On fuzzy vs. metric similarity search in complex databases

Alan Eckhardt[1,2], Tomáš Skopal[1], and Peter Vojtáš[1,2]

[1] Department of Software Engineering, Charles University,
[2] Institute of Computer Science, Czech Academy of Science,
Prague, Czech Republic
{eckhardt,skopal,vojtas}@ksi.mff.cuni.cz,

Abstract. The task of similarity search is widely used in various areas of computing, including multimedia databases, data mining, bioinformatics, social networks, etc. For a long time, the database-oriented applications of similarity search employed the definition of similarity restricted to metric distances. Due to the metric postulates (reflexivity, non-negativity, symmetry and triangle inequality), a metric similarity allows to build a metric index above the database which can be subsequently used for *efficient* (fast) similarity search. On the other hand, the metric postulates limit the domain experts (providers of the similarity measure) in similarity modeling. In this paper we propose an alternative non-metric method of indexing for efficient similarity search. The requirement on metric is replaced by the requirement on fuzzy similarity satisfying the transitivity property with a tuneable fuzzy conjunctor. We also show a duality between the fuzzy approach and the metric one.

1 Introduction

The task of similarity search is widely used in various areas of computing, including multimedia databases, data mining, bioinformatics, social networks, etc. In fact, retrieval of any semantically unstructured data entities requires a form of an aggregated qualification that returns data relevant to a query. A popular type of such a mechanism is similarity querying.

Unlike exact search (e.g., SQL SELECT used in relational databases), the only practicable way how to process and retrieve the vast volumes of unstructured data is the *content-based similarity search*, i.e., we consider the real content of each particular database object rather than its external annotation or metadata. For example, similarity search in a database of images considers the raw image data (colors, shapes, textures, etc.), not keywords or other external annotation. Unlike traditional strong-typed rows in relational database tables or XML with a schema, the unstructured objects (like images) have no universal and/or unique syntactic and semantic structure. Hence, the most general and feasible abstraction used in content-based retrieval is the *query-by-example* concept, where the database objects are ranked according to similarity to a query object (the example). Only such database objects are retrieved, which have been

ranked as sufficiently similar to the query object. The *similarity measure* returns a real-valued similarity score for any two objects on the input.

Right at the beginning we must emphasize that this paper does not deal with similarity modeling, i.e., with the domain-specific *effectiveness* of search. That is, we do not propose new or improved similarity search paradigm or better similarity measures which should improve the quality of query result. Instead, we focus on the *efficiency* of the search, which is just a database-specific problem. In simple words, the core problem could be formulated as follows. A domain expert (e.g., expert in computer vision/graphics, or even an expert outside computer science, like radiologist, geologist, biologist etc.) has a database of data entities (e.g., images, time series, audio tracks, 3D models, etc.) and a domain-specific similarity measure defined for that data (e.g., Smith-Waterman similarity for matching protein sequences). When the database is sufficiently small and/or the similarity measure is computationally cheap, the expert can use a naive way of similarity search – a query example object is sequentially compared with all the objects in the database, selecting the most similar object to the query. However, a problem arises when the database is large or the similarity measure is expensive (e.g., having super-linear time complexity with respect to the object size). At that moment the expert needs a database-specific help that will preserve his "perfect" domain-specific model and, at the same time, will provide more efficient (faster) search. Hence, here we enter the context of our paper – efficient similarity search. Basically, efficient similarity search means keeping the number of similarity computations needed to answer a query as low as possible. In the following section we briefly discuss the metric approach to efficient similarity search and show its domain-specific restrictions.

In this paper, we focus on spaces that are not inherently metric, specifically those where triangle inequality does not hold. Two methods, one based on space transformation and the other on fuzzy logic operators, for making use of efficient search even on these spaces, are described and compared. In Section 2 we describe the basics about similarities and metric distances. An efficient query answering based on pivots using metricity of data is described there. In Section 3 we describe the TriGen algorithm that transforms non-metric similarity function so that it becomes metric, and this allows to search non-metric spaces as they would be metric ones. Finally, Section 4 and 5 describe the actual contribution of this paper – how the fuzzy logic can be used to see a non-metric space as a metric one and how this view can be applied for efficient query answering. We also study the duality between metric and fuzzy similarity approaches which can lead to better understanding of the proposed fuzzy approach.

2 Metric approach

In the database domain, the models of similarity retrieval are based on simplifying similarity space abstraction. Let a complex unstructured object $\mathcal{O}$ be modelled by a *model object* $o \in \mathbb{U}$, where $\mathbb{U}$ is a model universe, which could be a Cartesian product over attribute sets, a set of various structures (polygons,

graphs, other sets, etc.), string closure, sequence closure, etc. A database $\mathcal{S}$ is then represented by a dataset $\mathbb{S} \subset \mathbb{U}$.

Definition 1 (similarity & dissimilarity measure)
Let $s : \mathbb{U} \times \mathbb{U} \mapsto \mathbb{R}$ be a *similarity measure*, where $s(o_i, o_j)$ is considered as a similarity score of objects $\mathcal{O}_i$ and $\mathcal{O}_j$. In many cases it is more suitable to use a *dissimilarity measure* $\delta : \mathbb{U} \times \mathbb{U} \mapsto \mathbb{R}$ equivalent to a similarity measure $s(\cdot, \cdot)$ as $s(q, o_i) > s(q, o_j) \Leftrightarrow \delta(q, o_i) < \delta(q, o_j)$. A dissimilarity measure (also *distance*) assigns a higher score to less similar objects, and vice versa. The pair $\mathcal{D} = (\mathbb{U}, \delta)$ is called a *dissimilarity space* – a kind of topological space. □

2.1 Metric distances

The distance measures often satisfy some of the metric properties ($\forall o_i, o_j, o_k \in \mathbb{U}$):

$$\delta(o_i, o_j) = 0 \Leftrightarrow o_i = o_j \quad \text{reflexivity} \tag{1}$$
$$\delta(o_i, o_j) > 0 \Leftrightarrow o_i \neq o_j \quad \text{non} - \text{negativity} \tag{2}$$
$$\delta(o_i, o_j) = \delta(o_j, o_i) \quad \text{symmetry} \tag{3}$$
$$\delta(o_i, o_j) + \delta(o_j, o_k) \geq \delta(o_i, o_k) \quad \text{triangleinequality} \tag{4}$$

The *reflexivity* (1) permits the zero distance just for identical objects. Both reflexivity and *non-negativity* (2) guarantee every two distinct objects are positively dissimilar. If δ satisfies reflexivity, non-negativity and *symmetry* (3), we call δ a *semimetric*. Finally, if a semimetric δ satisfies also the *triangle inequality* (4), we call δ a *metric* (or metric distance). The triangle inequality is a kind of transitivity property; it says if o_i, o_j and o_j, o_k are similar, then also o_i, o_k are similar. If there is an upper bound d^+ such that $\delta : \mathbb{U} \times \mathbb{U} \mapsto \langle 0, d^+ \rangle$, we call δ a *bounded metric*. In such case $\mathcal{M} = (\mathbb{U}, \delta)$ is called a (bounded) *metric space*.

To complete the enumeration, we also distinguish *pseudometrics* (not satisfying the reflexivity), *quasimetrics* (not satisfying symmetry) and *ultrametrics* (a stronger type of metric, where the triangle inequality is restricted to ultrametric inequality – $max\{\delta(o_i, o_j), \delta(o_j, o_k)\} \geq \delta(o_i, o_k)$).

2.2 Traditional approach to efficient metric search

We base our comparison with traditional metric search [1, 2]. It uses metric distance (dissimilarity) for efficiently searching the answer to a query. A range query is represented by an object $q \in \mathbb{U}$ and the maximal distance $\epsilon > 0$, while it selects database objects within the distance ϵ from q (e.g. $\{o : \delta(q, o) \leq \epsilon\}$, $o \in \mathbb{S}$).

When the dataset is large (and/or intrinsically high-dimensional), there is a preprocessing needed, so that subsequent frequent querying would not imply an exhaustive sequential search (evaluation of $|\mathbb{S}|$ distance computations for each query). As a simple yet representative metric access methods, the LAESA [3] uses a set of pivots $P \subseteq \mathbb{S}$ by use of which an index is created. The distances

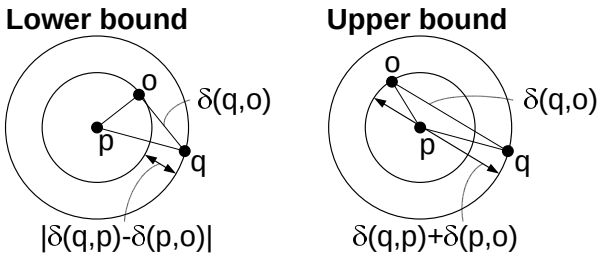

Fig. 1. Using pivots with distances for metric search.

$\delta(p, o)$ from each pivot $p \in P$ to each object o in the dataset $\mathbb{S}$ are computed in advance, forming a distance matrix of size $|P| \times |\mathbb{S}|$ – the LAESA index.

Then, in the time of a query (q, ϵ), the lower and the upper bounds on the distances between the query and all database objects are computed using these pivots and the index matrix without the need of any additional distance computation $\delta(\mathrm{q}, \cdot)$. An example of estimation of both bounds is on Figure 1. Three points o, p and q represent an object from database, a pivot, and the query object. Then the distance $\delta(q, o)$ between q and o can be bounded by the following formula:

$$|\delta(q, p) - \delta(p, o)| \leq \delta(q, o) \leq \delta(q, p) + \delta(p, o) \tag{5}$$

In the following text, we will refer to *lower bound* and *upper bound*. In case of metric distance the lower bound is $|\delta(q, p) - \delta(p, o)|$ (i.e. the lowest possible distance between two objects) and the upper bound is $\delta(q, p) + \delta(p, o)$ (i.e. the highest possible distance). To be as effective as possible, the upper/lower bounds should be as tight as possible. That is the reason we use multiple pivots – the lower bound distance is then defined as $max_{p_i \in P}\{|\delta(q, p_i) - \delta(p_i, o)|\}$. Similarly, the upper bound is defined as $min_{p_i \in P}\{\delta(q, p_i) + \delta(p_i, o)\}$.

A range query can be efficiently processed using lower and upper bounds in a filter-and-refine manner. Prior to the query processing, the distances $\delta(q, p_i)$ are computed. In the filter step, each database object that has its lower bound distance from q larger than ϵ is safely filtered out from further processing. Conversely, a database object that has its upper bound smaller than ϵ is safely confirmed as a part of the result. Note that in the refine step we need not to compute any additional distance computation, as all the distances are either stored in the matrix ($\delta(p_i, o)$ distances) or computed prior to the query processing ($\delta(q, p_i)$ distances). All the non-filtered/non-confirmed database objects are sequentially processed in the refine step (implying thus a distance computation for each object in the rest). When the number of pivots is large enough and/or their distribution in space is "good", the filter step could prune a substantial part of the entire dataset, so the query processing becomes efficient.

2.3 Limitations of the metric approach

As the quantity/complexity of multimedia data grows, there is a need for more complex similarity measuring. Here the metric model exhibits its drawbacks, since the domain experts (being not computer scientists) are forced to "implant" metric properties into their non-metric measures, which is often impossible.

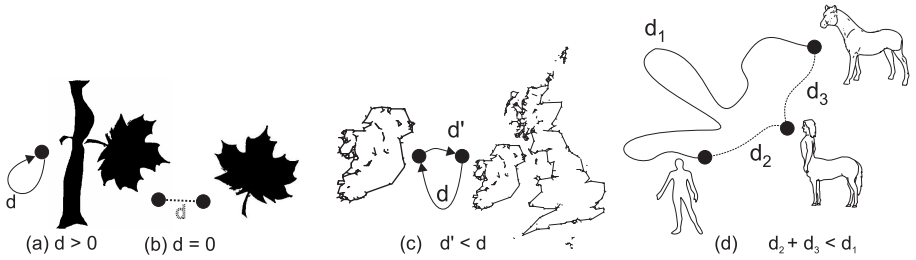

Fig. 2. Objections against metric properties in similarity measuring: (a) reflexivity (b) non-negativity (c) symmetry (d) triangle inequality

However, a non-metric similarity has also a qualitative justification. In particular, the reflexivity and non-negativity have been refuted by claiming that different objects could be differently self-similar [4, 5]. For example, in Figure 2a the leaf on a trunk can be viewed as positively self-dissimilar if we consider the less similar parts of the objects (here the trunk and the leaf). Or, alternatively, in Figure 2b the leaf-on-trunk and leaf could be treated as identical if we consider the most similar parts of the objects (the leaves). The symmetry was questioned by showing that a prototypical object can be less similar to an indistinct one than vice versa [6, 7]. In Figure 2c, the more prototypical "Great Britain and Ireland" is more distant to the "Ireland alone" than vice versa. The triangle inequality is the most attacked property. Some theories point out the similarity has not to be transitive [8, 9]. Demonstrated by the well-known example, a man is similar to a centaur, the centaur is similar to a horse, but the man is completely dissimilar to the horse (see Figure 2d).

3 Generalized metric search

To overcome the problem with restrictiveness of the metric space model (as discussed in Section 2.3), the *TriGen algorithm* [10, 11] allows to index databases under non-metric similarity measures. Instead of full metric, the domain expert is not forced to implant the triangle inequality into his dissimilarity measure – it is accomplished automatically by the TriGen algorithm using a suitable transformation of the original dissimilarity. Simply, the TriGen turns a non-metric into a metric. We also emphasize this transformation does not alter the

effectiveness of the retrieval model, that is, the resulting metric is equivalent to the original non-metric (with respect to the task of similarity search).

In particular, TriGen can non-trivially put more or less of the triangle inequality into any semimetric δ (i.e., into any reflexive, non-negative, symmetric distance), hence, keeping the database indexable by triangle inequality. Thus, any semimetric distance can be turned into an equivalent full metric (allowing exact search by metric access methods), or into a semimetric which satisfies the triangle inequality to some user-defined extent (allowing approximate search). For its functionality, the TriGen needs a (small) sample of the database objects.

The principle behind TriGen is a usage of triangle triplets and T-bases. A triplet of numbers (a, b, c) is *triangle triplet* if $a + b \geq c, b + c \geq a, a + c \geq b$. If, for a distance δ, all triplets $(\delta(o_i, o_j), \delta(o_j, o_k), \delta(o_i, o_k))$ on all possible objects o_i, o_j, o_k are triangle triplets, then δ satisfies the triangle inequality. Using triangle triplets, we measure the *T-error* – a degree of triangle inequality violation, computed as the proportion of non-triangle triplets in all examined distance triplets.

A *T-base* $f(x, w)$ is an increasing function (where $f(0, w) = 0$) which turns a value $x \geq 0$ of an input (semi)metric δ into a value of a target (semi) metric δ^f, i.e., $\delta^f(\cdot, \cdot) = f(\delta(\cdot, \cdot), w)$. Besides the input distance value x, the T-base is parameterized also by a fixed weight $w \in \langle -\infty, \infty \rangle$ which determines how concave or convex f should be. The higher $w > 0$, the more concave f, which means also the lower T-error of any δ^f. Conversely, the lower $w < 0$, the more convex f and the higher T-error of any δ^f.

4 Fuzzy similarity approach

Although the TriGen algorithm freed the domain experts from manual implementation of the triangle inequality into their similarity measures, it still relied on the metric space model. Hence, a dissimilarity was transformed into a metric, and from that moment on the metric was used by the metric access method. However, we must realize the metricity of a similarity measure is not a qualitative goal, it is just database-specific requirement that enables efficient similarity search. To satisfy triangle inequality, the TriGen algorithm actually "inflates" the original non-metric space to become a metric one. However, in highly non-metric spaces this inflation could be very high, leading to almost useless distance distributions where each object in the database is very far from the rest of objects. Such a highly inflated space cannot be efficiently indexed because of its high intrinsic dimensionality (a variant of the curse of dimensionality, for details see [10]).

Fortunately, the metric postulates are surely not the only topological properties a similarity measure may satisfy. We can imagine infinitely many such properties, however, only some of them could be empirically observed from the data (from distance distribution, actually), and, furthermore, only some of them are useful for an implementation of efficient similarity search. In this section we propose such an alternative to the metric model – the fuzzy similarity &

fuzzy logic model. Our motivation for using fuzzy logic as a method for efficient similarity search is that fuzzy logic has a great variety of operators available. So, instead of transforming the similarity itself, fuzzy logic allows us to accommodate the operators + and − that work with similarities. Hence the inherent properties of data space $\mathbb{U}$ remain untouched, only the handling with these values is modified. Note that this idea is complementary to the generalized metric model used by TriGen. Instead of modifying the similarity measure in order to fulfil the metric postulates, the fuzzy approach preserves the similarity measure but modifies the formulas employed for efficient similarity search.

We present an idea of using fuzzy similarity that fulfils the transitivity property with a tuneable fuzzy conjunctor. If triangle inequality of dissimilarity does not hold, it is still possible that there is a fuzzy conjunctor such that transitivity of similarity holds. We show that the usual indexing techniques based on pivots and triangle inequality for range queries can be applied also in the fuzzy similarity approach.

We stress that this paper has nothing to deal with fuzzy databases. Our databases are not fuzzy. We have crisp data and queries are range and k-NN queries. We have a domain expert similarity measure. Fuzzy theory is used here only for obtaining alternative indexing. Connections to fuzzy databases are out of scope of this paper; main emphasis is on indexing structures for efficient retrieval.

4.1 Fuzzy similarity

Triangle inequality was substantial in the above mentioned metric model of similarity search. Nevertheless, when observing real-world data, a non-metric similarity measure can have some interesting properties [12]. In this section we deal with similarities rather than distances (see Definition 1). We combine these similarities using fuzzy logic connectives. The similarity $s(o_1, o_2)$ is the degree of fuzzy predicate "o_1 and o_2 are the same". As s is inverse to δ, all inequalities have to be reversed ($\geq$ replaced by $\leq$ and vice versa).

Metric triangle inequality is in the case of similarities replaced by transitivity, where a generalized fuzzy conjunction T is used (implication I will be used, too). Interested reader may read e.g. [13] for much more information about fuzzy conjunctions. In the following text, we assume at least basic knowledge of fuzzy conjunctions and implications.

Three basic fuzzy conjunctions and implications(Lukasiewicz, product and Gödel) are shown in the following table.

	Conjunction C_λ^F	Implication I_λ^F
Lukasiewicz	$max(0, x + y - 1)$	$min(1, 1 - x + y)$
product	$x * y$	$min\{y/x, 1\}$, 1 if $x = 0$
Gödel	$min\{x, y\}$	y if $x > y$, 1 if $x \leq y$

Note that all conjunctions above have different behaviour. For more flexible way of this behaviour change, families of t-norms were defined using parameter λ.

The advantage of using fuzzy logic is that it has been studied for a long time. The result is that there is quite a large number of possible parametric families of conjunctions, while every family is well studied (see [13]). Among others, we note Frank, Schweizer-Sklar, Hamacher, Frank, Yager and Dombi, where λ ranges from $-\infty$ to $+\infty$ in most cases, but some families are defined on $[0,1]$ or $[-1,+\infty)$. The differences between families are in their characteristics and also in the speed of computation.

In the rest of the paper, we will work with parametric family of Frank t-norms defined bellow, but others may be also used.

$$T_\lambda^F(x,y) = log_\lambda\left(1 + \frac{(\lambda^x - 1)(\lambda^y - 1)}{\lambda - 1}\right) \quad \lambda \neq 0, 1, \infty \tag{6}$$

$$T_0^F(x,y) = min\{x,y\} \tag{7}$$

$$T_1^F(x,y) = x * y \tag{8}$$

$$T_\infty^F(x,y) = max(0, x + y - 1) \tag{9}$$

Using conjunction T, we say that a two variable function s with $1 \geq s(o_1,o_2) \geq 0$ is T-transitive, if it fulfils the following property:

$$s(o_1,o_3) \geq T(s(o_1,o_2), s(o_2,o_3)) \quad \mathrm{T-transitivity} \tag{10}$$

Let us note that the operator $+$ is fixed in metric triangle inequality; there is a flexible conjunction T in transitivity. The advantage is that there can be many conjunctions T for which the data satisfy the condition 10.

4.2 Fuzzy similarity applied in indexing and search

A data space is said to be a T-similarity space, if for all objects $o_1, o_2, o_3 \in \mathbb{U}$ we have

$$s(o_1,o_3) \geq T(s(o_1,o_2), s(o_2,o_3))$$

To derive inequalities analogous to (5) we need residuation from fuzzy logic (here I_T is a fuzzy implication which is residual to the conjunction T, see e.g. [13] page 50).

$$T(x,y) \leq z \rightarrow I_T(x,z) \geq y \tag{11}$$

As in the metric case, let us consider a query object q, a pivot p, and an object o from the database. Let us assume T-transitivity of s. We know that $T(s(q,p), s(q,o)) \leq s(p,o)$ from T-transitivity. From residuation (11), we get

$$I_T(s(q,p), s(p,o)) \geq s(q,o) \tag{12}$$

From the fact that $T(s(p,o), s(q,o)) \leq s(q,p)$, we get

$$I_T(s(p,o), s(q,p)) \geq s(q,o) \tag{13}$$

Estimation of similarity of query and object o is

$$T(s(q,p),s(p,o)) \leq s(q,o) \tag{14}$$

$$s(q,o) \leq min\{I_T(s(q,p),s(p,o)), I_T(s(p,o),s(q,p))\} \tag{15}$$

Then a range query (e.g. $\{o : s(q,o) > 1-\epsilon\}$) can be efficiently processed by using these inequalities and $s(q,p)$, in a similar way as lower/upper bounds are used for metric-based similarity search.

Similarly to T-bases in TriGen, also here we solve a problem of finding the right parameter λ that fits the underlying similarity scores among objects in database.

Tuning problem. Given data $\mathbb{S}$ and a similarity s. Find minimal λ, $\lambda(\mathbb{S},s)$ such that s is a T_λ^F-transitive on $\mathbb{S}$.

Example. Let us suppose that we have $\mathbb{S} = \{o_1, o_2, o_3\}$ forming a non-metric triplet $\delta(o_1,o_2) = 0.2$, $\delta(o_2,o_3) = 0.8$ and $\delta(o_1,o_3) = 0.5$. We transform distance to similarity using formula $s = 1/(1+\delta)$ and get $s(o_1,o_2) \approx 0.83$, $s(o_2,o_3) \approx 0.55$ and $s(o_1,o_3) \approx 0.66$.

Let us try to find $s(o_1,o_3)$ using the other two similarities. Using (14) and (15) we get different estimates on $s(o_1,o_3)$ for different λ.

Lower bound	Upper bound	λ
0.38	0.72	$\lambda = \infty$
$T_\lambda^F(0.83, 0.55)$	$I_\lambda^F(0.83, 0.55)$	$\lambda \in (1, \infty)$
0.46	0.66	$\lambda = 1$
$T_\lambda^F(0.83, 0.55)$	$I_\lambda^F(0.83, 0.55)$	$\lambda \in (0, 1)$
0.55	0.55	$\lambda = 0$

From the summary above, we can see that T-transitivity of s holds for $\lambda \leq 1$.

5 Duality between metric and fuzzy similarity approach

There is an interesting phenomenon, when looking to definitions of metric and fuzzy similarity. We see relating patterns in symmetry, reflexivity, non-negativity, but most striking is the duality between triangle inequality and transitivity.

For a metric distance measure δ, the triangle inequality reads as follows

$$\delta(o_1,o_3) \leq \delta(o_1,o_2) + \delta(o_2,o_3) \tag{16}$$

For a similarity s and a t-norm T, the transitivity reads as follows

$$s(o_1,o_3) \geq T(s(o_1,o_2), s(o_2,o_3)) \tag{17}$$

We omit now the problem that s is bounded and δ has not to be.

What we see is that + in (16) corresponds to T in (17). From the fuzzy point of view + is a "sort of disjunction" (not considering unboundedness).

Inequality $\leq$ in (16) corresponds to $\geq$ in (17). Note that (17) is equivalent to a fuzzy Datalog rule

$$s(o_1, o_2) \ \& \ s(o_2, o_3) \longrightarrow s(o_1, o_3) \tag{18}$$

where s is a binary predicate and & is a fuzzy conjunction with truth value function being T (more on this see [14]).

Inverting inequality and switching from a disjunction to a conjunction points to an order inverting duality between s and δ. From a pragmatic point of view, this duality can be, e.g., $s = 1 - \delta$, $\delta = 1 - s$, or $s = \frac{1}{1+d}$, $\delta = \frac{1}{1+s}$ or any order inverting function with some reasonable properties.

From a fuzzy logic point of view such a duality points to the duality between a t-norm and a t-conorm, or to deMorgan laws with an order inverting negation (usually $\neg(x) = 1 - x$).

From the point of view of range querying and an indexing scheme supporting efficient range querying, we see a duality between inequalities used to restrict range query when using pivot indexing (as described in Section 2.2).

In a metric space, restricting search if indexed via pivots is done using following inequalities

$$|\delta(q, p) - \delta(p, o)| \leq \delta(q, o)$$

and

$$\delta(q, o) \leq \delta(q, p) + \delta(p, o)$$

Again in fuzzy similarity model the corresponding inequalities

$$min\{I_T(s(q, p), s(p, o)), I_T(s(p, o), s(q, p))\} \geq s(q, o) \tag{19}$$

and

$$s(q, o) \geq T(s(q, p), s(p, o)) \tag{20}$$

can be also used for the restriction of range queries using pivot indexing.

Again, the duality between (5) and (20) resembles duality between triangle inequality and transitivity, inverting order and replacing (a quasi disjunction) + with a t-norm T.

Much more interesting is the duality between (5) and (19). Again duality is order inverting. The difference $-$ corresponds to residuation (from fuzzy logic point of view residuation to Lukasiewicz is expressed by $-$). Moreover absolute value corresponds to min - choosing from two possibilities of computing residuation. Moreover one of residual values I_T is not interesting when being equal to 1 and one of $\delta(q, p) - \delta(p, o)$ and $\delta(p, o) - \delta(q, p)$, namely the negative one is not interesting. The duality is much more clear when rewriting the absolute value to max:

$$max\{\delta(q, p) - \delta(p, o), \delta(p, o) - \delta(q, p)\} \leq \delta(q, o) \tag{21}$$

Now everything is nicely dual: min to max, $\leq$ to $\geq$ and $-$ to I_T. So it seems that the duality is really deep.

5.1 Consequences of duality

Now the crucial problem for us is: "Is it just a duality or does it bring (or at least, is there a chance that it brings) an improvement?". A detailed discussion on this question is out of the scope of this paper. Nevertheless, we make some initial observations.

Mapping metric to fuzzy similarity. We have already observed that (omitting boundedness problem) any order inverting function mapping s to δ and vice versa is a candidate for the transformations between distance and similarity.

We have observed that using $s = 1 - \delta$ (and $\delta = 1 - s$, resp.) does not make any difference. The main reason is that it makes metric data a transitive set only with respect to Lukasiewicz logic, because Lukasiewicz corresponds directly to metric $+$ and $-$.

Using $s = \frac{1}{1+\delta}$ (and $\delta = \frac{1}{1+s}$) is more interesting. There are other possible transformations between distance and similarity. The deeper inspection of them is out of the scope of the paper.

We have tested two facts.

1. How is the similarity distribution histogram changing.
2. How does the λ parameter behave, namely making the data set a T_λ transitive set.

Here it is also interesting to consider the percentage (amount of) data fulfilling the transitivity. From the fuzzy logic point of view, we have some data which violate the transitivity constraint given by the conjunction. Consider an extreme t-norm: $T_D(x, y) = x$ for $y = 1$, y for $x = 1$, and 0 otherwise. When a triple of objects comes with similarities $s(o_1, o_2) = 1$, $s(o_2, o_3) = 0.5$, $s(o_1, o_3) = 0.2$, there is no fuzzy t-norm where the following equation holds: $s(o_1, o_3) = T(s(o_1, o_2), s(o_2, o_3))$. These exceptions have to be handled separately – fortunately, they can be detected beforehand.

6 Conclusion

We have presented a fuzzy similarity search model which enables efficient range queries processing by use of pivots in non-metric spaces. This model is based on transformation of operators rather than similarity, which may result in conservation of properties of similarity.

The duality between distance and similarity has also been studied in depth, providing a new insight into the similarity querying as a whole. This duality is the key aspect when we try to use fuzzy logic in query processing.

6.1 Future work

It remains for future work to design algorithms for the tuning problem. It would be also interesting to compare querying load for our approach and the one from

[10, 15] on different non-metric data. There is also a possibility to combine both methods – to use TriGen as a preprocessing that transforms the similarity, and then to deal with such a transformed data using fuzzy operators. Because fuzzy conjunctions can be more flexible than +, the requirements on how drastically the similarity is transformed by TriGen should be lower, thus making TriGen faster and less destructive.

Acknowledgements. The work on this paper was supported by Czech projects MSM 0021620838, 1ET 100300517, GACR 201/09/H057 and GACR 201/09/0683.

References

1. Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity Search: The Metric Space Approach (Advances in Database Systems). Springer (November 2005)
2. Samet, H.: Foundations of Multidimensional and Metric Data Structures (The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2005)
3. Mico, M.L., Oncina, J., Vidal, E.: A new version of the nearest-neighbour approximating and eliminating search algorithm (aesa) with linear preprocessing time and memory requirements. Pattern Recogn. Lett. **15**(1) (1994) 9–17
4. Krumhansl, C.L.: Concerning the applicability of geometric models to similar data: The interrelationship between similarity and spatial density. Psychological Review **85**(5) (1978) 445–463
5. Tversky, A.: Features of similarity. Psychological review **84**(4) (1977) 327–352
6. Rosch, E.: Cognitive reference points. Cognitive Psychology **7** (1975) 532–47
7. Rothkopf, E.: A measure of stimulus similarity and errors in some paired-associate learning tasks. J. of Experimental Psychology **53**(2) (1957) 94–101
8. Ashby, F., Perrin, N.: Toward a unified theory of similarity and recognition. Psychological Review **95**(1) (1988) 124–150
9. Tversky, A., Gati, I.: Similarity, separability, and the triangle inequality. Psychological Review **89**(2) (1982) 123–154
10. Skopal, T.: Unified framework for fast exact and approximate search in dissimilarity spaces. ACM Trans. Database Syst. **32**(4) (2007) 29
11. Skopal, T.: On fast non-metric similarity search by metric access methods. In: Proc. 10th International Conference on Extending Database Technology (EDBT'06). LNCS 3896, Springer (2006) 718–736
12. Pekalska, E., Harol, A., Duin, R., Spillman, D., Bunke, H.: Non-euclidean or non-metric measures can be informative. In: Structural, Syntactic, and Statistical Pattern Recognition, Proc. SSSPR2006 (Hong Kong, China, August 2006). Volume 4109 of Lecture Notes in Computer Science. Springer Verlag, Berlin (2006) 871–880
13. Klement, E.P., Mesiar, R., Pap, E.: Triangular Norms, volume 8 of Trends in Logic. Volume 8. Kluwer Academic Publishers, Dordrecht (2000)
14. Pokorný, J., Vojtás, P.: A data model for flexible querying. In: ADBIS '01: Proceedings of the 5th East European Conference on Advances in Databases and Information Systems, London, UK, Springer-Verlag (2001) 280–293
15. Skopal, T., Lokoč, J.: Nm-tree: Flexible approximate similarity search in metric and non-metric spaces. In: DEXA '08: Proceedings of the 19th international conference on Database and Expert Systems Applications, Berlin, Heidelberg, Springer-Verlag (2008) 312–325