Invariance to Ordinal Transformations in Rank-Aware Databases

Vilem Vychodil*

Dept. Computer Science, Palacky University Olomouc

Abstract

We study influence of ordinal transformations on results of queries in rank-aware databases which derive their operations with ranked relations from totally ordered structures of scores with infima acting as aggregation functions. We introduce notions of ordinal containment and equivalence of ranked relations and prove that infima-based algebraic operations with ranked relations are invariant to ordinal transformations: Queries applied to original and transformed data yield results which are equivalent in terms of the order given by scores, meaning that top-k results of queries remain the same. We show this important property is preserved in alternative query systems based of relational calculi developed in context of Gödel logic. We comment on relationship to monotone query evaluation and show that the results can be attained in alternative rank-aware approaches.

1 Introduction

In this paper, we describe invariance to ordinal transformations in query systems which incorporate ranking of query results and allow to compare the importance or relevance of query results based on their scores. We present general observations which may be applied in various rank-aware approaches, see [25] for an extensive and systematic survey of existing approaches. In particular, we present detailed analysis in a particular relational query system where queries are expressed by arbitrary complex algebraic expressions and answered by relations with tuples annotated by scores (so-called ranked relations or ranked data tables). While we analyze the issues of ordinal transformations in one particular rank-aware model, the presented technique is indeed general and can be applied to other models which we demonstrate by showing analogous results for RankSQL proposed by [28].

The practical contribution of the results presented in our paper is in exposing transformations of input ranking criteria which do not alter results of queries. Typically, rank-aware queries may be understood as classic queries which in addition incorporate ranking criteria like “low price”, “high availability”, “close distance”, etc. Such criteria may be defined in many ways and/or may depend

*e-mail: `vychodil@binghamton.edu`, phone: +420 585 634 705, fax: +420 585 411 643

on various parameters. For instance a "distance of locations" of houses in a city may be based on a geographical distance, a road traveling distance, or it may be based on socio-economic parameters such as criminality rate and rating of schools, etc. Therefore, there is a natural question whether results of queries change if the ranking criteria are altered.

To further explain the issues studied in this paper, we can consider the motivation presented in the classic paper by Fagin [15]: We assume a query system which admits queries that can be answered by relations with tuples annotated by scores. We assume that the scores have *comparative meaning* (higher scores mean better matches) and users are interested in listing query results sorted by scores with highest scores coming first. Recall that in [15], scores of results of queries which are expressed as conjunctions of subqueries are computed using monotone and strict aggregation functions, typically triangular norms [27] on the real unit interval. For instance, consider an expression

$$\mathtt{\mathit{notExceeding}}(\mathtt{PRICE}, \mathtt{\$800,000})\ \mathtt{AND}\ \mathtt{\mathit{near}}(\mathtt{LOCATION}, \mathtt{"Old\ Palo\ Alto"}) \tag{1}$$

which may be regarded as query for houses sold at `$800,000` (or a similar price which does not exceed the value too much) in `Old Palo Alto` (or near that general area). The subqueries *notExceeding*(PRICE, $800,000) and *near*(LOCATION, "Old Palo Alto") may be understood as restrictions using ranking criteria *notExceeding* and *near* which we further call *general restriction conditions.* The result of evaluating (1) may be seen as a ranked relation which results by first evaluating the subqueries *notExceeding*(PRICE, $800,000) and *near*(LOCATION, "Old Palo Alto") which produce ranked relations as the results of subqueries and then aggregating the scores by a conjunctive aggregation function $\otimes$. If the scores come from the real unit interval, it is natural to assume that

$$a \otimes 1 = a, \tag{2}$$

$$a \otimes b = b \otimes a, \tag{3}$$

$$a \otimes (b \otimes c) = (a \otimes b) \otimes c \tag{4}$$

are satisfied for all $a, b, c \in [0, 1]$ and $\otimes$ is monotone (isotone) with respect to the usual ordering of reals:

$$\text{if } a \leq b, \text{ then } a \otimes c \leq b \otimes c \tag{5}$$

for all $a, b, c \in [0, 1]$. Functions satisfying such conditions are called triangular norms [27] and may be understood as generalizations of (truth functions of) the classic conjunction [17].

The main concern of [15] are algorithms for efficient computation of top-k results of queries like (1). Interestingly, the paper shows a simplification of the main algorithm for returning the top-k answers of monotone queries in case the utilized aggregation function $\otimes$ coincides with $\min(x, y)$. In this paper, we show that the choice of minimum as the basic conjunctive aggregation function has another important (and desirable) consequence:

Consequence 1. *Top-k results of queries do not change if ordinal transformations are applied to the input data and all restriction conditions which appear in queries.*

By an ordinal transformation we mean a transformation which modifies scores but preserves the order of tuples given by the scores (a precise definition follows in the paper). We call Consequence 1 the *invariance to ordinal transformations.* It can be shown that the consequence does not hold in case of general aggregation functions. Therefore, Consequence 1 describes an important property of a particular query system supporting top-k queries. Systems supporting top-k queries are recently gaining interest [29, 32, 36] and investigations in this direction may be exploited as optimization techniques. For instance, in case of reiterated queries executed with different parameters, observations like Consequence 1 may help identify that certain changes in input parameters will have no influence on the results of top-k queries. Such observations are beneficial especially in case of processing large data collections [6].

We investigate the invariance to ordinal transformations independently on the chosen structure of scores. Instead of assuming a subset of reals with its natural ordering as the set of scores, we assume that the set of scores forms a totally ordered set where infima (greatest lower bounds) exist for arbitrary subsets of scores. Note that in this setting, an infimum of a finite non-empty set of scores is its minimum element with respect to the total order of scores. Considering such general structures of scores, we introduce algebraic operations on ranked relations. The operations include the join, restriction, projection, union, difference, residuum, and division and they can be seen as particularizations of operations used in [4] considering the operation of infimum as the aggregation function. Such a set of operations is adequate for formulating complex queries, including non-monotone ones. Using the proposed operations, queries like (1) may be regarded as particular joins (or intersections) of general restrictions. The results elaborated in Section 4 and Section 5 show properties of ordinal transformations and the invariance theorems.

Let us stress one practical aspect about the consequences of the invariance theorems:

Consequence 2. *When using infima-based algebraic operations with ranked relations, the scores in ranked tables have no quantitative meaning.*

In other words, the meaning of scores is *purely comparative.* For instance, if a ranked relation consists of exactly two tuples with ranks a and b such that $a < b$, then any kind of distance or closeness of a and b is irrelevant; $a = 0.3$ and $b = 0.9$ represent the same relationship as $a = 0.89$ and $b = 0.9$ because in both cases $a < b$. In fact, considering the generality of our structures of scores, we may replace the numerical scores by symbolic ones as long as the order of the corresponding scores preserves (and reflects) the order of the numerical scores. For instance, instead of numerical values 0, 0.8, and 1, one can use symbolic names "not at all", "more or less", and "fully" provided that the order $<$ is defined as "not at all" $<$ "more or less" $<$ "fully". As an immediate consequence for users of a database system implementing infima-based operations is that the scores can be completely hidden from users since their values (and their mutual similarity) do not represent any quantitative information.

Our paper presents an order-theoretic treatment of invariance issues in ranked-aware databases which are traditionally studied form the point of view of query execution. Let us note that according to a taxonomy introduced in [25], in this paper we work with a model which uses *exact methods over certain data.* Indeed, we would like to stress that unlike the approaches to probabilistic databases [9] which also contain explicit scores, we are concerned with *certain data*. Possible extensions of our observations to models for uncertain data should prove interesting but it is not the objective of our paper.

The present paper is related to our previous analysis of preservation of similarity of query results for similar input data [3] where the similarity is formalized as closeness of scores without considering the issues of order preservation. Indeed, [3] introduces formulas for expressing lower bounds of similarity of query results performed with pairwise similar input data. The notion of similarity (of input data and results of queries) in [3] is based on residuated implications [20] and captures the fact that ranked relations consist of tuples with similar scores in terms of their closeness. As such, the notion does not express the fact that the order of tuples is preserved. In Section 5, we present notes on how this approach can be combined with the present one. Also note that issues related to ordinal transformations of object-attribute data were studied from the point of view of formal concept analysis [18] in [2] where the author shows that ordinally equivalent input data induce almost isomorphic concept lattices.

Our paper is organized as follows. Section 2 presents preliminaries from partially ordered sets and lattices which are used in our paper as the basic structures of scores. Section 3 describes the rank-aware model for which we make the analysis of the invariance to ordinal transformations. Section 4 is devoted to the properties of order equivalence of ranked relations which plays an important role in the analysis. Section 5 contains the invariance and additional discussion. Section 6 shows how our results relate to a relational calculus developed in the context of Gödel logic. Section 7 discusses issues of efficient query evaluation which arise in the model and comments on the relationship to other approaches.

2 Preliminaries

In this section, we recall preliminary notions of partially ordered sets and lattices. The notions are used in further sections to formalize structures of scores which are used in the considered model of data. More details on the notions presented in this section can be found in [5].

A partial order on a non-empty set L is a binary relation $\leq$ on L which is reflexive ($a \leq a$), antisymmetric ($a \leq b$ and $b \leq a$ yield $a = b$), and transitive ($a \leq b$ and $b \leq c$ yield $a \leq c$). A pair $\mathbf{L} = \langle L, \leq \rangle$ where $\leq$ is a partial order on L is called a partially ordered set (shortly, a poset). A partial order $\leq$ on L is called a total (or a linear) order whenever for any $a, b \in L$, we have $a \leq b$ or $b \leq a$ in which case $\mathbf{L} = \langle L, \leq \rangle$ is called a totally ordered set or a chain.

An element $a \in L$ is called the least element of $K \subseteq L$ in $\mathbf{L} = \langle L, \leq \rangle$ whenever $a \leq b$ for all $b \in K$. Dually, we consider the notion of a greatest element. If both the least and the greatest elements exist for the whole L, we say that $\mathbf{L}$ is bounded and denote the fact by $\mathbf{L} = \langle L, \leq, 0, 1 \rangle$ where 0 and 1 stand for the least and the greatest element in $\mathbf{L}$, respectively.

For subsets of a partially ordered set $\mathbf{L} = \langle L, \leq \rangle$, we consider their greatest lower bounds and least upper bounds as follows: For $K \subseteq L$, we put

$$\operatorname{lc} K = \{a \in L;\, a \leq b \text{ for all } b \in K\}, \tag{6}$$

$$\operatorname{uc} K = \{b \in L;\, a \leq b \text{ for all } a \in K\}, \tag{7}$$

and call $\operatorname{lc} K$ and $\operatorname{uc} K$ the lower and upper cones of K in $\mathbf{L}$, respectively. If $\operatorname{lc} K$ has the greatest element, it is called the infimum (the greatest lower bound) of K in $\mathbf{L}$ and denoted by $\inf K$. Dually, if $\operatorname{uc} K$ has the least element, it is called the supremum (the least upper bound) of K in $\mathbf{L}$ and denoted by $\sup K$. If for any $a, b \in L$, the elements $\inf\{a, b\}$ and $\sup\{a, b\}$ exist, then $\leq$ is called a lattice order and $\mathbf{L}$ is called a lattice ordered set (shortly, a lattice). Each totally ordered set is a lattice because if $a \leq b$, then obviously $\inf\{a, b\} = a$ and $\sup\{a, b\} = b$. In addition, if $\mathbf{L}$ is totally ordered then for any non-empty and finite K, it follows that $\inf K$ and $\sup K$ coincide with the least and greatest elements in K, respectively. If for any $K \subseteq L$, the elements $\inf K$ and $\sup K$ exist, then $\leq$ is called a complete lattice order and $\mathbf{L}$ is called a complete lattice ordered set (shortly, a complete lattice). Each complete lattice is a bounded lattice because $\inf L = \sup \emptyset = 0$ (the least element of $\mathbf{L}$) and $\inf \emptyset = \sup L = 1$ (thre greatest element of $\mathbf{L}$). In general, a (bounded) lattice may not be complete (consider a subset of reals $L = [0, 1] \setminus \{0.5\}$ equipped with the usual ordering $\leq$ of reals).

There is an alternative view of partially ordered sets and lattices via algebraic structures: Let $\mathbf{L} = \langle L, \sqcap, \sqcup \rangle$ be an algebra with two binary operations $\sqcap$ (called a meet) and $\sqcup$ (called a join) such that both $\sqcap$ and $\sqcup$ are commutative, associative, idempotent (i.e., $a \sqcap a = a \sqcup a = a$ for any $a \in L$), and satisfy the laws of absorption: $a \sqcap (a \sqcup b) = a$ and $a \sqcup (a \sqcap b) = a$ for all $a, b \in L$, then the algebra $\mathbf{L}$ is called a lattice. It can be easily shown that for a partially ordered set $\mathbf{L} = \langle L, \leq \rangle$ which is a lattice in the order-theoretic sense, we can consider an algebra on L with $a \sqcap b = \inf\{a, b\}$ and $a \sqcup b = \sup\{a, b\}$ which is a lattice in the latter sense. Conversely, for a lattice $\mathbf{L} = \langle L, \sqcap, \sqcup \rangle$, we may introduce $a \leq b$ iff $a = a \sqcap b$ (or, equivalently, $a \sqcup b = b$). As a consequence, we may understand lattices as both special partially ordered structures and special algebras. In the paper, whenever we consider a (complete) lattice $\mathbf{L}$, we automatically consider the lattice order $\leq$ and treat inf and sup as operations on L.

In the follows sections, we assume that scores we use to annotate tuples in relations come from a bounded totally ordered set $\mathbf{L} = \langle L, \leq, 0, 1 \rangle$ and, optionally, we assume that $\mathbf{L}$ is in addition a complete lattice. The operations inf and sup are used to obtain greatest lower and least upper bounds of finite (or arbitrary if $\mathbf{L}$ is complete) subsets of scores. For instance, if one considers conjunctive (or disjunctive) queries consisting of several subqueries, inf (or sup) is used to aggregate scores from the subqueries to obtain a score for the composed query. Details are discussed in Section 3.

Remark 1. As a borderline case of complete totally ordered lattices we may take the two-element Boolean algebra which is uniquely given up to isomorphism. That is, for $L = \{0, 1\}$ where 0 denotes the truth value "false" and 1 denotes the truth value "true" and putting $0 \leq 1$, we obain a complete totally ordered lattice where inf and sup coincide with the truth functions of the classic logical connectives "conjunction" and "disjunction". In addition, a truth function of negation may be introduced as a complement, i.e., $0' = 1$ and $1' = 0$ and the resulting structure $\mathbf{L} = \langle L, \inf, \sup,', 0, 1\rangle$ is a two-element Boolean algebra. From the point of view of the ranked approach used in this paper, 0 and 1 may be seen as two borderline scores—0 represents a mismatch while 1 represents a match. Because of the well-known property of functional completeness of Boolean algebras, every n-ary truth function may be expressed by means of terms consisting of variables and inf, sup, and $'$. For instance, a truth function for implication (logical conditional) can be introduced as $a \to b = \sup\{a', b\}$.

Considering Remark 1 and the fact that general bounded totally ordered sets serve as structures of ranks, inf and sup may be seen as generalizations of truth functions of logical connectives "conjunction" and "disjunction". A natural question is whether we can obtain analogies of truth functions of other important locical connectives like the negation and implication. This question is important because in the classic relational model such connectives as crucial for expressing many relational operations like the difference, semidifference, and division which cannot be expressed just using inf and sup. We therefore consider additional binary connectives which are adjoint to inf and sup and serve as generalizations of truth functions of logical connectives "implication" and "non-implication" (so-called abjunction): For a bounded lattice $\mathbf{L} = \langle L, \leq, 0, 1\rangle$, consider a binary operaton $\to$ such that

$$\inf\{a, b\} \leq c \text{ iff } a \leq b \to c \tag{8}$$

holds true for all $a, b, c \in L$. Note that $\to$ may not exist but if it exists for given $\mathbf{L}$, then it is given uniquely. In the terminology or ordered sets, $\to$ is called a relative pseudo-complement or a residuum. Alternatively, $\to$ may be introduced as a binary operation on $\mathbf{L}$ which satisfies the following conditions

$$a \to a = 1, \tag{9}$$

$$\inf\{a, a \to b\} = \inf\{a, b\}, \tag{10}$$

$$\inf\{a \to b, b\} = b, \tag{11}$$

$$a \to \inf\{b, c\} = \inf\{a \to b, a \to c\}, \tag{12}$$

for every $a, b, c \in L$. The resulting structure $\mathbf{L} = \langle L, \inf, \sup, \to, 0, 1\rangle$ is called a Heyting algebra. Note that Heyting algebras are used as semantic structures of the intuitionistic logic [24]. That is, in the intuitionistic logic, they play an analogous role as the Boolean algebras in the classic logic. If $\mathbf{L}$ satisfies the following additional condition

$$\sup\{a \to b, b \to a\} = 1 \tag{13}$$

for all $a, b \in L$, then it is called a Gödel algebra. Analogously as the Heyting algebras are the semantic structures of the intuitionistic logic, Gödel algebras are semantic structures of Gödel logic which is a stronger logic than the intuitionistic logic but it is not as strong as the Boolean logic. According to [23], Gödel logic may be seen as a schematic extension of the Basic logic.

Condition (8) is called the adjointness property of inf and $\rightarrow$. It ensures that $\rightarrow$ is a faithful truth function of a general implication. In particular, if $\mathbf{L}$ is totally ordered, we get that

$$a \rightarrow b = \begin{cases} 1, & \text{if } a \leq b, \\ b, & \text{otherwise,} \end{cases} \tag{14}$$

for all $a, b \in L$. Therefore, if we restrict ourselves just to $\{0,1\} \subseteq L$, we get $1 \rightarrow 0 = 0$ and $0 \rightarrow 0 = 0 \rightarrow 1 = 1 \rightarrow 1 = 1$, i.e., on $\{0,1\}$, $\rightarrow$ acts as a truth function of the classic implication. In general, we have $a \rightarrow b = 1$ iff $a \leq b$. Now, a generalization of the classic negation and equivalence (logical biconditional) can be introduced by

$$\neg a = a \rightarrow 0, \tag{15}$$

$$a \leftrightarrow b = \inf\{a \rightarrow b, b \rightarrow a\}, \tag{16}$$

for all $a, b \in L$. Taking (14) into account, we have

$$\neg a = \begin{cases} 1, & \text{for } a = 0, \\ 0, & \text{otherwise,} \end{cases} \tag{17}$$

$$a \leftrightarrow b = \begin{cases} 1, & \text{for } a = b, \\ \inf\{a, b\}, & \text{otherwise,} \end{cases} \tag{18}$$

where the $\inf\{a, b\}$ in (18) is in fact a minimum of a and b since all elements in a totally ordered $\mathbf{L}$ are comparable.

Analogously as $\rightarrow$ is adjoint to inf in sense of (8), we may apply the duality principle and introduce a generalization of logical non-implication which is adjoint to sup. Namely, following the ideas of [34], see also [33], we may consider a binary operation $\ominus$ such that

$$a \ominus b \leq c \text{ iff } a \leq \sup\{b, c\} \tag{19}$$

for all $a, b, c \in L$. Alternatively, we may postulate the following equalities:

$$a \ominus a = 0, \tag{20}$$

$$\sup\{a \ominus b, b\} = \sup\{a, b\}, \tag{21}$$

$$\sup\{a, a \ominus b\} = a, \tag{22}$$

$$\sup\{a, b\} \ominus c = \sup\{a \ominus c, b \ominus c\}, \tag{23}$$

for every $a, b, c \in L$. Analogously as in the case of $\rightarrow$, if $\mathbf{L}$ is totally ordered, it follows that

$$a \ominus b = \begin{cases} 0, & \text{if } a \leq b, \\ a, & \text{otherwise,} \end{cases} \tag{24}$$

for all $a, b \in L$. Therefore, $\ominus$ is indeed a generalization of a non-implication (a logical difference bounded by 0 and 1) because on $\{0,1\} \subseteq L$, we have $0 \ominus 0 = 0 \ominus 1 = 1 \ominus 1 = 0$ and $1 \ominus 0 = 1$.

In the following sections, we utilize $\rightarrow$ in the definitions of rank-aware relational containment and division and $\ominus$ is utilized in a rank-aware relational difference.

If $\mathbf{L}$ is a totally ordered Gödel algebra which is defined on the real unit interval (with $\leq$ being the usual ordering of reals), then we call $\mathbf{L}$ the standard Gödel algebra [23] and denote it by $[0,1]_{\mathbf{G}}$.

3 Rank-Aware Relational Model of Data

In this section, we introduce a relational rank-aware model of data. Namely, we describe structures formalizing data tables which appear in the model and relational operations which constitute the core of relational queries that take scores into account. The model may be viewed as a particularization of a model based on complete residuated lattices which has been outlined in [4] which results by a choice of special structures of scores based on Gödel algebras described in Section 2.

First, we recall the basic notions which appear in the (classic) relational model of data. In the paper we consider *relation schemes* as finite sets of *attributes.* We tacitly identify attributes with their names, i.e., attributes are considered as "names of columns" in data tables. As usual, we assume that each attribute in a relation scheme has its *type* which defines a (possibly infinite but at most denumerable) set of admissible values for the attribute. We write $u = v$ whenever two value u, v of a particular type are indistinguishable and $u \neq v$ otherwise. *Tuples*, which formalize "rows in data tables" are considered as maps assigning to each attribute from relation schemes a value of its type; we denote by $r(y)$ the *y-value* of tuple r. Furthermore, we denote by $\mathrm{Tupl}(R)$ the *set of all tuples* on the relation scheme R. Again, note that $\mathrm{Tupl}(R)$ may be infinite. For $r_1, r_2 \in \mathrm{Tupl}(R)$, we put $r_1 = r_2$ whenever $r_1(y) = r_2(y)$ for all $y \in R$ and $r_1 \neq r_2$ otherwise. Tuples $r \in \mathrm{Tupl}(R)$ and $s \in \mathrm{Tupl}(S)$ are called *joinable* whenever $r(y) = s(y)$ for all $y \in R \cap S$. If $r \in \mathrm{Tupl}(R)$ and $s \in \mathrm{Tupl}(S)$ are joinable, then rs, called the *join* of r and s, is a tuple in $\mathrm{Tupl}(R \cup S)$ such that $(rs)(y) = r(y)$ for $y \in R$ and $(rs)(y) = s(y)$ for $y \in S$.

Remark 2. Note that the join of tuples is also called a concatenation and it may be seen as a set-theoretic union of tuples since tuples are considered as sets of attribute-value pairs, see [30]. In a special case for $R = \emptyset$ (the empty relation scheme), $\mathrm{Tupl}(R)$ consists of a single tuple—the empty tuple which, according to the set-theoretic representation of tuples, may be identified with the empty set. Thus, we write $\mathrm{Tupl}(\emptyset) = \{\emptyset\}$.

Let $\mathbf{L} = \langle L, \leq, 0, 1\rangle$ be a totally ordered complete lattice. The elements in L are called *scores.* The scores have a comparative meaning. That is, if $a < b$ for $a, b \in L$, then b is a score of a better match than a. As a consequence, 1 is the score of a best match (a full match) and 0 is the score of a worst match (no match). Considering $\mathbf{L}$ as the structure of scores, a *ranked data table* (shortly, an RDT) $\mathcal{D}$ on relation scheme R which uses scores in $\mathbf{L}$ is understood as a map

$$\mathcal{D}\colon \mathrm{Tupl}(R) \rightarrow L \tag{25}$$

such that $\{r \in \mathrm{Tupl}(R); \mathcal{D}(r) > 0\}$, called the *answer set* of $\mathcal{D}$, is finite. That is, only finitely many tuples in $\mathrm{Tupl}(R)$ are assigned non-zero scores by an RDT $\mathcal{D}$ on R; $\mathcal{D}(r)$ is the score of tuple r

#	ID	BDRM	SQFT
1.000	85	5	4580
0.971	56	3	3400
0.937	71	3	3280
0.643	82	4	2350
0.426	58	4	1760
0.148	93	2	1130

#	ID	AGENT	PRICE
0.997	71	Black	798,000
0.964	58	Black	829,000
0.940	71	Adams	849,000
0.798	45	Adams	654,000
0.789	82	Adams	648,000
0.778	85	Black	998,000
0.708	45	Black	598,000
0.708	93	Black	598,000

Figure 1: Examples of ranked tables: Houses with area roughly greater than `3,500` sq. ft. (left), houses offered by agents for about \$800,000 (right).

in RDT $\mathcal{D}$. RDTs defined on non-empty relation schemes may be represented by two-dimensional tables with rows corresponding to tuples from the answer set, columns corresponding to attributes, and an extra column (denoted by `#`) containing the scores. For illustration, if $L = [0, 1]$ and $\leq$ is the usual order of reals, the tables in Fig. 1 may be viewed as ranked data tables with scores in $L = [0, 1]$. In the figure, the tuples from the answer set are sorted according to their scores.

In the borderline case of $R = \emptyset$, the answer set of $\mathcal{D}$ on R contains at most the empty tuple $\emptyset$. If the answer set is empty then clearly $\mathcal{D}(\emptyset) = 0$. Otherwise, $\mathcal{D}$ is uniquely given by the non-zero score $\mathcal{D}(\emptyset) \in L$. Note that this naturally generalizes the two borderline relations on the empty relation scheme R which appear in the classic model: the empty relation on R and the relation on R containing the empty tuple.

Furthermore, we consider equality of RDTs as follows: For RDTs $\mathcal{D}_1$ and $\mathcal{D}_2$ on R we put $\mathcal{D}_1 = \mathcal{D}_2$ whenever $\mathcal{D}_1(r) = \mathcal{D}_2(r)$ for all $r \in \mathrm{Tupl}(R)$, i.e., whenever $\mathcal{D}_1$ and $\mathcal{D}_2$ are equal as maps. The *range* (*or scores*) of RDT $\mathcal{D}$ on relation scheme R, denoted $L(\mathcal{D})$, is a subset of L defined by

$$L(\mathcal{D}) = \{\mathcal{D}(r);\, r \in \mathrm{Tupl}(R)\}. \tag{26}$$

That is, $L(\mathcal{D})$ is the set of all scores from L which appear in $\mathcal{D}$. Thus, $L(\mathcal{D})$ is finite for any $\mathcal{D}$. Let us note here that if $L(\mathcal{D}) \subseteq \{0, 1\}$, then $\mathcal{D}$ may be viewed as a ranked representation of a classic relation on a relation scheme. Indeed, for an ordinary (finite) $\mathcal{R} \subseteq \mathrm{Tupl}(R)$, we can introduce a corresponding RDT $\mathcal{D}_{\mathcal{R}}$ by putting $\mathcal{D}_{\mathcal{R}}(r) = 1$ whenever $r \in \mathcal{R}$ and $\mathcal{D}_{\mathcal{R}}(r) = 0$ otherwise. Conversely, for RDT $\mathcal{D}$ on R, we may consider a corresponding $\mathcal{R}_{\mathcal{D}} \subseteq \mathrm{Tupl}(R)$ as $\mathcal{R}_{\mathcal{D}} = \{r;\, \mathcal{D}(r) = 1\}$. Taking into account just RDTs with ranges being subsets of $\{0, 1\}$, the two transformations are mutually inverse. As a consequence, in the same spirit as in the Codd model [8], RDTs may represent both the results of queries and base data, i.e., our approach uses only a single type of structures. As a consequence, we do not mix the classic relations and the ranked data tables.

Remark 3. (a) The fundamental notion of a ranked data table may seem like a digression from the relational model of data and in particular from its modern understanding as it is described in *The Third Manifesto* (*TTM*, see [10]) because tuples in relations are annotated by an additional information which is the score. If one wishes the approach to adhere to *TTM*, he can consider the score as an additional attribute (named #) which is present in the relation scheme. The type of the attribute # is *score*. In other words, RDTs may be seen as ordinary relations on relation schemes with a special designated attribute # the values of which come from the universe of $\mathbf{L}$.

(b) As we have mentioned in the introduction, our model is not related to probabilistic databases which are currently extensively studied. In particular, the scores *cannot* be interpreted as probabilities. Let us note that the scores need not come from a real unit interval, so in general it does not make sense to consider the scores as probability values. Even if the scores do come from a unit interval, their values are not related to probabilities assigned to any events because there is no uncertainty involved in the data or in query evaluation as we shall se later.

(c) Note that various approaches where tuples in relations are annotated by values coming from general algebraic structures exist. Most notably, the authors of [26] consider conditional tables which may be understood as relations with tuples annotated by Boolean formulas, i.e., annotated by values coming from particular free Boolean algebras. A general approach to relations annotated by element from semi-rings is presented in [22], see also [16], [21], and [1].

We now describe a set of relational operations which are used to express queries over ranked data tables. Important types of monotone as well as non-monotone queries in rank-aware databases may be expressed by a combination of the following operations with RDTs which generalize their classic relational operations in the original relational model of data. For the introduced operations, we adopt the widely used Codd-style notation.

Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be RDTs on $R \cup S$ and $S \cup T$ with $R \cap S = \emptyset$, $R \cap T = \emptyset$, and $S \cap T = \emptyset$. The (*natural*) *join* of $\mathcal{D}_1$ and $\mathcal{D}_2$, denoted $\mathcal{D}_1 \bowtie \mathcal{D}_2$, is defined by

$$(\mathcal{D}_1 \bowtie \mathcal{D}_2)(rst) = \inf\{\mathcal{D}_1(rs), \mathcal{D}_2(st)\}, \tag{27}$$

for all $r \in \mathrm{Tupl}(R)$, $s \in \mathrm{Tupl}(S)$, and $t \in \mathrm{Tupl}(T)$. Recall that rs, st and rst in (27) denote the results of joins of tuples which are in this case trivially joinable. Since $\mathbf{L}$ is totally ordered, the score $(\mathcal{D}_1 \bowtie \mathcal{D}_2)(rst)$ in (27) is in fact taken as the minimum of the scores $\mathcal{D}_1(rs)$ and $\mathcal{D}_2(st)$. Also note that the commutativity, associativity, and idempotency of inf implies that $\bowtie$ has these properties as well. In addition, any $\mathcal{D}$ (over any R) with an empty answer set is an annihilator with respect to $\bowtie$ and $\mathcal{D}$ on $\emptyset$ such that $\mathcal{D}(\emptyset) = 1$ is a neutral element with respect to $\bowtie$. The join of the illustrative RDTs in Fig. 1 is shown in Fig. 2.

Remark 4. As we have noted in the introduction, the operations with RDTs we use in this paper may be viewed as particular cases of those used in [4]. In [4], the basic structures of scores are complete residuated lattices [17] which may be viewed as generalization of the structures of scores defined on the real unit interval by left-continuous triangular norms [14]. The general counterpart to (27) in [4] is $\bowtie_{\otimes}$ defined by

$$(\mathcal{D}_1 \bowtie_{\otimes} \mathcal{D}_2)(rst) = \mathcal{D}_1(rs) \otimes \mathcal{D}_2(st) \tag{28}$$

for all $r \in \mathrm{Tupl}(R)$, $s \in \mathrm{Tupl}(S)$, and $t \in \mathrm{Tupl}(T)$. Obviously, (27) is a particular case of (28) with $\otimes$ being inf, i.e., $a \otimes b = \inf\{a, b\}$ for all $a, b \in L$. Further in the paper we show that joins defined

#	ID	BDRM	SQFT	AGENT	PRICE
0.937	71	3	3280	Adams	849,000
0.937	71	3	3280	Black	798,000
0.778	85	5	4580	Black	998,000
0.643	82	4	2350	Adams	648,000
0.426	58	4	1760	Black	829,000
0.148	93	2	1130	Black	598,000

Figure 2: Join of ranked tables from Figure 1.

#	ID	BDRM	SQFT	AGENT	PRICE
0.939	71	3	3280	Black	798,000
0.938	71	3	3280	Adams	849,000
0.778	85	5	4580	Black	998,000
0.643	82	4	2350	Adams	648,000
0.426	58	4	1760	Black	829,000
0.148	93	2	1130	Black	598,000

Figure 3: RDT "similar" to that in Figure 2.

by (27) are invariant to ordinal transformations provided that totally ordered Gödel algebras are used as structures of scores. We also show that the property does not hold in the general setting of complete residuated lattices. A similar remark can be made for all the operations introduced below.

We introduce restrictions (selections) of RDTs utilizing general maps serving as restriction conditions: By a *restriction condition* on R we mean any map $\theta\colon \mathrm{Tupl}(R) \to L$ with each $\theta(r)$ interpreted as the score expressing whether (and to what degree) tuple r matches θ. Note that the ordinary restriction conditions based on classic comparators of domain values are covered by this general notion. For instance, if $\theta\colon \mathrm{Tupl}(R) \to L$ is defined so that $\theta(r) = 1$ whenever $r(y_1) = r(y_2)$ and $\theta(r) = 0$ otherwise, then θ may be seen as representing a classic restriction condition based on equality of the values of attributes y_1 and y_2.

Given RDT $\mathcal{D}$ on R, we define the *restriction* of $\mathcal{D}$ using θ on R by

$$(\sigma_\theta(\mathcal{D}))(r) = \inf\{\mathcal{D}(r), \theta(r)\} \tag{29}$$

for all $r \in \mathrm{Tupl}(R)$. Obviously, the score of r in $\sigma_\theta(\mathcal{D})$ at most as high as its score in $\mathcal{D}$ which is a natural property of a restriction. Note that the ranked tables in Fig. 1 may be seen as results of particular restrictions of base data tables with all scores (of tuples present in the tables) set to 1.

For $\mathcal{D}$ on R and $S \subseteq R$, the *projection* $\pi_S(\mathcal{D})$ of $\mathcal{D}$ onto S is defined by

$$(\pi_S(\mathcal{D}))(s) = \sup\{\mathcal{D}(st);\, t \in \mathrm{Tupl}(R \setminus S)\} \tag{30}$$

for all $s \in \mathrm{Tupl}(S)$. Here, notice the use of sup instead of inf which corresponds to the close relationship of projections and existentially quantified queries. Recall that in the classic setting, the fact that s belongs to a projection of a relation onto S means that *there exists* t such that st is in the relation. In a similar sense, the score of s in $\pi_S(\mathcal{D})$ is defined by (30) as the highest score of st over all t (note that two different tuples in the answer set of $\mathcal{D}$ may be projected onto the same tuple on S).

For $\mathcal{D}_1$ and $\mathcal{D}_2$ on the same relation scheme R, we introduce the *union* of $\mathcal{D}_1$ and $\mathcal{D}_2$ which is defined componentwise using sup as

$$(\mathcal{D}_1 \cup \mathcal{D}_2)(r) = \sup\{\mathcal{D}_1(r), \mathcal{D}_2(r)\}, \tag{31}$$

for all $r \in \mathrm{Tupl}(R)$. In addition, we may consider an intersection based on inf but this operation is superfluous because it can be understood as a join (27) of two RDTs on the same relation scheme.

Since $\mathbf{L}$ is linearly ordered, $\ominus$ which is adjoint to sup as in (19) exists and it is given by (24). Therefore, for $\mathcal{D}_1$ and $\mathcal{D}_2$ on the same relation scheme R, we may introduce the *difference* of $\mathcal{D}_1$ and $\mathcal{D}_2$ by

$$(\mathcal{D}_1 - \mathcal{D}_2)(r) = \mathcal{D}_1(r) \ominus \mathcal{D}_2(r), \tag{32}$$

for all $r \in \mathrm{Tupl}(R)$, i.e., using (24) and the total ordering of $\mathbf{L}$,

$$(\mathcal{D}_1 - \mathcal{D}_2)(r) = \begin{cases} 0, & \text{if } \mathcal{D}_1(r) \leq \mathcal{D}_2(r), \\ \mathcal{D}_1(r), & \text{otherwise,} \end{cases} \tag{33}$$

for all $r \in \mathrm{Tupl}(R)$.

Finally, we consider operations with RDTs which are related to universally quantified queries. In the classic model, queries of the form of categorical propositions "every φ is ψ" may be expressed by divisions (or more general constructs such as the imaging operator considered in [11]) which are in the classic model expressible by means of other operations (joins, projections, and difference). From the logical point of view this is a consequence of the fact that the universal quantifier is definable using negations and the existential quantifier. As we shall see in Section 6, this property does not hold in a weaker logic which is closely related to the rank-aware model. Therefore, in our case, we have to introduce an operation in order to be able to properly express queries of the form of categorical propositions "every φ is ψ". In our case, such an operation will be a variant of the Small Divide as it is considered in [12].

Note that analogously as in the case of $\ominus$, the residuum $\rightarrow$ satisfying (8) always exists and is uniquely given by (14) owing to the linearity of $\mathbf{L}$. Let $\mathcal{D}_1$ (so-called mediator) be an RDT on $R \cup S$ such that $R \cap S = \emptyset$, $\mathcal{D}_2$ (so-called divisor) be an RDT on S, and let $\mathcal{D}_3$ (so-called dividend) be an RDT on R. In this setting, we introduce a *division* $\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2$ as an RDT on R such that

$$\bigl(\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2\bigr)(r) = \inf\{\mathcal{D}_2(s) \rightarrow \mathcal{D}_1(rs);\, s \in \mathrm{Tupl}(S)\} \cup \{\mathcal{D}_3(r)\}, \tag{34}$$

for all $r \in \mathrm{Tupl}(R)$. Directly from (34), the answer set of $\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2$ is finite since it is a subset of the answer set of $\mathcal{D}_3$. By moment's reflection, we can see that (34) can equivalently be written as

$$\bigl(\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2\bigr)(r) = \inf\{\inf\{\mathcal{D}_3(r), \mathcal{D}_2(s) \rightarrow \mathcal{D}_1(rs)\};\, s \in \mathrm{Tupl}(S)\}. \tag{35}$$

Observe that according to (35) and (14), $\big(\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2\big)(r) = \mathcal{D}_3(r)$ iff

$$\text{for all } s \in \mathrm{Tupl}(S)\colon\ \mathcal{D}_2(s) > \mathcal{D}_1(rs) \text{ implies } \mathcal{D}_3(r) \leq \mathcal{D}_1(rs). \tag{36}$$

Therefore, we can distinguish two cases as follows:

$$\big(\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2\big)(r) = \begin{cases} \mathcal{D}_3(r), & \text{if (36) holds,} \\ \inf\{\mathcal{D}_2(s) \rightarrow \mathcal{D}_1(rs);\, s \in \mathrm{Tupl}(S)\}, & \text{otherwise.} \end{cases} \tag{37}$$

Using (14) again, we have

$$\big(\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2\big)(r) = \begin{cases} \mathcal{D}_3(r), & \text{if (36) holds,} \\ \inf\{\mathcal{D}_1(rs);\, \mathcal{D}_2(s) > \mathcal{D}_1(rs),\, s \in \mathrm{Tupl}(S)\}, & \text{otherwise.} \end{cases} \tag{38}$$

As a consequence, the rank of a tuple in the result of a division can always be computed in finitely many steps because each divisor has a finite answer set.

Closely related to the division is the notion of a subsethood (inclusion of RDTs) which, in our case, can also be expressed by a score. Namely, for RDTs $\mathcal{D}_1$ and $\mathcal{D}_2$ on the same relation scheme R, we put

$$\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2) = \inf\{\mathcal{D}_1(r) \rightarrow \mathcal{D}_2(r);\, r \in \mathrm{Tupl}(R)\} \tag{39}$$

$$= \inf\{\mathcal{D}_2(r);\, \mathcal{D}_1(r) > \mathcal{D}_2(r) \text{ and } r \in \mathrm{Tupl}(R)\} \tag{40}$$

and call $\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2)$ the *subsethood score* of $\mathcal{D}_1$ in $\mathcal{D}_2$. That is, S is not a relational operation because its result is a score in $\mathbf{L}$ (and not an RDT). The subsethood scores generalize the concept of containment of relations. Indeed, if ranked tables $\mathcal{D}_1$ and $\mathcal{D}_2$ are considered as results of queries Q_1 and Q_2, then $\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2)$ is the score expressing the degree to which "if a tuple satisfies Q_1, then it satisfies Q_2" is satisfied by all tuples. In particular, it is easily seen that $\mathcal{D}_1 = \mathcal{D}_2$ iff $\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2) = \mathrm{S}(\mathcal{D}_2, \mathcal{D}_1) = 1$. Subsethood scores are related to division as follows: For $R = \emptyset$ and $\mathcal{D}_1$ and $\mathcal{D}_2$ being RDTs on S, we get that $\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2) = (\mathcal{D}_2 \div^{\mathcal{D}} \mathcal{D}_1)(\emptyset)$, where $\mathcal{D}$ is the RDT on $\emptyset$ such that $\mathcal{D}(\emptyset) = 1$.

Example 1. If we consider the RDTs in Fig. 2 and Fig. 3 and denote them as $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively, then $\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2) = 1$ because all scores in $\mathcal{D}_1$ are lower than or equal to the scores of the corresponding tuples in $\mathcal{D}_2$ (note that here we use the fact that $\inf \emptyset = 1$), i.e., we may say that, taking the scores into account, $\mathcal{D}_1$ is *fully included* in $\mathcal{D}_2$. On the contrary, $\mathrm{S}(\mathcal{D}_2, \mathcal{D}_1) < 1$. Namely, $\mathrm{S}(\mathcal{D}_2, \mathcal{D}_1) = \inf\{0.937\} = 0.937$. Analogously as the subsethood scores, we may consider a related notion of a *similarity score* $\mathrm{E}(\mathcal{D}_1, \mathcal{D}_2)$ of $\mathcal{D}_1$ and $\mathcal{D}_2$ defined as

$$\mathrm{E}(\mathcal{D}_1, \mathcal{D}_2) = \inf\{\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2), \mathrm{S}(\mathcal{D}_2, \mathcal{D}_1)\}. \tag{41}$$

In this case, $\mathrm{E}(\mathcal{D}_2, \mathcal{D}_1) = 0.937$.

Let us note that we can introduce a ternary operation with RDTs which is defined componentwise using $\rightarrow$ in a similar way as the union of RDTs which is defined componentwise using sup: For $\mathcal{D}_1$, $\mathcal{D}_2$, and $\mathcal{D}_3$ on the same relation scheme R, we put

$$\big(\mathcal{D}_1 \rightarrow^{\mathcal{D}_3} \mathcal{D}_2\big)(r) = \inf\{\mathcal{D}_3(r), \mathcal{D}_1(r) \rightarrow \mathcal{D}_2(r)\} \tag{42}$$

for all $r \in \mathrm{Tupl}(R)$ and call $\mathcal{D}_1 \rightarrow^{\mathcal{D}_3} \mathcal{D}_2$ the $\mathcal{D}_3$-*residuum* of $\mathcal{D}_1$ with respect to $\mathcal{D}_2$. Note that the operation is correct in that the result is always an RDT, i.e., there are only finitely many tuples for which (42) is non-zero. Analogously as in the case of $\div$, we get

$$\big(\mathcal{D}_1 \rightarrow^{\mathcal{D}_3} \mathcal{D}_2\big)(r) = \begin{cases} \mathcal{D}_3(r), & \text{if } \mathcal{D}_1(r) \leq \mathcal{D}_2(r) \text{ or } \mathcal{D}_3(r) \leq \mathcal{D}_2(r), \\ \mathcal{D}_2(r), & \text{otherwise,} \end{cases} \tag{43}$$

which follows easily by (14). Since $\rightarrow$ acts in a similar way as the truth function of the classic implication, (43) may be seen as expressing the score of a condition "r belongs to $\mathcal{D}_3$ and if it belongs to $\mathcal{D}_1$, then it belongs to $\mathcal{D}_2$".

Remark 5. (a) The operations of join, projection, union, difference, and division behave the same way as their ordinary counterparts when the scores in the input RDTs are only 0 and 1 (i.e., their range is a subset of $\{0, 1\}$). In addition, the restriction also behaves as the ordinary restriction provided that the input RDT has only scores 0 and 1 and that the range of the restriction condition is also a subset of $\{0, 1\}$. In general, restrictions produce RDTs with general scores: The tables in Fig. 1 may be seen as such examples.

(b) Form the point of view of the representation of RDTs as ordinary relations with a special attribute #, see Remark 3 (a), we may think of the operations introduced in this section as derived operations which always produce a relation with # (representing the output RDT) from other relations with # (representing the input RDTs). From the perspective of *TTM* and in particular the relational query language *Tutorial D*, the operations may be implemented as user defined operators in a similar fashion as the operators supporting operations with temporal data described in [13].

4 Ordinal Equivalence of Tables

We introduce notions of ordinal inclusion and equivalence of ranked data tables based on positions of tuples in tables given by scores. In the next section, we utilize the notion in a characterization of important order-related properties of the relational operations with RDTs. Intuitively, we may consider $\mathcal{D}_1$ and $\mathcal{D}_2$ (on the same relation scheme) ordinally equivalent if the sequences of tuples in $\mathcal{D}_1$ and $\mathcal{D}_2$ sorted by scores are identical. Formally, we introduce the notion as follows.

Definition 1. For any $\mathcal{D}$ on R and $r \in \mathrm{Tupl}(R)$, we put

$$\mathcal{U}(\mathcal{D}, r) = \{r' \in \mathrm{Tupl}(R);\, \mathcal{D}(r') \geq \mathcal{D}(r)\}. \tag{44}$$

For any $\mathcal{D}_1$ and $\mathcal{D}_2$ on R we say that $\mathcal{D}_1$ is *ordinally included* in $\mathcal{D}_2$, written $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$, whenever

$$\mathcal{U}(\mathcal{D}_1, r) \subseteq \mathcal{U}(\mathcal{D}_2, r) \tag{45}$$

for all $r \in \mathrm{Tupl}(R)$. Moreover, we call $\mathcal{D}_1$ and $\mathcal{D}_2$ *ordinally equivalent*, written $\mathcal{D}_1 \equiv \mathcal{D}_2$, whenever

$$\mathcal{D}_1 \sqsubseteq \mathcal{D}_2 \text{ and } \mathcal{D}_2 \sqsubseteq \mathcal{D}_1. \tag{46}$$

#	FOO
0.600	77

#	FOO
0.500	77

Figure 4: Two distinct RDTs such that $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$ and $\mathcal{D}_2 \sqsubseteq \mathcal{D}_1$.

Remark 6. We can immediately observe properties of $\mathcal{U}$, $\sqsubseteq$, and $\equiv$ which follow directly by the definition: First, $r \in \mathcal{U}(\mathcal{D}, r)$ follows by the reflexivity of $\leq$. Second, if $\mathcal{D}(r) = 0$, then $\mathcal{U}(\mathcal{D}, r) = \mathrm{Tupl}(R)$ and it is infinite if R contains an attribute of an infinite type. If $\mathcal{D}(r) > 0$, then $\mathcal{U}(\mathcal{D}, r)$ is always finite and it is a subset of the answer set of $\mathcal{D}$ which follows directly by (44). Third, $\sqsubseteq$ is reflexive (a consequence of the reflexivity of $\subseteq$) and transitive (a consequence of the transitivity of $\subseteq$) and thus $\sqsubseteq$ is a preorder (also called a quasi order [5]). In general, $\sqsubseteq$ is not a partial order because it is not antisymmetric. Indeed, consider the RDTs $\mathcal{D}_1$ and $\mathcal{D}_2$ on $R = \{\texttt{FOO}\}$ in Fig. 4. For the only tuple r which appears in the answer set of both the RDTs, we have $\mathcal{U}(\mathcal{D}_1, r) = \{r\} = \mathcal{U}(\mathcal{D}_2, r)$ which from it readily follows that $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$ and $\mathcal{D}_2 \sqsubseteq \mathcal{D}_1$. Fourth, in general, $\sqsubseteq$ has no relationship to the inclusion of answer sets. For instance, if $\mathcal{D}_2$ has an empty answer set (i.e., $\mathcal{D}_2(r) = 0$ for all $r \in \mathrm{Tupl}(R)$), then trivially $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$ for any $\mathcal{D}_1$ on R. Fifth, by definition, $\equiv$ is the symmetric interior of $\sqsubseteq$ (i.e., the greatest symmetric relation contained in both $\sqsubseteq$ and its inverse) and therefore it is an equivalence relation.

Dually to $\mathcal{U}(\mathcal{D}, r)$, we may introduce $\mathcal{L}(\mathcal{D}, r)$ by

$$\mathcal{L}(\mathcal{D}, r) = \{r' \in \mathrm{Tupl}(R);\, \mathcal{D}(r') \leq \mathcal{D}(r)\} \tag{47}$$

for any $r \in \mathrm{Tupl}(R)$. Therefore, in contrast to $\mathcal{U}(\mathcal{D}, r)$ which represents the set of tuples in $\mathcal{D}$ which have scores at least as high as $\mathcal{D}(r)$, $\mathcal{L}(\mathcal{D}, r)$ is the set of tuples with scores at most as high as $\mathcal{D}(r)$. It is easy to see that $\sqsubseteq$ and $\equiv$ can equivalently be defined using (47) instead of (44) which is justified by the following assertion.

Theorem 2. *$\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$ iff for all $r \in \mathrm{Tupl}(R)$, we have $\mathcal{L}(\mathcal{D}_1, r) \subseteq \mathcal{L}(\mathcal{D}_2, r)$.*

Proof. Let $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$ and consider any $r \in \mathrm{Tupl}(R)$. Furthermore, let $r' \in \mathcal{L}(\mathcal{D}_1, r)$, i.e., $\mathcal{D}(r') \leq \mathcal{D}(r)$ by (47). Using (44), the last inequality gives $r \in \mathcal{U}(\mathcal{D}_1, r')$ and thus $r \in \mathcal{U}(\mathcal{D}_2, r')$ because $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$. Now, from $r \in \mathcal{U}(\mathcal{D}_2, r')$ it follows that $r' \in \mathcal{L}(\mathcal{D}_2, r)$. As a consequence, $\mathcal{L}(\mathcal{D}_1, r) \subseteq \mathcal{L}(\mathcal{D}_2, r)$. The converse implication can be shown by analogous arguments utilizing the fact that for any $\mathcal{D}$ on R and arbitrary tuples $r, r' \in \mathrm{Tupl}(R)$, we have $r' \in \mathcal{U}(\mathcal{D}, r)$ iff $r \in \mathcal{L}(\mathcal{D}, r')$. $\square$

Example 2. Recall the RDTs $\mathcal{D}_1$ and $\mathcal{D}_2$ given by the tables in Fig. 2 and Fig. 3, respectively. As one can check, we have $\mathcal{D}_2 \sqsubseteq \mathcal{D}_1$, i.e., $\mathcal{D}_2$ is ordinally included in $\mathcal{D}_1$. On the other hand, $\mathcal{D}_1 \not\sqsubseteq \mathcal{D}_2$ because for $r \in \mathrm{Tupl}(R)$ such that $r(\texttt{PRICE}) = \texttt{\$798,000}$, we have

$$\mathcal{U}(\mathcal{D}_1, r) = \{r, r'\} \nsubseteq \{r\} = \mathcal{U}(\mathcal{D}_2, r),$$

where $r'(\texttt{PRICE}) = \texttt{\$849,000}$. As a consequence, $\mathcal{D}_1$ and $\mathcal{D}_2$ are not ordinally equivalent.

The relations of ordinal inclusion and equivalence of ranked tables are closely related to order-preserving maps and isomorphisms on the structure of scores. The following definition recalls standard notions of maps between ordered sets which we use to get insight into the notions of ordinal inclusion and equivalence.

Definition 3. Let $f\colon L_1 \to L_2$ be a map such that $L_1, L_2 \subseteq L$. Then,
f is called *order preserving* whenever, for all $a, b \in L_1$,

$$a \leq b \text{ implies } f(a) \leq f(b); \tag{48}$$

f is called *order reflecting* whenever, for all $a, b \in L_1$,

$$f(a) \leq f(b) \text{ implies } a \leq b; \tag{49}$$

f is called *order embedding* whenever it is both order preserving and order reflecting;
f is called *order isomorphism* whenever it is a surjective order embedding.

For $\mathcal{D}$ on R and $f : L_1 \to L_2$ such that $L_1, L_2 \subseteq L$, we may consider the usual *composition* $\mathcal{D} \circ f$ (written in the diagrammatic notation) defined by

$$(\mathcal{D} \circ f)(r) = f(\mathcal{D}(r)) \tag{50}$$

for all $r \in \mathrm{Tupl}(R)$; $\mathcal{D} \circ f$ is a correctly defined map since both $\mathcal{D}$ and f are considered as maps, see (25). Observe that if $f(0) > 0$, then $\mathcal{D} \circ f$ may not be a ranked table since infinitely many tuples in $\mathcal{D} \circ f$ may get a non-zero score when R contains an attribute of an infinite type (we tacitly ignore the fact in the rest of the paper because it is not relevant to our investigation). On the other hand, if $f(0) = 0$, then there are only finitely many $r \in \mathrm{Tupl}(R)$ such that $(\mathcal{D} \circ f)(r) > 0$, i.e., $\mathcal{D} \circ f$ is always an RDT. In fact, in this case the answer set of $\mathcal{D} \circ f$ is a subset of the answer set of $\mathcal{D}$. In addition, if f is order reflecting then it is easily seen that the answer sets of $\mathcal{D} \circ f$ and $\mathcal{D}$ coincide: $f(\mathcal{D}(r)) = 0$ yields $f(\mathcal{D}(r)) \leq f(0)$, i.e., $\mathcal{D}(r) \leq 0$ by (49), meaning that if r is in the answer set of $\mathcal{D}$, then it is in the answer set of $\mathcal{D} \circ f$.

The basic relationship of ordinal inclusion and equivalence relations and particular order-preserving maps is described by the following two assertions.

Theorem 4. *Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be RDTs on R. Then, $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$ iff there is an order-preserving map $f\colon L \to L$ such that $\mathcal{D}_1 \circ f = \mathcal{D}_2$.*

Proof. In order to prove the only-if part of the assertion, assume that $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$ and consider $f\colon L \to L$ defined by

$$f(a) = \inf\{\mathcal{D}_2(r);\, r \in \mathrm{Tupl}(R) \text{ and } \mathcal{D}_1(r) \geq a\} \tag{51}$$

for all $a \in L$. Observe that f depends on both $\mathcal{D}_1$ and $\mathcal{D}_2$ and it is order preserving. Indeed, if $a \leq b$, then $\mathcal{D}_1(r) \geq b$ implies $\mathcal{D}_1(r) \geq a$, and so

$$\{\mathcal{D}_2(r);\, r \in \mathrm{Tupl}(R) \text{ and } \mathcal{D}_1(r) \geq a\} \supseteq \{\mathcal{D}_2(r);\, r \in \mathrm{Tupl}(R) \text{ and } \mathcal{D}_1(r) \geq b\}$$

from which it follows that

$$\inf\{\mathcal{D}_2(r);\, r \in \mathrm{Tupl}(R) \text{ and } \mathcal{D}_1(r) \geq a\} \leq \inf\{\mathcal{D}_2(r);\, r \in \mathrm{Tupl}(R) \text{ and } \mathcal{D}_1(r) \geq b\},$$

i.e., $f(a) \leq f(b)$. Moreover, using (50) and (51), we have

$$(\mathcal{D}_1 \circ f)(r) = f(\mathcal{D}_1(r)) = \inf\{\mathcal{D}_2(r');\, r' \in \mathrm{Tupl}(R) \text{ and } \mathcal{D}_1(r') \geq \mathcal{D}_1(r)\},$$

i.e., in order to prove $\mathcal{D}_1 \circ f = \mathcal{D}_2$, it suffices to show that $\mathcal{D}_2(r)$ is the least element of

$$K = \{\mathcal{D}_2(r');\, r' \in \mathrm{Tupl}(R) \text{ and } \mathcal{D}_1(r') \geq \mathcal{D}_1(r)\}.$$

Clearly, $\mathcal{D}_2(r) \in K$ owing to the reflexivity of $\geq$ in the special case of $r = r'$. Now, consider a general $\mathcal{D}_2(r') \in K$, i.e., $r' \in \mathrm{Tupl}(R)$ such that $\mathcal{D}_1(r') \geq \mathcal{D}_1(r)$. Using (44), it means $r' \in \mathcal{U}(\mathcal{D}_1, r)$ and so $r' \in \mathcal{U}(\mathcal{D}_2, r)$ using the assumption $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$. As a consequence, $\mathcal{D}_2(r') \geq \mathcal{D}_2(r)$, proving that $\mathcal{D}_2(r)$ is the least element of K which further gives $\mathcal{D}_1 \circ f = \mathcal{D}_2$.

The if-part is easy to see: Let $f : L \to L$ be a map satisfying (48) and $\mathcal{D}_1 \circ f = \mathcal{D}_2$. Take $r' \in \mathcal{U}(\mathcal{D}_1, r)$. Then, $\mathcal{D}_1(r') \geq \mathcal{D}_1(r)$ and so $f(\mathcal{D}_1(r')) \geq f(\mathcal{D}_1(r))$ because f is order preserving. Using $\mathcal{D}_1 \circ f = \mathcal{D}_2$, we obtain

$$\mathcal{D}_2(r') = f(\mathcal{D}_1(r')) \geq f(\mathcal{D}_1(r)) = \mathcal{D}_2(r),$$

meaning $r' \in \mathcal{U}(\mathcal{D}_2, r)$. Hence, $\mathcal{U}(\mathcal{D}_1, r) \subseteq \mathcal{U}(\mathcal{D}_2, r)$ which proves $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$. $\square$

Example 3. Consider RDTs $\mathcal{D}_1$ and $\mathcal{D}_2$ as in Example 2. Since $\mathcal{D}_2 \sqsubseteq \mathcal{D}_1$, Theorem 4 yields there is a map $f\colon L \to L$ such that $\mathcal{D}_1 = \mathcal{D}_2 \circ f$. A map f satisfying this property is not given uniquely. The map given by (51) described in the proof of Theorem 4 is given by

$$f(a) = \begin{cases} 0, & \text{if } a = 0, \\ 0.148, & \text{if } 0 < a \leq 0.148, \\ 0.426, & \text{if } 0.148 < a \leq 0.426, \\ 0.643, & \text{if } 0.426 < a \leq 0.643, \\ 0.778, & \text{if } 0.643 < a \leq 0.778, \\ 0.937, & \text{if } 0.778 < a \leq 0.939, \\ 1, & \text{if } a > 0.939, \end{cases}$$

for all $a \in L$.

For the next theorem, recall that $L(\mathcal{D})$, called the range of $\mathcal{D}$, represents the set of scores which appear in $\mathcal{D}$ and in general it includes 0, see (26).

Theorem 5. *Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be RDTs on R. Then, $\mathcal{D}_1 \equiv \mathcal{D}_2$ iff there is an order isomorphism $f\colon L(\mathcal{D}_1) \to L(\mathcal{D}_2)$ such that $\mathcal{D}_1 \circ f = \mathcal{D}_2$.*

Proof. Let $\mathcal{D}_1 \equiv \mathcal{D}_2$, i.e., $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$ and $\mathcal{D}_2 \sqsubseteq \mathcal{D}_1$. By Theorem 4, there are order-preserving maps $f : L \to L$ and $g\colon L \to L$ such that $\mathcal{D}_1 \circ f = \mathcal{D}_2$ and $\mathcal{D}_1 = \mathcal{D}_2 \circ g$. Furthermore, consider the restrictions $f|_{L(\mathcal{D}_1)}$ and $g|_{L(\mathcal{D}_2)}$ of f and g to $L(\mathcal{D}_1)$ and $L(\mathcal{D}_2)$, respectively. Under this notation, $f|_{L(\mathcal{D}_1)}$ and $g|_{L(\mathcal{D}_2)}$ are order preserving maps of the form $f|_{L(\mathcal{D}_1)}\colon L(\mathcal{D}_1) \to L(\mathcal{D}_2)$ and $g|_{L(\mathcal{D}_2)}\colon L(\mathcal{D}_2) \to L(\mathcal{D}_1)$ which satisfy

$$\mathcal{D}_1 \circ f|_{L(\mathcal{D}_1)} = \mathcal{D}_2 \text{ and } \mathcal{D}_1 = \mathcal{D}_2 \circ g|_{L(\mathcal{D}_2)}.$$

Therefore, we have

$$\begin{aligned}\mathcal{D}_1(r) &= \big(\mathcal{D}_2 \circ g|_{L(\mathcal{D}_2)}\big)(r)\\ &= g|_{L(\mathcal{D}_2)}(\mathcal{D}_2(r))\\ &= g|_{L(\mathcal{D}_2)}\big(\big(\mathcal{D}_1 \circ f|_{L(\mathcal{D}_1)}\big)(r)\big)\\ &= g|_{L(\mathcal{D}_2)}\big(f|_{L(\mathcal{D}_1)}(\mathcal{D}_1(r))\big)\end{aligned}$$

for all $r \in \mathrm{Tupl}(R)$ which is in the answer set of $\mathcal{D}_1$. As a consequence, the composed map $f|_{L(\mathcal{D}_1)} \circ g|_{L(\mathcal{D}_2)}$ is the identity map on $L(\mathcal{D}_1)$. Using analogous arguments, $g|_{L(\mathcal{D}_2)} \circ f|_{L(\mathcal{D}_1)}$ is the identity map on $L(\mathcal{D}_2)$. This shows that $f|_{L(\mathcal{D}_1)}$ is an order embedding: $f|_{L(\mathcal{D}_1)}(a) \leq f|_{L(\mathcal{D}_1)}(b)$ implies $g|_{L(\mathcal{D}_2)}(f|_{L(\mathcal{D}_1)}(a)) \leq g|_{L(\mathcal{D}_2)}(f|_{L(\mathcal{D}_1)}(b))$ because $g|_{L(\mathcal{D}_2)}$ is order preserving and as a consequence of the fact that $f|_{L(\mathcal{D}_1)} \circ g|_{L(\mathcal{D}_2)}$ is the identity, we get that $a \leq b$. In addition, $f|_{L(\mathcal{D}_1)}\colon L(\mathcal{D}_1) \to L(\mathcal{D}_2)$ is surjective because for each $a \in L(\mathcal{D}_2)$, we have that $f|_{L(\mathcal{D}_1)}(g|_{L(\mathcal{D}_2)}(a)) = a$. Altogether, $f|_{L(\mathcal{D}_1)}$ is the desired order isomorphism.

In order to prove the if-part of Theorem 5, let us consider an order isomorphism $f\colon L(\mathcal{D}_1) \to L(\mathcal{D}_2)$ such that $\mathcal{D}_1 \circ f = \mathcal{D}_2$. Now, f can be extended to a map $f^\sharp\colon L \to L$ by putting

$$f^\sharp(a) = \inf\{f(a');\, a' \in L(\mathcal{D}_1) \text{ and } a' \geq a\}$$

for all $a \in L$. Observe that $f^\sharp$ is indeed an extension of f: For $a \in L(\mathcal{D}_1)$, it follows that $f(a)$ belongs to

$$K = \{f(a');\, a' \in L(\mathcal{D}_1) \text{ and } a' \geq a\}$$

because of the reflexivity of $\geq$. Moreover, if $f(a') \in K$, then $a' \geq a$ and so $f(a') \geq f(a)$ owing to the fact that f is order preserving. Thus, $f(a)$ is the least element of K and, as a consequence, $f^\sharp(a) = \inf\{f(a');\, a' \in L(\mathcal{D}_1) \text{ and } a' \geq a\} = f(a)$. Furthermore, the fact that f is order preserving ensures that $f^\sharp$ is order preserving as well. Indeed, take any $a, b \in L$ such that $a \leq b$. Then, analogously as in the proof of Theorem 4, we have

$$\{f(a');\, a' \in L(\mathcal{D}_1) \text{ and } a' \geq a\} \supseteq \{f(a');\, a' \in L(\mathcal{D}_1) \text{ and } a' \geq b\}$$

and thus

$$\inf\{f(a');\, a' \in L(\mathcal{D}_1) \text{ and } a' \geq a\} \leq \inf\{f(a');\, a' \in L(\mathcal{D}_1) \text{ and } a' \geq b\}$$

which proves $f^\sharp(a) \leq f^\sharp(b)$. Therefore, $\mathcal{D}_1 \sqsubseteq \mathcal{D}_2$ owing to Theorem 4. In addition, $\mathcal{D}_2 \sqsubseteq \mathcal{D}_1$ follows using the same arguments using the inverse f^{-1} of f. Note that f being an order isomorphism ensures that f is a bijection, so the inverse of f exists. $\square$

5 Invariance Theorems

In this section, we present two invariance theorems which are the main observations of this paper. As a result of the invariance theorems, it follows that results of arbitrary complex queries composed of (27)–(40) are invariant to ordinal transformations: If the input data are transformed by f into ordinally equivalent data, then the results of queries performed with the original and the new data are also ordinally equivalent. As a practical consequence, if a transformation of the input data does not change the order in which tuples appear in tables when sorted by scores, then the same property holds for results of arbitrary queries.

Theorem 6. *Let $f : L \to L$ be order preserving. Then, for any RDTs $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}$ for which both sides of the following equalities are defined, we have*

$$(\mathcal{D}_1 \bowtie \mathcal{D}_2) \circ f = (\mathcal{D}_1 \circ f) \bowtie (\mathcal{D}_2 \circ f), \tag{52}$$

$$\sigma_\theta(\mathcal{D}) \circ f = \sigma_{\theta \circ f}(\mathcal{D} \circ f), \tag{53}$$

$$(\mathcal{D}_1 \cup \mathcal{D}_2) \circ f = (\mathcal{D}_1 \circ f) \cup (\mathcal{D}_2 \circ f), \tag{54}$$

$$\pi_S(\mathcal{D}) \circ f = \pi_S(\mathcal{D} \circ f). \tag{55}$$

Proof. In order to prove (52), we check that

$$f(\inf\{\mathcal{D}_1(rs), \mathcal{D}_2(st)\}) = \inf\{f(\mathcal{D}_1(rs)), f(\mathcal{D}_2(st))\}.$$

Since $\mathbf{L}$ is linear, we may proceed by cases: First, assume that $\mathcal{D}_1(rs) \leq \mathcal{D}_2(st)$. Then, $f(\mathcal{D}_1(rs)) \leq f(\mathcal{D}_2(st))$ because f is order preserving and thus

$$\begin{aligned} f(\inf\{\mathcal{D}_1(rs), \mathcal{D}_2(st)\}) &= f(\mathcal{D}_1(rs)) \\ &= \inf\{f(\mathcal{D}_1(rs))\} \\ &= \inf\{f(\mathcal{D}_1(rs)), f(\mathcal{D}_2(st))\}. \end{aligned}$$

Second, assume $\mathcal{D}_1(rs) \geq \mathcal{D}_2(st)$ and proceed as above with $\leq$ replaced by $\geq$.

Analogously, we may proceed for (53). It suffices to check that

$$f(\inf\{\mathcal{D}(r), \theta(r)\}) = \inf\{f(\mathcal{D}(r)), \theta(r)\}$$

during which we distinguish two cases: (i) $\mathcal{D}(r) \leq \theta(r)$ and thus $f(\mathcal{D}(r)) \leq f(\theta(r))$; (ii) $\mathcal{D}(r) \geq \theta(r)$ and $f(\mathcal{D}(r)) \geq f(\theta(r))$.

Now, (54) follows by the same argument as in the case of (52) with sup in place of inf. Indeed, we check that

$$f(\sup\{\mathcal{D}_1(r), \mathcal{D}_2(r)\}) = \sup\{f(\mathcal{D}_1(r)), f(\mathcal{D}_2(r))\}$$

holds by cases in which we use the fact that $\mathcal{D}_1(r) \leq \mathcal{D}_2(r)$ iff $\sup\{\mathcal{D}_1(r), \mathcal{D}_2(r)\} = \mathcal{D}_2(r)$ together with the assumption that f is order preserving, and dually for $\geq$.

In case of (55), it suffices to check that f commutes with suprema of finite subsets of L which is indeed the case. In a more detail, let $\mathcal{D}$ be an RDT on R and $S \subseteq R$. In this setting, it suffices to prove that

$$f(\sup\{\mathcal{D}(st);\, t \in \mathrm{Tupl}(R \setminus S)\}) = \sup\{f(\mathcal{D}(st));\, t \in \mathrm{Tupl}(R \setminus S)\}$$

for any $s \in \mathrm{Tupl}(S)$. Observe that for any $s \in \mathrm{Tupl}(S)$,

$$K = \{\mathcal{D}(st);\, t \in \mathrm{Tupl}(R \setminus S)\}$$

is a finite set of scores which is a subset of the (finite) range of $\mathcal{D}$. In addition, K is non-empty because $\mathrm{Tupl}(R \setminus S)$ is always non-empty and s and t are trivially joinable. Therefore, owing to the fact that $\mathbf{L}$ is totally ordered, there is $t' \in \mathrm{Tupl}(R \setminus S)$ such that $\mathcal{D}(st')$ is the greatest element of K. Since f is order preserving, it readily follows that $f(\mathcal{D}(st'))$ is the greatest element of

$$f(K) = \{f(\mathcal{D}(st));\, t \in \mathrm{Tupl}(R \setminus S)\}.$$

Therefore, under this notation, we have

$$\begin{aligned}
f(\sup\{\mathcal{D}(st);\, t \in \mathrm{Tupl}(R \setminus S)\}) &= f(\sup K) \\
&= f(\mathcal{D}(st')) \\
&= \sup f(K) \\
&= \sup\{f(\mathcal{D}(st));\, t \in \mathrm{Tupl}(R \setminus S)\},
\end{aligned}$$

which proves (55). □

Under stronger assumptions than in Theorem 6, we establish the following observation of invariance for the remaining operations with RDTs.

Theorem 7. *Let $f\colon L \to L$ be order embedding and let $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$ be RDTs for which both sides of the following equalities are defined. Then,*

$$(\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2) \circ f = (\mathcal{D}_1 \circ f) \div^{\mathcal{D}_3 \circ f} (\mathcal{D}_2 \circ f), \tag{56}$$

$$(\mathcal{D}_1 \to^{\mathcal{D}_3} \mathcal{D}_2) \circ f = (\mathcal{D}_1 \circ f) \to^{\mathcal{D}_3 \circ f} (\mathcal{D}_2 \circ f). \tag{57}$$

If $f(0) = 0$, then

$$(\mathcal{D}_1 - \mathcal{D}_2) \circ f = (\mathcal{D}_1 \circ f) - (\mathcal{D}_2 \circ f). \tag{58}$$

If $f(1) = 1$, then

$$\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2) \circ f = \mathrm{S}(\mathcal{D}_1 \circ f, \mathcal{D}_2 \circ f). \tag{59}$$

Proof. In case of (58), we distinguish two cases based on (33). First, if we have $\mathcal{D}_1(r) \leq \mathcal{D}_2(r)$, then $f(\mathcal{D}_1(r)) \leq f(\mathcal{D}_2(r))$ because f is order preserving and so

$$((\mathcal{D}_1 - \mathcal{D}_2) \circ f)(r) = f(0) = 0 = ((\mathcal{D}_1 \circ f) - (\mathcal{D}_2 \circ f))(r),$$

taking into account the fact that $f(0) = 0$. Second, assume that $\mathcal{D}_1(r) \nleq \mathcal{D}_2(r)$. In this case, $\mathcal{D}_1(r) > \mathcal{D}_2(r)$ because $\mathbf{L}$ is totally ordered and so $f(\mathcal{D}_1(r)) \geq f(\mathcal{D}_2(r))$ because f is order preserving. Since f is also order reflecting, we must have $f(\mathcal{D}_1(r)) > f(\mathcal{D}_2(r))$ because $f(\mathcal{D}_1(r)) = f(\mathcal{D}_2(r))$ would yield $\mathcal{D}_1(r) \leq \mathcal{D}_2(r)$, a contradiction. Therefore, we have $f(\mathcal{D}_1(r)) \nleq f(\mathcal{D}_2(r))$ and so

$$\begin{aligned}
((\mathcal{D}_1 - \mathcal{D}_2) \circ f)(r) &= f((\mathcal{D}_1 - \mathcal{D}_2)(r)) \\
&= f(\mathcal{D}_1(r)) \\
&= (f(\mathcal{D}_1) - f(\mathcal{D}_2))(r),
\end{aligned}$$

which proves (58).

In case of (56), we may proceed by cases considering the condition (36). In a more detail, let $\mathcal{D}_1$ be an RDT on $R \cup S$ such that $R \cap S = \emptyset$, $\mathcal{D}_2$ be an RDT on S, and $\mathcal{D}_3$ be an RDT on R. Furthermore, assume that for a given $r \in \mathrm{Tupl}(R)$ and all $s \in \mathrm{Tupl}(S)$, we have that $\mathcal{D}_2(s) > \mathcal{D}_1(rs)$ implies $\mathcal{D}_3(r) \leq \mathcal{D}_1(rs)$. In this case, $\big(\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2\big)(r) = \mathcal{D}_3(r)$. Moreover, the fact that f is an order embedding gives that $f(\mathcal{D}_2(s)) > f(\mathcal{D}_1(rs))$ implies $\mathcal{D}_2(s) > \mathcal{D}_1(rs)$ and so $\mathcal{D}_3(r) \leq \mathcal{D}_1(rs)$, i.e., $f(\mathcal{D}_3(r)) \leq f(\mathcal{D}_1(rs))$. As a consequence,

$$\begin{aligned}
f\big(\big(\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2\big)(r)\big) &= f(\mathcal{D}_3(r)) \\
&= \big((\mathcal{D}_1 \circ f) \div^{\mathcal{D}_3 \circ f} (\mathcal{D}_2 \circ f)\big).
\end{aligned}$$

It remains to prove the equality in the case when (36) does not hold. That is, assume that for given $r \in \mathrm{Tupl}(R)$ there is $s \in \mathrm{Tupl}(S)$ such that $\mathcal{D}_2(s) > \mathcal{D}_1(rs)$ and $\mathcal{D}_3(r) > \mathcal{D}_1(rs)$. Therefore, for given $r \in \mathrm{Tupl}(R)$,

$$K = \{\mathcal{D}_1(rs);\, \mathcal{D}_2(s) > \mathcal{D}_1(rs),\, s \in \mathrm{Tupl}(S)\}$$

is non-empty and in addition it is finite because it is a subset of the range of $\mathcal{D}_1$. Since $\mathbf{L}$ is totally ordered, there is $s' \in \mathrm{Tupl}(S)$ such that $\mathcal{D}_1(rs')$ is the least element of K. The fact that f is an order embedding further gives that $f(\mathcal{D}_1(rs'))$ is the least element of

$$f(K) = \{f(\mathcal{D}_1(rs));\, f(\mathcal{D}_2(s)) > f(\mathcal{D}_1(rs)),\, s \in \mathrm{Tupl}(S)\}.$$

Hence,

$$\begin{aligned}
f\big(\big(\mathcal{D}_1 \div^{\mathcal{D}_3} \mathcal{D}_2\big)(r)\big) &= f(\mathcal{D}_1(rs')) \\
&= \big((\mathcal{D}_1 \circ f) \div^{\mathcal{D}_3 \circ f} (\mathcal{D}_2 \circ f)\big),
\end{aligned}$$

#	ID	PRICE
0.882	71	798,000
0.882	71	849,000
0.655	85	998,000
0.541	82	648,000
0.462	58	829,000
0.272	93	598,000

#	ID	PRICE
0.877	71	798,000
0.782	71	849,000
0.655	85	998,000
0.429	58	829,000
0.361	82	648,000
0.160	93	598,000

Figure 5: Join of transformed ranked data table projected onto $\{\mathtt{ID}, \mathtt{PRICE}\}$ (left) and result of the same query using the Goguen aggregation (right).

which concludes the proof of (56). Now, observe that (59) follows directly by (56). Indeed, for $\mathcal{D}_1$ and $\mathcal{D}_2$ on R and for an auxiliary $\mathcal{D}$ on $\emptyset$ such that $\mathcal{D}(\emptyset) = 1$, we have

$$\begin{aligned}
\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2) \circ f &= f\big(\big(\mathcal{D}_2 \div^{\mathcal{D}} \mathcal{D}_1\big)(\emptyset)\big) \\
&= \big((\mathcal{D}_2 \circ f) \div^{\mathcal{D} \circ f} (\mathcal{D}_1 \circ f)\big)(\emptyset) \\
&= \big((\mathcal{D}_2 \circ f) \div^{\mathcal{D}} (\mathcal{D}_1 \circ f)\big)(\emptyset) \\
&= \mathrm{S}(\mathcal{D}_1 \circ f, \mathcal{D}_2 \circ f)
\end{aligned}$$

provided that $f(1) = 1$ and thus $\mathcal{D} \circ f = \mathcal{D}$. Finally, (57) can be proved analogously as (56) by inspecting the cases in (43), the details are left to the reader. □

If $f : L \to L$ is an order isomorphism, then all conditions in Theorem 6 and Theorem 7 are satisfied including the facts that $f(0) = 0$ and $f(1) = 1$. Such f may be viewed as an ordinal transformation function of ranked data tables. We may say that $\mathcal{D}_1$ is ordinally transformed into $\mathcal{D}_2$ by f, written $\mathcal{D}_1 \mapsto_f \mathcal{D}_2$, whenever $\mathcal{D}_1 \circ f = \mathcal{D}_2$. Under this notation, (52)–(59) in the invariance theorems can be restated as follows: If $\mathcal{D}_1 \mapsto_f \mathcal{D}'_1$ and $\mathcal{D}_2 \mapsto_f \mathcal{D}'_2$, then

$$\mathcal{D}_1 \bowtie \mathcal{D}_2 \equiv \mathcal{D}'_1 \bowtie \mathcal{D}'_2 \tag{60}$$

in case of $\bowtie$ and analogously for σ_θ, π_S, $\cup$, $-$, $\div$, and S. Put in words, the results of an operation with transformed input data and the original input data are equivalent in terms of the order of tuples given by scores.

Example 4. To illustrate the invariance theorems on concrete data, consider the RDTs $\mathcal{D}_1$ and $\mathcal{D}_2$ as in Fig. 1. A map $f : [0,1] \to [0,1]$ given by

$$f(x) = \begin{cases} 2^{-0.5}\sqrt{x}, & \text{if } x \leq 0.5, \\ 2(x - 0.5)^2 + 0.5, & \text{otherwise,} \end{cases} \tag{61}$$

is an order isomorphism preserving 0 and 1. Fig. 5 (left) contains the result of

$$\pi_{\{\mathtt{ID},\mathtt{PRICE}\}}((\mathcal{D}_1 \circ f) \bowtie (\mathcal{D}_2 \circ f))$$

#	ID	BDRM	PRICE
0.778	85	5	998,000
0.699	71	3	798,000
0.699	71	3	849,000
0.643	82	4	648,000
0.426	58	4	829,000
0.148	93	2	598,000

#	ID	BDRM	PRICE
0.655	85	5	998,000
0.579	71	3	798,000
0.579	71	3	849,000
0.541	82	4	648,000
0.462	58	4	829,000
0.272	93	2	598,000

Figure 6: Result of $\sigma_\theta(\mathcal{D}_1 \bowtie \mathcal{D}_2)$ projected onto $S = \{\mathtt{ID}, \mathtt{BDRM}, \mathtt{PRICE}\}$ (left) and $\sigma_{\theta \circ f}((\mathcal{D}_1 \circ f) \bowtie (\mathcal{D}_2 \circ f))$ projected onto S (right).

#	ID	BDRM	PRICE
0.699	71	3	798,000
0.699	71	3	849,000
0.655	85	5	998,000
0.541	82	4	648,000
0.462	58	4	829,000
0.272	93	2	598,000

Figure 7: Result of $\pi_S(\sigma_\theta((\mathcal{D}_1 \circ f) \bowtie (\mathcal{D}_2 \circ f)))$.

which is equivalent to

$$\pi_{\{\mathtt{ID},\mathtt{PRICE}\}}(\mathcal{D}_1 \bowtie \mathcal{D}_2) \circ f$$

owing to (52) and (53). The tuples in the result, when sorted by scores, appear in the same order as in Fig. 2 showing $\mathcal{D}_1 \bowtie \mathcal{D}_2$. Our assumption that the join (27) (as well as the other operations) is defined using the infimum instead of a general aggregation function $\otimes$, see Remark 4, is essential. If we replace inf in (27) by $\otimes$ being the multiplication of reals (so-called Goguen aggregation, see [20]) and compute $\pi_{\{\mathtt{ID},\mathtt{PRICE}\}}((\mathcal{D}_1 \circ f) \bowtie (\mathcal{D}_2 \circ f))$, we get Fig. 5 (right) as the result where the order of tuples is not preserved.

As a further example, Fig. 6 (left) shows the result of a restriction of the join using the restriction condition θ defined by

$$\theta(r) = \begin{cases} 0.1(4 + r(\mathtt{BDRM})), & \text{if } r(\mathtt{DBRM}) \leq 6, \\ 1, & \text{otherwise}, \end{cases} \tag{62}$$

which may be seen as a restriction on the number of bedrooms 6 and more with a tolerance for lower numbers. Fig. 6 (right) shows the result for the tables transformed by f as above. Again, the tuples appear in the same order. Finally, Fig. 7 shows that without transforming θ, the order of tuples in the result would not be preserved, i.e., $\theta \circ f$ in (53) cannot be replaced by θ.

The invariance theorems can be seen as type of description of the independence of query results on possible changes in scores in the input data and restriction conditions in queries. An alternative characterization which does not utilize the position of tuples in relations but uses a notion of similarity was proposed in [3]. We now make a comment on how the approaches can be combined. As we have outlined in the introduction, [3] introduces lower bounds for similarity of query results based on similarity of input data. For instance, in the case of joins of RDTs, [3] shows that

$$\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2) \otimes \mathrm{S}(\mathcal{D}_3, \mathcal{D}_4) \leq \mathrm{S}(\mathcal{D}_1 \bowtie \mathcal{D}_3, \mathcal{D}_2 \bowtie \mathcal{D}_4), \tag{63}$$

$$\mathrm{E}(\mathcal{D}_1, \mathcal{D}_2) \otimes \mathrm{E}(\mathcal{D}_3, \mathcal{D}_4) \leq \mathrm{E}(\mathcal{D}_1 \bowtie \mathcal{D}_3, \mathcal{D}_2 \bowtie \mathcal{D}_4), \tag{64}$$

where S is defined as in (39) and E is defined as in (41), and $\otimes$ is a binary aggregation function with suitable properties (it is commutative, associative, and 1 is its neutral element). In our setting, (63) and (64) may be restated with $\otimes$ replaced by inf as

$$\inf\{\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2), \mathrm{S}(\mathcal{D}_3, \mathcal{D}_4)\} \leq \mathrm{S}(\mathcal{D}_1 \bowtie \mathcal{D}_3, \mathcal{D}_2 \bowtie \mathcal{D}_4), \tag{65}$$

$$\inf\{\mathrm{E}(\mathcal{D}_1, \mathcal{D}_2), \mathrm{E}(\mathcal{D}_3, \mathcal{D}_4)\} \leq \mathrm{E}(\mathcal{D}_1 \bowtie \mathcal{D}_3, \mathcal{D}_2 \bowtie \mathcal{D}_4). \tag{66}$$

Put in words, (65) says that the score to which $\mathcal{D}_1 \bowtie \mathcal{D}_3$ is contained in $\mathcal{D}_2 \bowtie \mathcal{D}_4$ as at least the score to which $\mathcal{D}_1$ is contained in $\mathcal{D}_2$ and $\mathcal{D}_3$ is contained in $\mathcal{D}_4$. Analogously, we may interpret (66) with "contained" replaced by "similar".

Now, using the fact that $f(\inf\{a, b\}) = \inf\{f(a), f(b)\}$ for all $a, b \in L$ together with the fact that f is order-preserving, we may conclude that

$$\begin{aligned}
\inf\{\mathrm{S}(\mathcal{D}_1 \circ f, \mathcal{D}_2 \circ f), \mathrm{S}(\mathcal{D}_3 \circ f, \mathcal{D}_4 \circ f)\} &= \inf\{\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2) \circ f, \mathrm{S}(\mathcal{D}_3, \mathcal{D}_4) \circ f\} \\
&= f(\inf\{\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2), \mathrm{S}(\mathcal{D}_3, \mathcal{D}_4)\}) \\
&\leq f(\mathrm{S}(\mathcal{D}_1 \bowtie \mathcal{D}_3, \mathcal{D}_2 \bowtie \mathcal{D}_4)) \\
&= \mathrm{S}(\mathcal{D}_1 \bowtie \mathcal{D}_3, \mathcal{D}_2 \bowtie \mathcal{D}_4) \circ f
\end{aligned}$$

and analogously for E. In much the same way, we get the following inequality:

$$\begin{aligned}
f(\inf\{\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2), \mathrm{S}(\mathcal{D}_3, \mathcal{D}_4)\}) &= \inf\{\mathrm{S}(\mathcal{D}_1, \mathcal{D}_2) \circ f, \mathrm{S}(\mathcal{D}_3, \mathcal{D}_4) \circ f\} \\
&= \inf\{\mathrm{S}(\mathcal{D}_1 \circ f, \mathcal{D}_2 \circ f), \mathrm{S}(\mathcal{D}_3 \circ f, \mathcal{D}_4 \circ f)\} \\
&\leq \mathrm{S}((\mathcal{D}_1 \circ f) \bowtie (\mathcal{D}_3 \circ f), (\mathcal{D}_2 \circ f) \bowtie (\mathcal{D}_4 \circ f)).
\end{aligned}$$

The inequality may be seen as an extension of the lower bound given by (65) which incorporates an ordinal transformation. Indeed, it reads: "the score to which the join of the transformed RDTs $\mathcal{D}_1$ and $\mathcal{D}_3$ is contained in the join of the transformed RDTs $\mathcal{D}_2$ and $\mathcal{D}_4$ is at least as high as the transformed score of containment of $\mathcal{D}_1$ in $\mathcal{D}_2$ and $\mathcal{D}_3$ in $\mathcal{D}_4$." Analogous combined similarity bounds of operation with transformed data can be obtained for the other relational operations, cf. [3].

6 Gödel logic and relational calculi

In the previous section, we have discussed the invariance to ordinal transformations for one particular query system—a system based on relational operations which may be composed to form complex queries. The system resembles the traditional relational algebra. In this section, we show that the same type of results on invariance to ordinal transformations can also be established in a query system which is based on evaluating formulas in database instances consisting of ranked data tables and is conceptually similar to the classic relational calculi. We establish the invariance theorems indirectly by showing that the query system based on evaluating formulas is equivalent to the system based on relational operations. By proving the equivalences of the query systems, we get new insights into the original query system. For instance, it turns our that Gödel logic plays an analogous role in the rank-aware approach investigated in this paper as the Boolean logic in the classic relational model of data. This connection allows us to derive conclusions about properties of the relational operations in our model based on provability of particular formulas in Gödel logic—we utilize this observation in Section 7. In the beginning of this section, we recall first-order Gödel logic in a form that is suitable for our development and then we show its relationship to our model.

A language $\mathscr{J}$ of a first-order Gödel logic is given by a set of relation symbols together with information about their arities. The relation symbols may also be called predicate symbols and in the database terminology they may be understood as relation variables whose values are bound to relations in database instances. Furthermore, we consider a denumerable set X of object variables. Analogously as in the case of the classic first-order logic, formulas are defined recursively based on atomic formulas using symbols for logical connectives and quantifiers:

(i) $\overline{0}$ is a formula (a constant of the truth value "false").

(ii) If $\mathbb{r}$ is n-ary relation symbol and $x_1, \ldots, x_n \in X$, then $\mathbb{r}(x_1, \ldots, x_n)$ is a formula.

(iii) If φ and ψ are formulas, then $(\varphi \wedge \psi)$ and $(\varphi \Rightarrow \psi)$ are formulas.

(iv) If φ is a formula and $x \in X$, then $(\forall x)\varphi$ and $(\exists x)\varphi$ are formulas.

All formulas we consider result by applications of (i)–(iv). Let us note that both (i) and (ii) introduce atomic formulas. In the first case, $\overline{0}$ may be seen as a nullary logical connective (i.e., a connective with no arguments). In the second case, each $\mathbb{r}(x_1, \ldots, x_n)$ is an atomic formula constructed as in the first-order Boolean logic except for the fact that we do not consider more complex terms than object variables—objects constants and general function symbols may also be introduced but this is not necessary for our application of the logic. Also note that a special case of (ii) are formulas of the form $\mathbb{r}()$ when $\mathbb{r}$ is a nullary relation symbol. In such a case, $\mathbb{r}()$ may be denoted just $\mathbb{r}$ and called a propositional symbol. Furthermore, (iii) introduces more complex formulas built using logical connectives $\wedge$ (conjunction) and $\Rightarrow$ (implication); here we adopt the common rules for omission of outer parentheses in formulas. Finally, (iv) defines universally and existentially quantified formulas in the same way as in the classic logic.

Remark 7. We can consider only $\overline{0}$, $\wedge$, and $\Rightarrow$ as the basic connectives. Indeed, formulas containing $\vee$ (disjunction) and possibly other connectives ($\neg$ for a negation, and $\Leftrightarrow$ for a biconditional) can be seen as abbreviations as follows:

$$\neg\varphi \text{ is } \varphi \Rightarrow \overline{0}, \tag{67}$$

$$\varphi \Leftrightarrow \psi \text{ is } (\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi), \tag{68}$$

$$\varphi \vee \psi \text{ is } ((\varphi \Rightarrow \psi) \Rightarrow \psi) \wedge ((\psi \Rightarrow \varphi) \Rightarrow \varphi). \tag{69}$$

Note that in Gödel logic, $\wedge$ is not definable based solely on $\Rightarrow$ and $\overline{0}$ as it is in the classical logic where $\varphi \wedge \psi$ can be seen as an abbreviation for $(\varphi \Rightarrow (\psi \Rightarrow \overline{0})) \Rightarrow \overline{0}$. This is due to the absence of the law of the double negation.

The semantic of formulas is introduced based on their evaluation in general structures for a given language $\mathcal{J}$ based on Gödel algebras. In the database terminology, the language defines a database scheme and the general structures may be seen as counterparts to the classic database instances.

Let $\mathbf{L}$ be a Gödel algebra. An $\mathbf{L}$-structure for language $\mathcal{J}$ is denoted $\mathbf{M}$ and consists of a non-empty universe set M and a set which contains, for each n-ary relation symbol $\mathbb{r}$ in the language, a map $\mathbb{r}^{\mathbf{M}} : M^n \to L$ where M^n denotes the usual n-th power of M. Under this notation, $\mathbb{r}^{\mathbf{M}}(m_1, \ldots, m_n)$ is a degree in L which can be interpreted as a score of a tuple consisting of the values $m_1, \ldots, m_n$ in $\mathbb{r}^{\mathbf{M}}$. Note that in this setting, we do not have names of attributes and therefore the order of arguments in $\mathbb{r}^{\mathbf{M}}(m_1, \ldots, m_n)$ matters (as it is usual in first-order logics, one may easily introduce “names of attributes” to keep the formalism closer to the style of relational database calculi). An $\mathbf{M}$-valuation (of object variables) is any map $v \colon X \to M$, $v(x)$ interpreted as the value of $x \in X$ under v. Now, the values of formulas (of the language $\mathcal{J}$) in $\mathbf{L}$-structure $\mathbf{M}$ (for $\mathcal{J}$) given an $\mathbf{M}$-valuation v is defined by the following rules. In case of the atomic formulas, we put

$$||\overline{0}||_{\mathbf{M},v} = 0, \tag{70}$$

$$||\mathbb{r}(x_1, \ldots, x_n)||_{\mathbf{M},v} = \mathbb{r}^{\mathbf{M}}(v(x_1), \ldots, v(x_n)). \tag{71}$$

For the formulas built using the binary connectives $\wedge$ and $\Rightarrow$, we put

$$||\varphi \wedge \psi||_{\mathbf{M},v} = \inf\{||\varphi||_{\mathbf{M},v}, ||\psi||_{\mathbf{M},v}\}, \tag{72}$$

$$||\varphi \Rightarrow \psi||_{\mathbf{M},v} = ||\varphi||_{\mathbf{M},v} \to ||\psi||_{\mathbf{M},v}, \tag{73}$$

From (69) it follows that

$$||\varphi \vee \psi||_{\mathbf{M},v} = \sup\{||\varphi||_{\mathbf{M},v}, ||\psi||_{\mathbf{M},v}\}. \tag{74}$$

Observe that if $\mathbf{L}$ is totally ordered, then (14) yields

$$||\varphi \Rightarrow \psi||_{\mathbf{M},v} = \begin{cases} 1, & \text{if } ||\varphi||_{\mathbf{M},v} \leq ||\psi||_{\mathbf{M},v}, \\ ||\psi||_{\mathbf{M},v}, & \text{otherwise.} \end{cases} \tag{75}$$

Finally, the value of quantified formulas is defined as follows provided that the right-hand sides of the following equalities are defined:

$$||(\forall x)\varphi||_{\mathbf{M},v} = \inf\{||\varphi||_{\mathbf{M},w};\, w =_x v\}, \tag{76}$$

$$||(\exists x)\varphi||_{\mathbf{M},v} = \sup\{||\varphi||_{\mathbf{M},w};\, w =_x v\}, \tag{77}$$

where $w =_x v$ means that w is an $\mathbf{M}$-valuation such that $w(y) = v(y)$ for all $y \in X$ such that $x \neq y$. Note that in general, (76) and (77) may not be defined because of the non-existence of infima and suprema of $\{||\varphi||_{\mathbf{M},w};\, w =_x v\} \subseteq L$. If for any φ of the language $\mathcal{J}$ (76) and (77) are defined under any $\mathbf{M}$-valuation, then $\mathbf{M}$ is called safe. If $\mathbf{L}$ is complete, then any $\mathbf{L}$-structure is trivially safe. More importantly, if each $r^{\mathbf{M}}$ is finite, meaning there are only finitely many $m_1, \ldots, m_n \in M$ for which $r^{\mathbf{M}}(m_1, \ldots, m_n) > 0$, then $\mathbf{M}$ is safe as well.

At this point, we can already describe how the interpretation of formulas in Gödel logic can be used as a basis of a query system and put it with correspondence to the query system based on relational operations. We describe the query system only to the extent to be able to derive conclusions on the invariance to ordinal transformations because a detailed description of relational calcluli is beyond the scope and need of this paper. Interested readers can find more details on pseudo-tuple calculus in [35].

Now, consider any finite $\mathbf{L}$-structure $\mathbf{M}$ (i.e., every $\mathbb{r}^{\mathbf{M}}$ is finite in the same sense as above) and a formula φ with free variables $x_1, \ldots, x_n$. Under this notation, $\mathbf{M}$ and φ induce a map $\mathcal{D}_{\mathbf{M},\varphi}\colon \mathrm{Tupl}(R) \to L$, where $R = \{x_1, \ldots, x_n\}$ and

$$\big(\mathcal{D}_{\mathbf{M},\varphi}\big)(r) = ||\varphi||_{\mathbf{M},v} \tag{78}$$

such that $r(x_i) = v(x_i)$ for all $i = 1, \ldots, n$. Clearly, $\mathcal{D}_{\mathbf{M},\varphi}$ given by (78) is a ranked data table on R (free variables in φ are considered as names of attributes) and it can be seen as a result of a query given by φ in a database instance represented by the safe $\mathbf{L}$-structure $\mathbf{M}$.

Remark 8. Let us note that $\mathcal{D}_{\mathbf{M},\varphi}$ is defined correctly by (78) because $||\varphi||_{\mathbf{M},v}$ depends only on $\mathbf{M}$-valuation of variables which appear free in φ. Also note that in the definition of $\mathcal{D}_{\mathbf{M},\varphi}$, we have tacitly assumed that variables in φ are used as attribute names and, at the same time, we have disregarded their types. An explicit (and rigorous) treatment of types can be incorporated but it does not bring new insight into the invariance issues and we therefore use this simplification. The role of $\mathbf{L}$-structures as database instances is basically the same as in the classic model except for the fact that each $\mathbb{r}^{\mathbf{M}}$ represents an RDT instead of a classic relation. Indeed, a propositional symbol $\mathbb{r}$ may be seen as a name and $\mathbb{r}^{\mathbf{M}}$ (the interpretation of $\mathbb{r}$ in $\mathbf{M}$) may be seen as a current value of $\mathbb{r}$ considering $\mathbf{M}$.

The equality of the considered query systems can be proved by showing that for a query formulated in one of the systems there is a corresponding equivalent query in the second one and *vice versa.* The arguments are similar as in the ordinary non-ranked model and we therefore focus only on the essential differences.

From Relational Operations to Queries in Gödel Logic Let us assume that $\mathcal{D}$ is a result of a query which uses RDTs $\mathcal{D}_1, \ldots, \mathcal{D}_n$, restriction conditions $\theta_1, \ldots, \theta_k$, and operations $\bowtie$, σ, π, $\cup$, $\div$, and renaming (in the ordinary sense). Then, there is a finite $\mathbf{M}$ and a formula φ such that $\mathcal{D}$ coincides with $\mathcal{D}_{\mathbf{M},\varphi}$. The construction of $\mathbf{M}$ and φ is straightforward and goes along the same lines as in the ordinary case except for the fact that the division is not a derivable operation. First, let $\mathbf{M}$ be an RDT where each RDT $\mathcal{D}_i$ is represented by $\mathbb{r}_i^{\mathbf{M}}$ and each restriction condition θ_j is represented by $\mathbb{s}_j^{\mathbf{M}}$. Observe that since we consider only finitely many input RDTs, the universe of $\mathbf{M}$ can be considered as a finite set and all $\mathbb{s}_j^{\mathbf{M}}$'s can be restricted to this finite universe. In case of queries resulting by $\bowtie$, σ, π, and $\cup$, the desired formula is constructed as in the classic case from formulas corresponding to subqueries. For instance, let us assume that the query is of the form of a projection onto $S = \{y_1, \ldots, y_p\}$ for $p \geq 0$ and its subquery (the argument for the projection) produces an RDT on $R = \{y_1, \ldots, y_q\}$ for $q \geq p$. If we assume that a formula ψ is a counterpart to the subquery, then the counterpart of the projection is

$$(\exists y_{p+1}) \cdots (\exists y_q)\psi, \tag{79}$$

i.e., the same formula as in the classic case. In case of the division, which is not a fundamental operation, we proceed analogously. Namely, we use a formula

$$\vartheta \wedge (\forall y_1) \cdots (\forall y_p)(\psi \Rightarrow \chi), \tag{80}$$

where ψ, χ, and ϑ are formulas corresponding to subqueries, and $\{y_1, \ldots, y_p\}$ is the set of all attributes which are common to the results of subqueries corresponding to ψ and χ, cf. (34). Altogether, query of arbitrary complexity formulated in terms of the relational operations with RDTs can equivalently be expressed by a formula of Gödel logic.

From Queries in Gödel Logic to Relational Operations Conversely, consider any finite $\mathbf{L}$-structure $\mathbf{M}$ with a universe M and a formula φ. Let $\mathcal{D}_M$ denote an RDT on $\{y\}$ such that $\{r(y); \mathcal{D}_M(r) = 1\} = M$ and $L(\mathcal{D}_M) = \{1\}$. Since $\mathbf{M}$ is finite, such an RDT always exists. Let 0_R denote an empty RDT on R (i.e., the answer set of 0_R is empty). Under this notation, one can construct a relational expression which involves $\mathcal{D}_M$, finitely many RDTs 0_R, RDTs corresponding to all $\mathbb{r}^{\mathbf{M}}$, and relational operations $\bowtie$, σ, π, $\cup$, $\rightarrow$, and $\div$ such that $\mathcal{D}_{\mathbf{M},\varphi}$ coincides with the value of the expression. Again, the construction is fully analogous to the classic one except for the fact that we consider two fundamental quantifiers and fundamental connectives $\overline{0}$ and $\Rightarrow$ which cannot be defined in terms of the other ones. Indeed, if φ is $\overline{0}$, the corresponding expression is $0_\emptyset$, i.e., 0_R for $R = \emptyset$. If φ is $\mathbb{r}(x_1, \ldots, x_n)$, then we can consider the RDT corresponding to $\mathbb{r}^{\mathbf{M}}$. For φ being either of $\psi \wedge \chi$ and $\psi \Rightarrow \chi$, we utilize the relational operations $\bowtie$ and $\rightarrow$ in conjunction with $\mathcal{D}_M$ (and optionally the renaming of attributes); note here that as in the classic case, ψ and χ may have different sets of variables which appear free in ψ and χ, respectively. If φ is $(\exists x)\psi$, we proceed as in the classic case using π and $\bowtie$. For φ being $(\forall x)\psi$, the expression is built using $\div$ and $\mathcal{D}_M$. Namely, the expression is of the form $\mathcal{D}_\psi \div^{\mathcal{D}} \mathcal{D}_x$, where (i) $\mathcal{D}_x$ is $\mathcal{D}_M$ with the attribute y renamed to x; (ii) $\mathcal{D}$ is a join of finitely many $\mathcal{D}_{y_1}, \ldots, \mathcal{D}_{y_n}$ such that all free variables in ψ except for x are exactly $y_1, \dots, y_n$ and each $\mathcal{D}_{y_i}$ results from $\mathcal{D}_M$ by renaming y to y_i; (iii) $\mathcal{D}_\psi$ results by the expression corresponding to ψ.

Owing to the correspondence between the query system based on relational operations with RDTs and the system based on evaluating formulas of Gödel logic, we conclude that every query formulated by a formula of Gödel logic is invariant to ordinal transformations. This observation is a direct consequence of Theorem 6 and Theorem 7 and is summarized in the following corollary.

Corollary 8. *Let φ be a formula of language $\mathcal{J}$ and $\mathbf{M}$ be a finite $\mathbf{L}$-structure for $\mathcal{J}$. Furthermore, let f be an order embedding such that $f(0) = 0$ and $f(1) = 1$. Then,*

$$\mathcal{D}_{\mathbf{M},\varphi} \circ f = \mathcal{D}_{\mathbf{M}\circ f,\varphi}, \tag{81}$$

where $\mathbf{M} \circ f$ is a finite $\mathbf{L}$-structure for $\mathcal{J}$ such that $\mathbb{r}^{\mathbf{M}\circ f} = \mathbb{r}^{\mathbf{M}} \circ f$ for any relation symbol $\mathbb{r}$ of the language $\mathcal{J}$. □

We now turn our attention to the axiomatization of Gödel logic and its consequences for the query systems. Gödel logic has a complete Henkin-style axiomatization, i.e., a special deductive system. The axiomatization can be used to find proofs of properties of relational operations owing to the relationship between the two query systems considered in this section. The deductive system (for the language $\mathcal{J}$) consists of the following axioms of logical connectives:

$$\varphi \Rightarrow (\varphi \wedge \varphi), \tag{82}$$
$$(\varphi \Rightarrow \psi) \Rightarrow ((\psi \Rightarrow \chi) \Rightarrow (\varphi \Rightarrow \chi)), \tag{83}$$
$$(\varphi \wedge \psi) \Rightarrow \varphi, \tag{84}$$
$$(\varphi \wedge \psi) \Rightarrow (\psi \wedge \varphi), \tag{85}$$
$$(\varphi \Rightarrow (\psi \Rightarrow \chi)) \Rightarrow ((\varphi \wedge \psi) \Rightarrow \chi), \tag{86}$$
$$((\varphi \wedge \psi) \Rightarrow \chi) \Rightarrow (\varphi \Rightarrow (\psi \Rightarrow \chi)), \tag{87}$$
$$((\varphi \Rightarrow \psi) \Rightarrow \chi) \Rightarrow (((\psi \Rightarrow \varphi) \Rightarrow \chi) \Rightarrow \chi), \tag{88}$$
$$\overline{0} \Rightarrow \varphi, \tag{89}$$

where φ, ψ, χ are arbitrary formulas of $\mathcal{J}$. In addition to the logical axioms, we admit the following axioms of substitution

$$(\forall x)\varphi \Rightarrow \varphi(x/y), \tag{90}$$
$$\varphi(x/y) \Rightarrow (\exists x)\varphi, \tag{91}$$

where x and y are variables such that y is free for x in φ in the usual sense, i.e., no free occurrence of x in φ lies within the scope of a quantifier which binds y, see [31]. Furthermore, we assume the following axioms of the distribution:

$$(\forall x)(\varphi \Rightarrow \psi) \Rightarrow (\varphi \Rightarrow (\forall x)\psi), \tag{92}$$
$$(\forall x)(\psi \Rightarrow \varphi) \Rightarrow ((\exists x)\psi \Rightarrow \varphi), \tag{93}$$
$$(\forall x)(\varphi \vee \psi) \Rightarrow (\varphi \vee (\forall x)\psi), \tag{94}$$

where φ is an arbitrary formula such that x is not free in φ. In addition to the axioms (82)–(94), the deductive system consists of deduction rules *modus ponens* "from φ and $\varphi \Rightarrow \psi$ infer ψ" (i.e., the law of detachment) and *generalization* "from φ infer $(\forall x)\varphi$". As usual, a proof by a set Σ of formulas is a finite sequence $\varphi_1, \ldots, \varphi_n$ where each φ_i is a logical axiom or a formula in Σ or it is derived by modus ponens or generalization from preceding formulas in the sequence; φ is provable by Σ, denoted $\Sigma \vdash \varphi$, if there is a proof $\varphi_1, \ldots, \varphi_n$ by Σ such that $\varphi = \varphi_n$.

The notion of provability is one paricular notion of (a syntactic) entailment in the logic. Other notion of entailment—the semantic entailment may be defined based on the notion of an **L**-model. In particular, for φ and a safe **L**-structure **M**, we put

$$||\varphi||_{\mathbf{M}} = \inf\{||\varphi||_{\mathbf{M},v};\ v \text{ is } \mathbf{M}\text{-valuation}\}. \tag{95}$$

Furthermore, a safe **L**-structure **M** is called a model of Σ whenever $||\varphi||_{\mathbf{M}} = 1$ for all $\varphi \in \Sigma$. We put $\Sigma \models \varphi$ and say that φ is semantically entailed by Σ whenever $||\varphi||_{\mathbf{M}} = 1$ for any **L**-model **M** of Σ where **L** is any totally ordered Gödel algebra. For convenience, we write $\vdash \varphi$ and $\models \varphi$ in case of $\Sigma = \emptyset$. The following completeness theorem is established (recall that $[0,1]_{\mathbf{G}}$ denotes the standard Gödel algebra defined on the real unit interval).

Theorem 9 (Completeness of first-order Gödel logic)**.** *Let Σ be any set of formulas of $\mathcal{J}$. The following are equivalent:*

(*i*) $\Sigma \vdash \varphi$*;*

(*ii*) $\Sigma \models \varphi$*;*

(*iii*) $||\varphi||_{\mathbf{M}} = 1$ *for each* $[0,1]_{\mathbf{G}}$*-model of* Σ*;*

(*iv*) *For each* $[0,1]_{\mathbf{G}}$*-structure* **M** *there is* $\psi \in \Sigma$ *such that* $||\psi||_{\mathbf{M}} \leq ||\varphi||_{\mathbf{M}}$*;*

(*v*) *For each* $[0,1]_{\mathbf{G}}$*-structure* **M** *and each* $a \in [0,1]$*:*
if $||\psi||_{\mathbf{M}} \geq a$ *for each* $\psi \in \Sigma$*, then* $||\varphi||_{\mathbf{M}} \geq a$*.*

Proof. See [23, Theorem 5.2.9 and Corollary 5.3.4]. □

Remark 9. Note that the term "completeness" in Theorem 9 refers to the syntactico-semantical completeness of the logic, cf. also [7], and *not* the functional completeness. In fact, the system of connectives used in the logic cannot be functionally complete because $[0,1]_{\mathbf{G}}$ admits uncountably many n-ary functions while the language of the logic and, therefore, the number of different formulas that can be written in the language, is countable. In this sense, the underlying logic of the rank-aware model depart from the classic logic where any n-ary function on $\{0,1\}$ is expressible using (the truth functions of) the fundamental connectives (e.g., $\Rightarrow$ and $\neg$). Also note that Gödel logic is indeed weaker than the classic logic. For instance, $\neg\neg\varphi \Rightarrow \varphi$ is not provable in Gödel logic. As a consequence, the relational operations with RDTs considered in our paper do not satisfy all laws that are satisfied in the classic relational model. For instance, there are $\mathcal{D}_1$ and $\mathcal{D}_2$ on the same relation scheme such that $\mathcal{D}_1 \cap \mathcal{D}_2 \neq \mathcal{D}_1 - (\mathcal{D}_1 - \mathcal{D}_2)$.

As an application of the established connection between the relational operations with RDTs and Gödel logic, we can introduce a derived operation of a semijoin. In the classic mode, a semijoin of $\mathcal{D}_1$ and $\mathcal{D}_2$ on R_1 and R_2, respectively, may be introduced by $\pi_{R_1}(\mathcal{D}_1 \bowtie \mathcal{D}_2)$ or, equivalently, by $\mathcal{D}_1 \bowtie \pi_{R_1 \cap R_2}(\mathcal{D}_2)$. From the perspective of Gödel logic, $\pi_{R_1}(\mathcal{D}_1 \bowtie \mathcal{D}_2)$ can be represented by an **L**-stricture $\mathbf{M}$ with $\mathbb{r}_1^{\mathbf{M}}$ and $\mathbb{r}_2^{\mathbf{M}}$ corresponding to $\mathcal{D}_1$ and $\mathcal{D}_2$, respectively, and a formula

$$(\exists z_1) \cdots (\exists z_k)(\mathbb{r}_1(x_1, \ldots, x_i, y_1, \ldots, y_j) \wedge \mathbb{r}_2(y_1, \ldots, y_j, z_1, \ldots, z_k)). \tag{96}$$

In Gödel logic, the formula is equivalent to

$$\mathbb{r}_1(x_1, \ldots, x_i, y_1, \ldots, y_j) \wedge (\exists z_1) \cdots (\exists z_k)\mathbb{r}_2(y_1, \ldots, y_j, z_1, \ldots, z_k). \tag{97}$$

This is a direct consequence of the fact that

$$\vdash (\exists x)(\varphi \wedge \psi) \Leftrightarrow (\varphi \wedge (\exists x)\psi) \tag{98}$$

provided that x is not free in φ, [23, Lemma 5.1.21]. Observe that (97) is in a correspondence with $\mathcal{D}_1 \bowtie \pi_{R_1 \cap R_2}(\mathcal{D}_2)$. Therefore, in our setting, we also have

$$\pi_{R_1}(\mathcal{D}_1 \bowtie \mathcal{D}_2) = \mathcal{D}_1 \bowtie \pi_{R_1 \cap R_2}(\mathcal{D}_2) \tag{99}$$

as in the classic model which allows us to define a semijoin of ranked data tables by the expression on either side of the equality in (99). In addition, owing to the observations in Theorem 6, the semijoin is also invariant to the ordinal scaling which follows directly by (52) and (55).

7 Computational Issues and Relationship to Other Approaches

The primary interest of our paper is the invariance to ordinary scaling. In this section, we make a digression and comment on algorithms for evaluating particular monotone queries and the relationship to other rank-aware approaches. We show that our observations on the connection of the relational operations with RDTs and Gödel logic can be used to derive laws for query transformations. In addition, we show that the algorithm for computing top-k query results [15] fits well into our formal model. Finally, we show that the observations on the invariance to ordinal transormations can also be applied in the approach by [28].

One of the crucial aspects of any model of data from the point of view of its applicability is the possibility to transform general queries to efficient logical and then physical query plans. In this section, we show that for a fragment of the discussed relational operations, one can consider similar transformations of logical query plans, i.e., transformations of expressions composed of relational operations with RDTs to equivalent expressions which are more suitable for an efficient execution, as in the ordinary relational model of data. We focus only on issues which are specific to our model.

First, we consider laws concerning natural join, projections, selections, and unions and show that our operations with RTDs admit important transformation laws which are used in the ordinary relational model.

Theorem 10. *Let $\mathcal{D}_1$ and $\mathcal{D}_2$ be RDTs on relation schemes R_1 and R_2, respectively. Then, the following properties hold true.*

(*i*) *If $\theta : \mathrm{Tupl}(R_1 \cup R_2) \to L$ and $\theta_1 : \mathrm{Tupl}(R_1) \to L$ are restriction conditions such that $\theta_1(r_1) = \theta(r_1 r_2)$ for any $r_1 \in \mathrm{Tupl}(R_1)$ and $r_2 \in \mathrm{Tupl}(R_2)$ which are joinable, then $\sigma_\theta(\mathcal{D}_1 \bowtie \mathcal{D}_2) = \sigma_{\theta_1}(\mathcal{D}_1) \bowtie \mathcal{D}_2$.*

(*ii*) *If $\theta_1 \colon \mathrm{Tupl}(R_1) \to L$ and $\theta \colon \mathrm{Tupl}(R) \to L$ are restriction conditions such that $\theta(r) = \theta(rr')$ for all $r \in \mathrm{Tupl}(R)$ and $r' \in \mathrm{Tupl}(R_1 \setminus R)$, then $\pi_R(\sigma_{\theta_1}(\mathcal{D}_1)) = \sigma_\theta(\pi_R(\mathcal{D}_1))$.*

(*iii*) *If $R_1 = R_2$ and $R \subseteq R_1$, then $\pi_R(\mathcal{D}_1 \cup \mathcal{D}_2) = \pi_R(\mathcal{D}_1) \cup \pi_R(\mathcal{D}_2)$.*

(*iv*) *If $S \subseteq R \subseteq R_1$, then $\pi_S(\pi_R(\mathcal{D}_1)) = \pi_S(\mathcal{D}_1)$.*

Proof. The assertion can be proved by observing formulas of Gödel logic corresponding to the equalities appearing in (*i*)–(*iv*) and considering the properties of $\vdash$ in Gödel logic. In case of (*i*), $\sigma_\theta(\mathcal{D}_1 \bowtie \mathcal{D}_2)$ can be represented by a formula $\theta \wedge (\varphi \wedge \psi)$ and, analogously, $\sigma_{\theta_1}(\mathcal{D}_1) \bowtie \mathcal{D}_2$ can be represented by a formula $(\theta_1 \wedge \varphi) \wedge \psi$ (we have tacitly identified restriction conditions with formulas). Therefore, (*i*) follows by the associativity of $\wedge$, i.e.,

$$\vdash (\theta \wedge (\varphi \wedge \psi)) \Leftrightarrow ((\theta \wedge \varphi) \wedge \psi),$$

see [23, Lemma 2.2.15], and the relationship of θ_1 and θ. Analogously, (*ii*) is a consequence of (98); (*iii*) is a consequence of

$$\vdash (\exists x)(\varphi \vee \psi) \Leftrightarrow (\varphi \vee (\exists x)\psi)$$

provided that x is not free in φ, see [23, Lemma 5.1.21]. Finally, (*iv*) follows directly by the fact that the left-hand and right-hand sides of the equality in (*iv*) translate into a single formula of Gödel logic of the form $(\exists x_1) \cdots (\exists x_n)\varphi$. □

As a consequence of Theorem 10 and (99), we may conclude that the usual optimization techniques based on pushing down restrictions and projections [19] still work in the ranke-aware model because the classic laws on which the optimizations are based are preserved in Gödel logic. Therefore, many monotone queries in the rank-aware model can be transformed into expressions of the form

$$\mathcal{D}_1 \bowtie \mathcal{D}_2 \bowtie \cdots \bowtie \mathcal{D}_n, \tag{100}$$

where $\mathcal{D}_i$ for $i = 1, \ldots, n$ are RDTs on R_i which result by computing projections and/or restrictions of RDTs representing base data (i.e., RDTs bound to relation variables in a database instance). Clearly, a tuple $r_1 r_2 \cdots r_n$, where $r_i \in \mathrm{Tupl}(R_i)$ for each $i = 1, \ldots, n$, belongs to the answer set of (100) iff all $r_1, r_2, \ldots, r_n$ are joinable and its score is

$$\inf\{\mathcal{D}_1(r_1), \mathcal{D}_2(r_2), \ldots, \mathcal{D}_n(r_n)\} > 0. \tag{101}$$

Hence, (100) may be understood as a query in a similar form as (1) with a few minor conceptual differences: (i) $\mathcal{D}_i$ in may not represent a result of an atomic query as in (1), (ii) we always consider inf as the interpretation of &, and (iii) the objects which match queries are in fact tuples constructed as joins of joinable tuples considered on general schemes $R_1, R_2, \ldots, R_n$. Nevertheless, in case one wants to compute only top k matches, i.e., if one wants to compute only a portion of the answer set of (100) consisting only of k tuples with highest scores, we can directly adopt the Fagin algorithm [15], namely, its improved version which consideres inf as the aggregation function, see [15, Theorem 4.4], provided that each $\mathcal{D}_i$ allows an efficient "sorted access" (tuples in the answer set of $\mathcal{D}_i$ may be listed sorted by scores in the descending order) which may be assumed in many natural situations. Except for the technical difference in using "joinable tuples", see (iii) above, the Fagin algorithm does not need to be further modified. Interested readers are refered to [15] for details and complexity analysis.

We now turn our attention to RankSQL and the extended relational algebra proposed in [28] which is arguably one of the most influential approaches to ranking in relational databases. Similar observation as in Section 5 can be made in the rank-relational approach described in their paper. Recall that according to [28], the basic structure which serves as a counterpart of the classic relations on relation schemes is a *rank-relation* $R_{\mathcal{P}}$ which is understood as a classic relation R equipped with *scores* and (strict total) tuple *order* $<_{R_{\mathcal{P}}}$ based on the scores. The score for each tuple $r \in R$ is computed as a result of a general (monotonic) *scoring function* $\mathcal{F}$ which is applied to *predicate scores* $p_i[r]$ of the tuple r. The predicate scores represent individual ranking criteria (like low price, high availability, close distance between locations, etc.) called predicates and denoted by $p_1, \ldots, p_n$. Note that $\mathcal{P}$ (called the *set of evaluated predicates*) is always a subset of $\{p_1, \ldots, p_n\}$ and the rank-relational model and its implementation relies on the *ranking principle* [28, page 133] based on *maximal possible scores* of tuples in R given $\mathcal{P}$, i.e., each $p_i[r]$ for which $p_i \notin \mathcal{P}$ (p_i is not evaluated) is considered to have the application-specific maximal possible value of p_i. Therefore, for general $\mathcal{P}$, each tuple $r \in R$ has its maximal possible score denoted $\overline{\mathcal{F}}_{\mathcal{P}}[r]$ and $<_{R_{\mathcal{P}}}$ (the tuple order in R given $\mathcal{P}$) is introduced based on such scores, namely, $r_1 <_{R_{\mathcal{P}}} r_2$ whenever $\overline{\mathcal{F}}_{\mathcal{P}}[r_1] < \overline{\mathcal{F}}_{\mathcal{P}}[r_2]$. Furthermore, queries in RankSQL are transformed into expressions of rank-relational algebra which introduces operations with rank-relations including restriction, union, intersection, difference, theta-join, and *rank*—a new operation which produces $R_{\mathcal{P}\cup\{p\}}$ based on $R_{\mathcal{P}}$ and $p \notin \mathcal{P}$. Let us stress that the operations with rank-relations indeed operate on rank-relations, i.e., based on scores and tuple orders of the input arguments, the operations define scores and tuple order of the result. For instance, in case of the union of $R_{\mathcal{P}_1}$ and $S_{\mathcal{P}_2}$, the result is a rank-relation $(R \cup S)_{\mathcal{P}_1 \cup \mathcal{P}_2}$ in which $r_1 <_{(R\cup S)_{\mathcal{P}_1\cup\mathcal{P}_2}} r_2$ whenever $\overline{\mathcal{F}}_{\mathcal{P}_1\cup\mathcal{P}_2}[r_1] < \overline{\mathcal{F}}_{\mathcal{P}_1\cup\mathcal{P}_2}[r_2]$.

From our perspective, we may view an important special case of the rank-relational approach in [28] as follows: We consider $\mathbf{L}$ (the structure of scores) as a totally ordered complete lattice and we let $\mathcal{F}$ be inf. That is, the scoring function always computes the minimum of given predicate scores and 1 represents the maximal possible score. In this setting, rank-relations may be viewed as RDTs with the order of tuples given implicitly by the scores; predicates p_i may be viewed as general restriction conditions, and the rank operator may be seen as a general restriction (29). Moreover, for the rank-relational querying, we may ask the same question as before: Does an ordinal transformation of the input ranking criteria yield the same results? The answer is positive. In a more detail, let $f : L \to L$ be an order embedding which preserves 0 and 1. Then, for each p_i (which is in fact a map from the set of all tuples on the scheme of R to L) we may consider the composed map $p_i \circ f$ and put $\mathcal{P} \circ f = \{p \circ f;\, p \in \mathcal{P}\}$. With respect to the above-mentioned interpretation of evaluated predicates, $\mathcal{P} \circ f$ may be seen as a set of ordinally transformed evaluated predicates. Moreover,f is an order embedding and for $\cup$ defined as above, we have

$$\begin{aligned}
&r_1 <_{(R \cup S)_{\mathcal{P}_1 \cup \mathcal{P}_2}} r_2 \text{ iff}\\
&\overline{\mathcal{F}}_{\mathcal{P}_1 \cup \mathcal{P}_2}[r_1] < \overline{\mathcal{F}}_{\mathcal{P}_1 \cup \mathcal{P}_2}[r_2] \text{ iff}\\
&f\big(\overline{\mathcal{F}}_{\mathcal{P}_1 \cup \mathcal{P}_2}[r_1]\big) < f\big(\overline{\mathcal{F}}_{\mathcal{P}_1 \cup \mathcal{P}_2}[r_2]\big).
\end{aligned}$$

Now, using the fact that $\mathcal{F}$ is inf, it follows that $f\big(\overline{\mathcal{F}}_{\mathcal{P}_1 \cup \mathcal{P}_2}[r]\big) = \overline{\mathcal{F}}_{(\mathcal{P}_1 \circ f) \cup (\mathcal{P}_2 \circ f)}[r]$ for all $r \in R$. Hence, by the definition of $<_{(R \cup S)_{\mathcal{P}_1 \cup \mathcal{P}_2}}$ and $<_{(R \cup S)_{(\mathcal{P}_1 \circ f) \cup (\mathcal{P}_2 \circ f)}}$,

$$\begin{aligned}
&r_1 <_{(R \cup S)_{\mathcal{P}_1 \cup \mathcal{P}_2}} r_2 \text{ iff}\\
&\overline{\mathcal{F}}_{(\mathcal{P}_1 \circ f) \cup (\mathcal{P}_2 \circ f)}[r_1] < \overline{\mathcal{F}}_{(\mathcal{P}_1 \circ f) \cup (\mathcal{P}_2 \circ f)}[r_2] \text{ iff}\\
&r_1 <_{(R \cup S)_{(\mathcal{P}_1 \circ f) \cup (\mathcal{P}_2 \circ f)}} r_2,
\end{aligned}$$

proving that $(R \cup S)_{\mathcal{P}_1 \cup \mathcal{P}_2}$ and $(R \cup S)_{(\mathcal{P}_1 \circ f) \cup (\mathcal{P}_2 \circ f)}$ are ordinally equivalent. One may proceed the same way for the other operations of the rank-relation algebra, see [28, page 134]. As a consequence, ordinal transformations do not have any effect on the results of top-k queries—scores of tuples may be different, however, the order in which tuples appear in the result is the same.

Finally, let us note that the approach in [28] is conceptually similar to ours in that both are capable to answer queries by relations with tuples annotated by scores which indicate degrees of matches of user preferences. It should be noted, however, that the approaches are technically different (even if we consider $\mathcal{F}$ as inf). More detailed on the technical differences can be found in [35].

8 Conclusion

Notions of ordinal containment and ordinal equivalence of relations consisting of tuples annotated by scores have been proposed. The ordinal containment and equivalence have been characterized in terms of the existence of suitable order-preserving functions and order isomorphisms between subsets of scores. It has been shown that infima-based algebraic operations with ranked relations are invariant to ordinal transformations: Queries applied to original and transformed data yield results which are equivalent in terms of the order given by scores. We have demonstrated that this property is not preserved if one considers algebraic operations with ranked relations based on general aggregation functions like triangular norms (other than the minimum triangular norm). As a result of our observation, we have concluded that under infima-based algebraic operations, the scores in ranked tables have no quantitative meaning. Generality of the result has been demonstrated by applying the observations in an alternative calculus-based query system grounded in

Gödel logic. Furthermore, relationship to other approaches has been investigated with the intention to show the connection to existing algorithms for monotone query evaluation and conceptually similar approaches to ranking in databases.

Acknowledgment

Supported by grant no. `P202/14-11585S` of the Czech Science Foundation.

References

[1] Yael Amsterdamer, Daniel Deutch, and Val Tannen, *Provenance for aggregate queries*, Proc. ACM PODS 2011 (New York, NY, USA), ACM, 2011, pp. 153–164.

[2] Radim Belohlavek, *Ordinally equivalent data: A measurement-theoretic look at formal concept analysis of fuzzy attributes*, Int. J. Approx. Reasoning **54** (2013), no. 9, 1496–1506.

[3] Radim Belohlavek, Lucie Urbanova, and Vilem Vychodil, *Sensitivity analysis for declarative relational query languages with ordinal ranks*, Proc. INAP/WLP 2011 (Hans Tompits, Salvador Abreu, Johannes Oetsch, Jrg Phrer, Dietmar Seipel, Masanobu Umeda, and Armin Wolf, eds.), LNAI, vol. 7773, Springer Berlin Heidelberg, 2013, pp. 58–76.

[4] Radim Belohlavek and Vilem Vychodil, *Query systems in similarity-based databases*, Proc. ACM SAC 2010, 2010, pp. 1648–1655.

[5] Garrett Birkhoff, *Lattice theory*, American Mathematical Society, 1940.

[6] L. Philip Chen C. and Chun-Yang Zhang, *Data-intensive applications, challenges, techniques and technologies: A survey on Big Data*, Information Sciences **275** (2014), 314–347.

[7] Petr Cintula and Petr Hájek, *Triangular norm based predicate fuzzy logics*, Fuzzy Sets and Systems **161** (2010), no. 3, 311–346.

[8] Edgar F. Codd, *A relational model of data for large shared data banks*, Commun. ACM **13** (1970), 377–387.

[9] Nilesh Dalvi, Christopher Ré, and Dan Suciu, *Probabilistic databases: diamonds in the dirt*, Commun. ACM **52** (2009), 86–94.

[10] Christopher J. Date and Hugh Darwen, *Databases, types, and the relational model: The third manifesto*, 3rd ed., Addison-Wesley, 2006.

[11] ———, *Database explorations: Essays on the third manifesto and related topics*, ch. 14: Image Relations, pp. 237–272, Trafford Publishing, 2010.

[12] ———, *Database explorations: Essays on the third manifesto and related topics*, ch. 12: A Brief History of the Relational Divide Operator, pp. 169–198, Trafford Publishing, 2010.

[13] Christopher J. Date, Hugh Darwen, and Nikos A. Lorentzos, *Time and relational theory: Temporal databases in the relational model and SQL*, Morgan Kaufmann, 2014.

[14] Francesc Esteva and Lluís Godo, *Monoidal t-norm based logic: Towards a logic for left-continuous t-norms*, Fuzzy Sets and Systems **124** (2001), no. 3, 271–288.

[15] Ronald Fagin, *Combining fuzzy information from multiple systems*, J. Comput. Syst. Sci. **58** (1999), no. 1, 83–99.

[16] J. Nathan Foster, Todd J. Green, and Val Tannen, *Annotated xml: Queries and provenance*, Proc. ACM PODS 2008 (New York, NY, USA), ACM, 2008, pp. 271–280.

[17] Nikolaos Galatos, Peter Jipsen, Tomacz Kowalski, and Hiroakira Ono, *Residuated Lattices: An Algebraic Glimpse at Substructural Logics, Volume 151*, 1st ed., Elsevier Science, San Diego, USA, 2007.

[18] Bernhard Ganter and Rudolf Wille, *Formal concept analysis: Mathematical foundations*, 1st ed., Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1997.

[19] Hector Garcia-Molina, Jeffrey D. Ullman, and Jennifer Widom, *Database systems: The complete book*, 2 ed., Prentice Hall Press, Upper Saddle River, NJ, USA, 2008.

[20] Joseph A. Goguen, *The logic of inexact concepts*, Synthese **19** (1979), 325–373.

[21] Todd J. Green, *Containment of conjunctive queries on annotated relations*, Proc. ICDT 2009 (New York, NY, USA), ACM, 2009, pp. 296–309.

[22] Todd J. Green, Grigoris Karvounarakis, and Val Tannen, *Provenance semirings*, Proc. ACM PODS 2007 (New York, NY, USA), ACM, 2007, pp. 31–40.

[23] Petr Hájek, *Metamathematics of Fuzzy Logic*, Kluwer Academic Publishers, 1998.

[24] Arend Heyting, *Die formalen Regeln der intuitionistischen Logik*, Sitzungsberichte der Preußischen Akademie der Wissenschaften zu Berlin (1930), 42–65.

[25] Ihab F. Ilyas, George Beskales, and Mohamed A. Soliman, *A survey of top-k query processing techniques in relational database systems*, ACM Comp. Surv. **40** (2008), no. 4, 11:1–11:58.

[26] Tomasz Imieliński and Witold Lipski, Jr., *Incomplete information in relational databases*, J. ACM **31** (1984), no. 4, 761–791.

[27] Erich Peter Klement, Radko Mesiar, and Endre Pap, *Triangular Norms*, Springer, 2000.

[28] Chengkai Li, Kevin Chen-Chuan Chang, Ihab F. Ilyas, and Sumin Song, *Ranksql: query algebra and optimization for relational top-k queries*, Proc. ACM SIGMOD 2005, 2005, pp. 131–142.

[29] Xiang Lian and Lei Chen, *Probabilistic top-k dominating queries in uncertain databases*, Information Sciences **226** (2013), 23–46.

[30] David Maier, *Theory of Relational Databases*, Computer Science Press, Rockville, MD, USA, 1983.

[31] Elliott Mendelson, *Introduction to Mathematical Logic*, Chapman and Hall, 1987.

[32] H.T.H. Nguyen and J. Cao, *Trustworthy answers for top-k queries on uncertain Big Data in decision making*, Information Sciences **318** (2015), 73–90.

[33] Ewa Orłowska and Anna Maria Radzikowska, *Double residuated lattices and their applications*, Relational Methods in Computer Science (Harrie C. M. Swart, ed.), LNCS, vol. 2561, Springer Berlin Heidelberg, 2002, pp. 171–189.

[34] Cecylia Rauszer, *Semi-boolean algebras and their applications to intuitionistic logic with dual operations*, Fund. Math. **83** (1973), 219–249.

[35] Ondrej Vaverka and Vilem Vychodil, *Relational division in rank-aware databases*, CoRR **abs/1507.00541** (2015).

[36] Dong Wang, Lei Zou, and Dongyan Zhao, *Top-k queries on RDF graphs*, Information Sciences **316** (2015), 201–217.