways of producing poetic fragments instead of full poems.

USER AND CONTEXT DETAILS

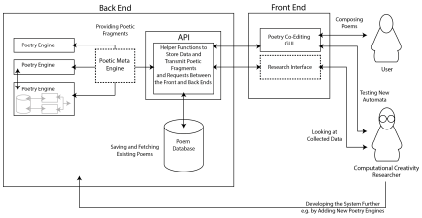
The first phase of the project revealed that teachers recognized poetry as way to promote creative and free expression in their pupils. The role of the teacher was also found to be important as a mediator in the use of any educational software. Teachers also think that the role of technology is to modernize and diversify learning, and to motivate and aid pupils. The need for quality material, especially for learning to write is evident.

The skills and interests of pupils wary a lot between individuals and age groups. Especially younger children have problems with basic tasks, such as saving their work, logging in into the software, and using the English language. Some may also struggle with basic reading and writing tasks in their native language. Some children may have difficulty in concentrating, or working in a group.

Hardware and software vary even within one class, but computers are mainly connected to the internet and online software is in frequent use. Computers are usually used alone or in pairs.

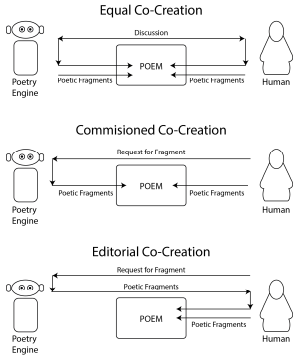
Important usability factors relate to offering simple writing mechanisms, enough support, using children's own vocabulary, and in overall building a clear and responsive interface. A visually pleasant feel was also promoted by the teachers.

THE PROTOTYPE

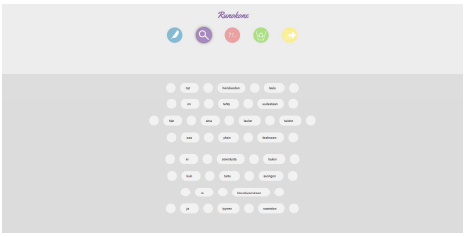


The system is developed as a web service to easily deliver it to multiple platforms without the need of installation. The back end uses the Python based Django framework with a database and the front GUI is developed with HTML5. Additionally the back end offers a python API for connecting poetry engines.

Notably there are at least two contributing creators in the system: the engine and the user. The API must therefore allow for communicating information between the two creators with respectively low latency. Additionally, the poetry engines may form a hierarchy, and communicate with the user through a higher level engine that orchestrates their activity. In an ideal situation the users and the engine would act as equal partners creating one poem. However as the user is more directly asking for the engine's help, the co-operation between them is more like a commission, or even editorial in nature.



The current prototype has a simple interface resembling fridge magnets: Words can be re-ordered by dragging and dropping, and new words can be added by clicking. Additionally, a few simple controls can be used to communicate with the engine asking for more material. Other controls include for example the possibility of exporting the poem outside the framework.



FUTURE WORK

The next step is to evaluate the design with actual users in different settings. The user evaluations offer also a possibility to look at the collaboration between the children and the poetry engine in more detail. Later on the system offers possibilities of expanding it's scope to include data gathering on it's use and peoples reactions to co-created poetry.

Naturally a part of the future work will be making the system widely available and disseminating information on it's use to teachers and pupils.

REFERENCES

[1] RITCHIE, G., MANURUNG, R., PAIN, H., WALLER, A., BLACK, R. AND O'MARA, D. 2007. A practical application of computational humour. In Proceedings of the 4th International Joint Conference on Computational Creativity, Anonymous , 91-98.

[2] TOIVANEN, J.M., JÄRVISALO, M. AND TOIVONEN, H. 2013. Harnessing Constraint Programming for Poetry Composition. Proceedings of the Fourth International Conference on Computational Creativity 160. .

[3] TOIVANEN, J., TOIVONEN, H., VALITUTTI, A. AND GROSS, O. 2012. Corpus-based generation of content and form in poetry. In Proceedings of the Third International Conference on Computational Creativity.

EFFICIENT AND SIMPLE ALGORITHMS FOR TIME-SCALED AND TIME-WARPED MUSIC SEARCH

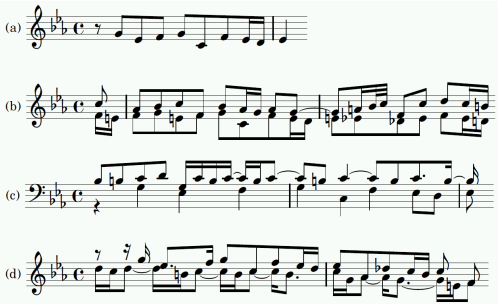
Antti Laaksonen, Department of Computer Science

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

MUSIC SEARCH

We present algorithms for finding occurrences of a given pattern in a polyphonic music database. Both the database and the pattern are given in symbolic form as a sequence of notes. Applications for polyphonic music search include music analysis and query by humming.

We consider three types of searches. In all searches we allow pattern transposition i.e. the pattern may be located in any key. In **exact search** we require that the tempo remains unchanged. In **time-scaled search** we allow constant scaling to the note lengths. In **time-warped search** we only require that the notes appear in the correct order.



For example, consider the opening theme from Bach's fugue (BWV 871) in *Das Wohltemperierte Klavier* in Staff (a). The theme appears more than twenty times throughout the fugue in exact, time-scaled and time-warped form. Staff (b) contains an exact occurrence, Staff (c) contains a time-scaled occurrence and Staff (d) contains a time-warped occurrence.

ALGORITHMS

Let n denote the number of notes in the database and m denote the number of notes in the pattern. The exact search problem can be solved using a simple $O(nm)$ algorithm [1]. The idea is to use m pointers to database notes. The first pointer goes through all possible beginning notes of a pattern occurrence. The other pointers correspond to the remaining pattern notes and follow the first pointer. We present new algorithms for time-scaled and time-warped search.

Time-scaled search

The previous algorithm [2] for time-scaled search uses precomputed lists of database note pairs and priority queues to track pattern occurrences that have a constant scaling factor. The algorithm works in $O(n^2 m \log n)$ time.

Instead of this, we use a technique similar to the exact search. The difference is that we have to go through all combinations of first two pointer positions. If we use binary search to calculate the remaining pointer positions, the running time of the algorithm is $O(n^2 m \log n)$. However, we achieve a better running time $O(n^2 m)$ by increasing each pointer stepwise as in the exact search algorithm.

Time-warped search

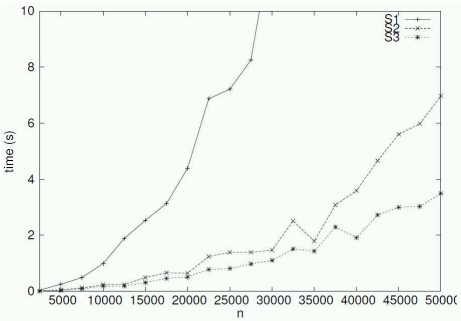
There are two previous algorithms for time-warped search. The first algorithm [3] resembles the previous time-scaled algorithm and its running time is $O(n^2 m \log n)$ as well. The second algorithm [4] uses dynamic programming and its running time is $O(n^2 m)$.

In our first approach, we first fix the first note of the occurrence and then check if all the remaining notes can be found after it. A straightforward implementation produces an algorithm with running time $O(n^2)$. If we use binary search instead, the running time is only $O(nm \log n)$.

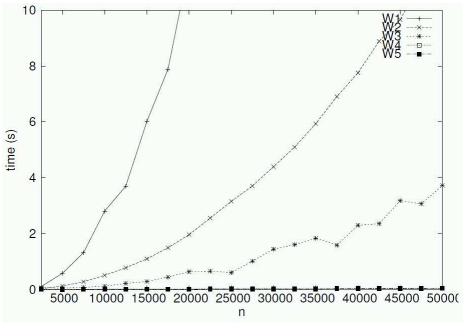
An alternative approach is to track the pattern occurrences simultaneously. First, all database notes are potential beginning notes for an occurrence. Then, we extend each occurrence one note at a time as long as it is possible. This algorithm can be implemented in $O(n(m+\log n))$ time. Moreover, if the set of possible pitches is constant, the running time is only $O(nm)$.

EXPERIMENT

In the first experiment, we compared the previous time-scaled algorithm (S1), our $O(n^2 m \log n)$ algorithm (S2) and our $O(n^2 m)$ algorithm (S3). While S1 and S2 have the same time complexity, S2 seems to be much faster in practice. S3 was still somewhat faster, especially for the largest databases.



In the second experiment, we compared the previous time-warped algorithms (W1 and W2), our $O(n^2)$ algorithm (W3), our $O(nm \log n)$ algorithm (W4) and our $O(n(m+\log n))$ algorithm (W5). W4 and W5 were clearly superior to the other algorithms, and W3 also performed well.



REFERENCES

1. Ukkonen, Lemström and Mäkinen: Geometric Algorithms for Transposition Invariant Content-Based Music Retrieval. In: 4th International Symposium on Music Information Retrieval (2003)

2. Lemström: Towards More Robust Geometric Content-Based Music Retrieval. In: 11th International Society for Music Information Retrieval Conference (2010)

3. Lemström, Laitinen: Transposition and Time-Warp Invariant Geometric Music Retrieval Algorithms. In: 3rd International Workshop on Advances in Music Information Research (2011)

4. Laitinen, Lemström: Dynamic Programming in Transposition and Time-Warp Invariant Polyphonic Content-Based Music Retrieval. In: 12th International Society for Music Information Retrieval Conference (2011)

TEXT SUMMARIZATION BASED ON WORD ASSOCIATIONS

Oskar Gross, Antoine Doucet, Hannu Toivonen

TEXT SUMMARIZATION

The goal of (multi)document summarization task is to produce a short **n** word summary given a single document or a set of documents.
We propose a methodology *association mixture* which utilizes word associations to pick sentences from the documents to generate summaries.

METHOD

The association mixture model has two components. Both components contain sentence wise word co-occurrence counts. We use the multinomial distribution to model the probabilities. The counts are used to approximate model parameters.

Background component contains word associations calculated from a large document corpus and describe common associations between words (e.g., *car-tyre, sylvester-stallone, hotel-paris*).

Independence component considers the word associations found in the document(s).

We assume that word associations reflect (on some level of abstraction) the information presented in the documents. Some of these associations are fairly common and some are specific to the documents. We are interested in latter ones.

The background component is used in order to 'cancel out' word associations which are expected (e.g., *los-angeles*). In addition, we are also interested in words which have a **high probability** of forming pairs (such as words *'the', 'who','we'* etc) in order to decrease the influence of pairs containing such words.

Association Strength

The word associations strength is obtained by contrasting the association mixture to the background component. The strength between words t_i and t_j is given in (1).

$$w(t_i, t_j) = -2 \log \frac{L(p^{\mathrm{B+D\text{-}ind}})}{L(p^{\mathrm{D}})}. \qquad (1)$$

Selecting Sentences

For finding the summary we will pick sentences which best cover the highest scoring Association Mixture associations. This is related to the *weighted set cover* problem - cover as much of the associations using the sentences where, due to the limited length of the summary, the cost is the number of words in the sentence.

EXPERIMENTS

We use DUC 2007 dataset for evaluation. The dataset:

- Consists of 45 topics;
- Each topic contains 25 documents;
- Human written summaries by DUC;
- Length of summaries: 250 words;
- ROUGE (Recall-Oriented Understudy for Gisting Evaluation) is used for measuring the accuracy against human summaries.

RESULTS

The comparison to the state-of-the-art methods is given in Table 1 and Figure 1. Figure 1 also illustrates how the different components of the association mixture behave together.

CONCLUSIONS

Methods performance is comparable to state-of-the-art methods.
We can observe, that already small corpus improves the performance of the background component.
Our method is unsupervised and is largely language independent.

EXAMPLE SUMMARY

Independence component considers the word associations found in the document(s).
The word associations strength is obtained by contrasting the association mixture to the background component.
The strength between words t_i and t_j is given in (1).
For finding the summary we will pick sentences which best cover the highest scoring Association Mixture associations.

Generated by the system by using text in this poster

METHOD	ROUGE-1	ROUGE-2	ROUGE-3	ROUGE-L
NIST BL	0.335	0.065	0.019	0.311
DSDR-LIN [2]	0.361	0.072	0.021	0.324
RANDOM	0.363	0.064	0.018	0.335
DSDR-NON [2]	0.396	0.074	0.020	0.353
NTDSDR [3]	0.398	0.082	-	0.362
CLASSY04	0.401	0.093	0.031	0.363
ASSOC MIX.⁺	0.424⁺	0.104⁺	0.036⁺	0.384⁺
ETTM* [4]	0.441*	0.104*	-	-
TTM* [4]	0.447*	0.107*	-	-

TABLE 1: * USES WORDNET AND TOPIC DESCRIPTIONS AS ADDITIONAL RESOURCES. ⁺ USES BACKGROUND CORPUS AS AN ADDITIONAL RESOURCE.

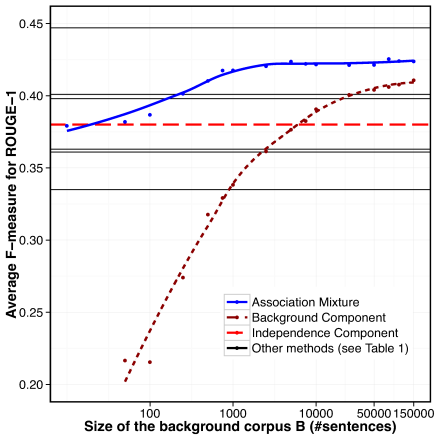


FIGURE 1: PERFORMANCE OF THE METHODS IN TERMS OF AVERAGE ROUGE-1 F-MEASURE, AS THE FUNCTION OF THE SIZE OF THE BACKGROUND CORPUS B.

PREVIOUS PUBLICATIONS

[1] GROSS, O., TOIVONEN, H., TOIVANEN, J. M., & VALITUTTI, A. (2012, NOVEMBER). LEXICAL CREATIVITY FROM WORD ASSOCIATIONS. IN KNOWLEDGE, INFORMATION AND CREATIVITY SUPPORT SYSTEMS (KICSS), 2012 SEVENTH INTERNATIONAL CONFERENCE ON (PP. 35-42). IEEE.

[2] Z. HE, C. CHEN, J. BU, C. WANG, L. ZHANG, D. CAI, AND X. HE. DOCUMENT SUMMARIZATION BASED ON DATA RECONSTRUCTION. IN TWENTY-SIXTH AAAI CONFERENCE ON ARTICIAL INTELLIGENCE, PAGES 620-626, 2012.

[3] Z. ZHANG, H. LI, ET AL. TOPICDSDR: COMBINING TOPIC DECOMPOSITION AND DATA RECONSTRUCTION FOR SUMMARIZATION. IN WEB-AGE INFORMATION MANAGEMENT, PAGES 338-350. SPRINGER, 2013.

[4] A. CELIKYILMAZ AND D. HAKKANI-TUR. DISCOVERY OF TOPICALLY COHERENT SENTENCES FOR EXTRACTIVE SUMMARIZATION. IN ACL, PAGES 491-499, 2011.

HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE