Algorithmic exploration of axiom spaces for efficient similarity search at large scale

Tomáš Skopal and Tomáš Bartoš

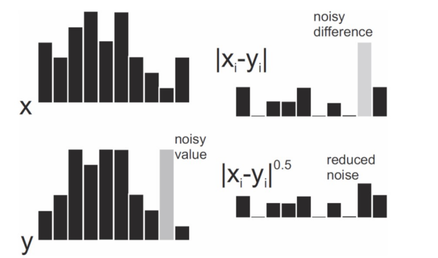
SIRET Research Group, Faculty of Mathematics and Physics,
Charles University in Prague, Czech Republic
www.siret.cz

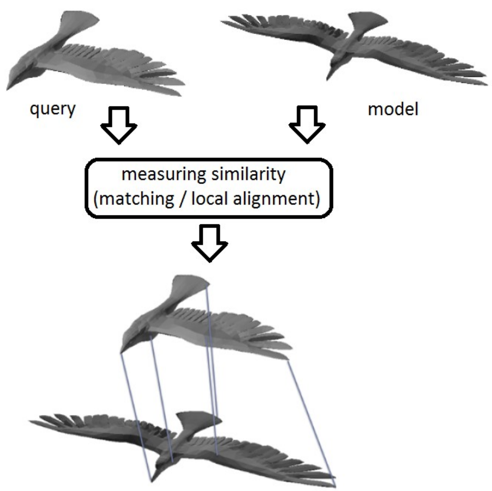
Outline

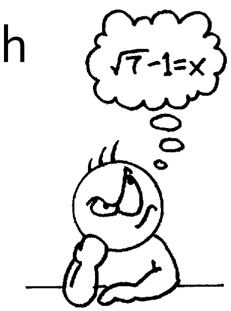
- nonmetric similarities
- indexing nonmetric similarities – related work
- motivation
  - ptolemaic indexing
- SIMDEX overview
  - main goals
  - framework stages
- preliminary experiments

Nonmetric similarities

- assuming nonmetric (unconstrained) similarity for complex measures
  - robustness (e.g., noise suppressed)
  - locality (partial matching)
  - comfort of modeling
    - domain expert not stressed by math
    - complex/algorithmic similarities undecidable







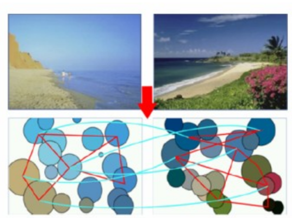
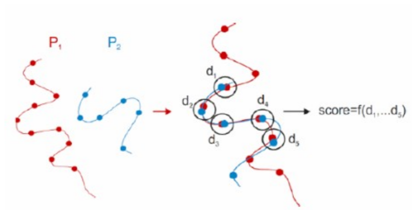


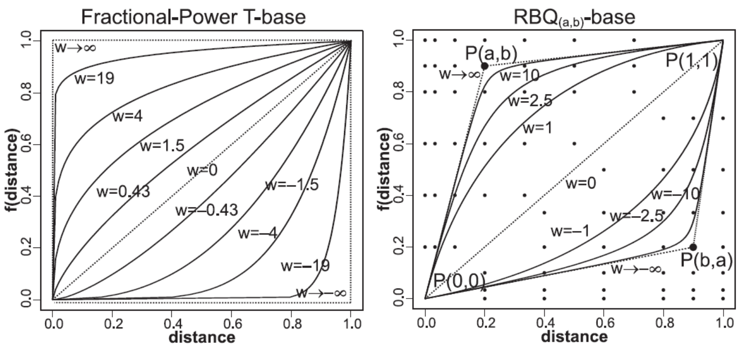
Image similarity



Protein similarity

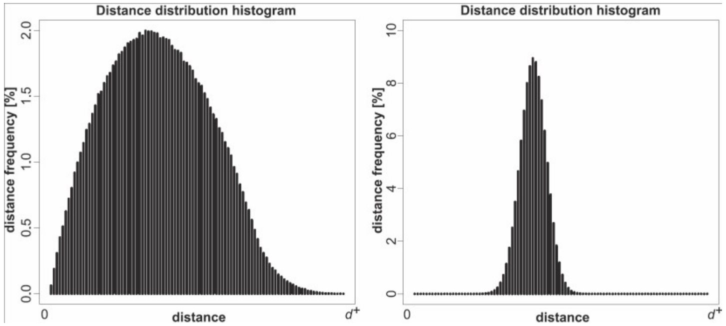
Indexing nonmetric similarities

- specific indexing (e.g., inverted index)
- general indexing
  - usually transformation into "simpler" space + indexing
  - Euclidean space + spatial access methods
    - NMDS, FastMap, MetricMap, SparseMap, BoostMap, …
    - mapping = altering the universe + distance function
  - metric space + MAMs
    - TriGen algorithm
    - mapping = universe is the same, just the distance function altered



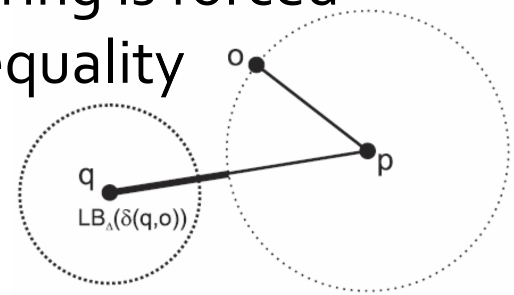
Any problem so far?

- is "metrization" of a nonmetric problem the best solution?
  - it is quite elegant solution, but the "devil lives in detail" – the target metric space is usually "overinflated" (high intrinsic dimensionality)



- why?
  - complex behavior of a similarity measuring is forced to comply with the "stupid" triangle inequality and simple filtering

$$LB_{\triangle}(\delta(q,o_i)) = |\delta(q,p) - \delta(p,o_i)|$$



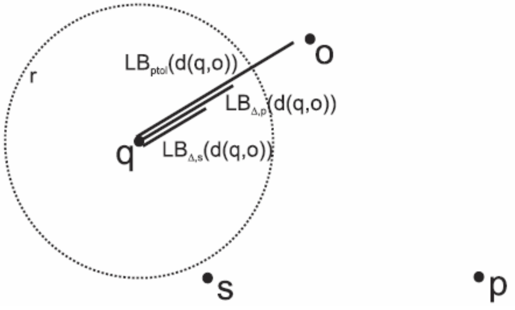
Motivation: Ptolemaic Indexing

- previous approaches
  - “rape data” to comply with an indexing formalism (metric space model)
- opposite approach
  - find an indexing formalism that comply with “data” the best
  - fuzzy similarity indexing [SISAP 2009 & 2011] – didn’t work ☹
  - **ptolemaic indexing** [SISAP 2011] – worked! ☺
    - ptolemaic inequality instead of (together with) the triangle one

$$\delta(x,v)\cdot\delta(y,u) \le \delta(x,y)\cdot\delta(u,v) + \delta(x,u)\cdot\delta(y,v)$$

    - works with for (signature) quadratic form distances (other practical distances? open problem)

$$\mathrm{LB}_{\mathrm{ptol}}(\delta(q,o)) = \frac{|\delta(q,p)\cdot\delta(o,s) - \delta(q,s)\cdot\delta(o,p)|}{\delta(p,s)}$$



SIMDEX idea

- so, we have metric indexing and ptolemaic indexing
  - we have a different way to construct the lower bounds to the original distance (or upper bound to similarity)

$$LB_{\triangle}(\delta(q,o_i)) = |\delta(q,p) - \delta(p,o_i)|$$

$$\mathrm{LB}_{\mathrm{ptol}}(\delta(q,o)) = \frac{|\delta(q,p)\cdot\delta(o,s) - \delta(q,s)\cdot\delta(o,p)|}{\delta(p,s)}$$

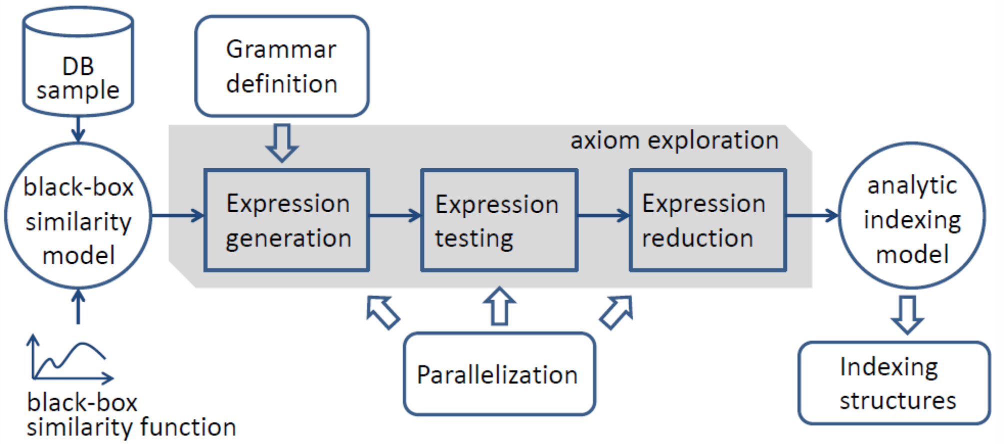
- how about to develop a framework that will discover (for a particular similarity model) an unknown **axiom**

$$\mathrm{LB}\left(\delta(q,o)\right) = \ ?$$

  such that the generated axiom will be computationally cheap and will perform better than any of the known (and named) axioms

SIMDEX framework

- no parameterized canonical forms but syntactically generated expressions
  - most general solution but very complex to handle
- stages
  - S1 – grammar definition
  - S2 – expression generation
  - S3 – expr. testing
  - S4 – expr. reduction
  - S5 – indexing
  - S6 – parallelization



SIMDEX framework

- S1 – Grammar definition
  - used to generate right-side lowerbound expressions
    - generally L3/Type-3 in Chomsky hierarchy
    - however, restriction specifics turn it into context-dependent language! (next slide)
  - terminals (combined)
    - descriptor variables ($q, o, p_1, \ldots, p_i$) and descriptor constants c_i used in the distance $\delta(\cdot, \cdot)$
    - functions f_i
    - standard arithmetic operators +,-,*,/, numeric constants
  - using the grammar a universe of expressions can be generated

SIMDEX framework

- S2 – Expression generation
  - exponential even when the grammar and recursion are limited
  - exploration of the expression universe
    - FIFO, LIFO, random, heuristic traversal
    - interleaved
  - restrictions complicating the language (context-dependent)
    - require $\delta(q,p_i)$, $\delta(p_i,o)$
    - avoid $\delta(q,o)$
    - avoid duplicates (lexical but also semantics, e.g., p_i, p_j the same)
    - avoid useless arithmetic operations (e.g., $\delta(p_i,o) - \delta(p_i,o)$)

SIMDEX framework

- S3 – Expression testing
  - testing each generated expression as an axiom candidate
  - application on the input distance/similarity matrix
  - either full axiom (all tests pass), or a partial
- S4 – Expression reduction
  - discarding weaker expressions
    (producing larger lowerbounds)
  - merging a set of expressions into a compound tighter form

SIMDEX framework

- S5 – Indexing
  - verifying the real usefulness of the passed expressions
  - Pivot table-like index can be always used (direct LB filter)
  - some expressions might be interpreted as “nestable” regions in the similarity space and so applicable to hierarchical indexing
    - such as the ball-regions for triangle inequality are
- S6 – Parallelization
  - the axiom space is huge even after all the optimization stages, so massive parallelization is critical
    - multicore CPU, manycore GPU, Map-Reduce on CPU farm

SIMDEX initial implementation

- covering stages S1-S3
- expressions generated by heuristics (fingerprints optimization)

Algorithm 1 SIMDEX (G, C, T, S, δ)

Require: Grammar definition G, validation conditions C, threshold probability value T, database sample S, distance function δ

$M_{\delta,S} \leftarrow$ new distance matrix (δ, S)
$expressions \leftarrow$ ExpressionGeneration(G, C)
for all E_i in $expressions$ **do**
  if $validate_C(E_i)$ equals **false then**
    $expressions$.Remove(E_i) {validity check fails}
    continue {skip further testing of the expression E_i}
  end if
  if $E\_test(E_i, M_{\delta,S}) < T$ **then**
    $expressions$.Remove(E_i) {probability test fails}
  end if
end for
return $expressions$ {remaining expressions compose the result set}

Preliminary experiments

Dataset	Expression	Success Ratio	MIN	MAX	AVG
Corel	triangle inequality	99 %	0.0034	0.9983	0.3764
	$\lvert\delta(q,p)\cdot\delta(o,p)\cdot(\delta(o,p)-\delta(q,p))\rvert$	100 %	0.1059	0.9991	0.5020
	$(\delta(q,p)-\delta(o,p))^2$	100 %	0.1352	0.9999	0.5054
	$\lvert(\delta(q,p_1)-\delta(o,p_1))(\delta(q,p_1)-\delta(o,p_2))\rvert$	100 %	0.0420	0.9999	0.5161
CoPhIR	triangle inequality	97.5 %	0.0021	0.9736	0.2696
	$(\delta(q,p)-\delta(o,p))^2$	100 %	0.0718	0.9979	0.3808
	$\lvert(\delta(q,p_1)-\delta(o,p_1))(\delta(q,p_2)-\delta(o,p_2))\rvert$	100 %	0.0845	0.9969	0.3935
Ratings	triangle inequality	100 %	0.6067	1.0	0.9037
	$\frac{1}{2\cdot\delta(o,p)}$	100 %	0.0119	0.5	0.4254
	$(\delta(q,p_1)+\delta(o,p_1))\cdot\frac{\delta(q,p_1)}{\delta(p_1,p_2)}$	100 %	0.0103	0.5845	0.4254
Listeria	triangle inequality	99 %	0	0.9559	0.1388
	$\delta(p_1,p_2)\cdot\frac{1}{\delta(p_1,p_2)+\delta(o,p_2)}$	100 %	0.0075	0.9994	0.2393
	$\delta(q,p_1)^2\cdot\frac{1}{\delta(o,p_2)\cdot\delta(q,p_2)}$	100 %	0.0008	0.9985	0.2401
	$(\delta(q,p_1)+\delta(o,p_1))\frac{\delta(q,p_1)}{\delta(p_1,p_2)}$	100 %	0.0032	0.9970	0.2555
Spectometry	triangle inequality	100 %	0.1823	0.93	0.7329
	$\delta(o,p)-\delta(o,p)^2$	100 %	0.0009	0.8758	0.6638
	$\lvert(\delta(q,p_1)\cdot\delta(o,p_2))-\delta(q,p_2)^2\rvert$	100 %	0.0148	0.9399	0.7054

Conclusions and future work

- SIMDEX sketched
  - universal algorithmical framework for discovering axioms suitable for indexing specific similarity models
  - breaking the metric space paradigm

- a lot of future work ahead!
  - all the stages need to be optimized

Challenges

- two challenges for the SISAP community
  - join us for developing the SIMDEX stages!
    (the axiom space is really huge to search by the current unoptimized implementation)
  - answer/prove the holy grail "SIMDEX spoiler" problem:

    Is the metric space model the "killer model" for general indexing, so that anything else (found by SIMDEX) is worse?

    (including a transformation step, like TriGen)

Thank you…

… for your attention!

questions?