Rank Neutrosophic Armstrong Axioms and Functional Dependencies

Meena Arora¹ & Ranjit Biswas²

¹Department of CSE, JSS Academy of Technical Education, Noida, India
²Department of CSE, Manav Rachna University, Faridabad, India
Email: meena23dec@gmail.com, ranjitbiswas@yahoo.com

ABSTRACT

Managers, in today's corporations, rely increasingly on the use of databases to obtain insights and updated information to make their decisions. Hence, queries in natural language with pre-defined syntactical structures are performed. I will propose a new type of Functional dependency termed as rank neutrosophic functional dependency for the searching techniques using the neutrosophic theory to meet the predicates posed in natural language in order to answer imprecise queries of the lay users. It may be claimed that the method could be well incorporated in the existing commercial query languages so that the users of any level of knowledge can get some results to his queries. I will develop such an Armstrong's Axioms and Functional Dependencies for neutrosophic-search as well as rank neutrosophic sets and relations such that they will be new and of fundamental nature.

Keywords: Rank Neutrosophic Set, Rank Neutrosophic Relation, Rank Neutrosophic Armstrong Axioms, Rank Neutrosophic Functional Dependencies (RNfd)

1. INTRODUCTION

A good relational database system should be capable of maintaining a good relationship among the data's and generate new relations among the existing data's in the database system. A database is said to be a relational system, when the database system satisfies Codd twelve rules and relational theory in algebra. In a relational database system there exist some relation among the stored data's; it is also possible to form relations from the stored data with the help of relational algebra. An important concept in relational schema design is that of a functional dependency. Normalization theory is based on the functional notion of functional dependency and moreover it helps in simplifying the structure of tables.

A functional dependency is a property of the semantics or meaning of the attributes. The database designers will use their understanding of the semantics of the attributes, how they relate to each one another. Certain FD's can be specified without referring specific relation, but as the property of those attributes. Functional dependency plays a key role in establishing and maintaining a relationship among the data's that are functionally related to one another and they are separated from other non-related data's thus providing clear relationship among the set of data present. Normalization is basically used to eliminate data redundancy and provide data integrity. Axioms or rules of inference provide a simpler technique for reasoning about functional dependencies. We can also use other rules to find the logically implied functional dependencies.

Finally I introduce the concept of Rank Neutrosophic Armstrongs axioms in relational databases. The need for introducing such axioms is justified and explained. A detailed analysis of the new notion of Rank Neutrosophic functional dependencies (RNfd) in relational databases is conducted to verify the so-called Armstrong's axioms of Rank Neutrosophic Relations.

2. NEUTROSOPHIC LOGIC

Neutrosophic logic was created by Florentin Smarandache (1995) [7] and is an extension/combination of the fuzzy logic, intuitionistic logic, paraconsistent logic, and the three-valued logics that use an indeterminate value.

Definition 2.1 Neutrosophic Logic: A logic in which each proposition is estimated to have the percentage of truth in a subset $T$, the percentage of indeterminacy in a subset $I$, and the percentage of falsity in a subset $F$, where $T$, $I$, $F$ are defined below, is called Neutrosophic Logic. Constants: $(T, I, F)$-truth-values, where $n_{inf} = \inf T + \inf I + \inf F \geq 0$, and $n_{sup} = \sup T + \sup I + \sup F \leq 3$.

2.1. Neutrosophic Set

Definition 2.1.1 (Neutrosophic Set): Let $X$ be a space of points (objects), with a generic element in $X$ denoted by $x$. A neutrosophic set $A$ in $X$ is characterized by a truth-membership function $T_A$, an indeterminacy-membership function $I_A$, and a falsity-membership function $F_A$. $T_A(x)$, $I_A(x)$ and $F_A(x)$ are real standard or non-standard subsets
of $0, 1^*$. That is

$$T_A : X \to \{0, 1^*\},$$

(2.1)

$$I_A : X \to \{0, 1^*\},$$

(2.2)

$$F_A : X \to \{0, 1^*\}.$$  

(2.3)

There is no restriction on the sum of $T_A(x), I_A(x)$ and $F_A(x)$ so $0 \leq \sup T_A(x) + \sup I_A(x) + \sup F_A(x) \leq 3^*$. 

2.2. Ranking

Ranking is defined as computing a similarity score between a tuple and the query,

Consider the query $Q = \text{SELECT}* \text{ FROM} R \text{ WHERE} A_1 = v_1 \text{ and ... and } A_m = v_m$

Query is a vector: $Q = (v_1, ..., v_m)$; Tuple is a vector: $T = (u_1, ..., u_m)$

Consider the applications: personalized search engines, shopping agents, logical user profiles, and soft catalogs. To answer the queries related with the above application, two approaches are given:

- Qualitative $\rightarrow$ Pare to semantics (deterministic);
- Quantitative $\rightarrow$ alter the query ranking.

3. RANK NEUTROSOPHIC SET

In this section, we define Rank Neutrosophic sets in an approximation space $(U; R)$ and set theoretic operations on them. RNS is an instance of neutrosophic set which can be used in real scientific and engineering applications.

A Rank Neutrosophic set $A$ of a set $U$ with $t_A(u) = 0$ and $f_A(u) = 1 \forall u \in U$ is called the zero Rank Neutrosophic set of $U$. A Rank Neutrosophic set $A$ of a set $U$ with $t_A(u) = 1$ and $f_A(u) = 0, \forall u \in U$ is called the unit Rank Neutrosophic set of $U$.

3.1. Rank Neutrosophic – cut $((\alpha, \beta) - \text{cut})$

For $\alpha, \beta \in [0,1]$, the objects $(\alpha, \beta)$ – cut and $(\alpha)$-cut of a Rank Neutrosophic set are defined in [1] in the following way.

Definition 3.1 $(\alpha, \beta)$ – cut or Rank Neutrosophic-cut.

Let $A$ be a Rank Neutrosophic set of a universe $X$ with the true-membership function $t_A$ and the false-membership function $f_A$. The $(\alpha, \beta)$ – cut of the Rank Neutrosophic set $A$ is a crisp subset $A(\alpha, \beta)$ of the set $X$ given by

$$A(\alpha, \beta) = \{x : x \in X; V_A(x) \geq [(\alpha, \beta)]\}$$

Clearly $A(0; 0) = X$. The $(\alpha, \beta)$-cuts are also called Rank Neutrosophic cuts of the Rank Neutrosophic set $A$.

4. RANK NEUTROSOPHIC RELATION

Let $X$ and $Y$ be two universes. A Rank Neutrosophic relation (RNR) denoted by $R(X \rightarrow Y)$ of the universe $X$ with the universe $Y$ is a Rank Set of the Cartesian product $X \times Y$. The true membership value $t_R(x, y)$ estimates the strength of the existence of the relation of R-type of the object $x$ with the object $y$, whereas the false membership value $f_R(x, y)$ estimates the strength of the non-existence of the relation of R-type of the object $x$ with the object $y$. The relation $R(X \rightarrow Y)$ could be in short denoted by the notation $R$, if there is no confusion.

Example: Consider two universes $X = \{a, b\}$ and $Y = \{p, q, r\}$. Let $R$ be a RNR of the universe $X$ with the universe $Y$ proposed by an intelligent agent as shown by the following Table 5.1:

<table>
<thead>
<tr>
<th>$(X \rightarrow Y)$</th>
<th>$p$</th>
<th>$q$</th>
<th>$r$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x$</td>
<td>(0.7, 0.2)</td>
<td>(0.3, 0.5)</td>
<td>(0.8, 0.2)</td>
</tr>
<tr>
<td>$y$</td>
<td>(0.2, 0.4)</td>
<td>(0.7, 0.3)</td>
<td>(0.4, 0.4)</td>
</tr>
</tbody>
</table>

The proposed RNR reveals the strength of Rank Neutrosophic relation of every pair of $X \times Y$.

5. RANK NEUTROSOPHIC ARMSTRONGS AXIOMS AND FUNCTIONAL DEPENDENCIES (RNFDs)

Finally, I introduce the concept of Rank Neutrosophic Functional Dependencies in relational databases. The need for introducing such dependencies is justified and explained. Then a detailed analysis of the new notion of Rank Neutrosophic functional dependencies (RNfd) in relational databases is conducted to verify the so-called Armstrong’s axioms of Rank Neutrosophic Relations.

I have considered a relational database (Codd’s model, [8]) and indicate a new type of functional dependency, called rank neutrosophic functional dependency (RNfd). In the Codd’s model of a relational database, the real world of interest is expressed by means of relations. Implementation of this model is in terms of precise data only. The comparison of data of the same data types is done with classical logic. But, in real-life applications, the data associated are often imprecise, Rank Neutrosophic or non-deterministic. All real data cannot be precise because of their fuzzy or intuitionistic fuzzy nature. Consequently, for comparing such data, classical logic is not appropriate. The most important concept in relational databases is that of the functional dependency of one set of attributes upon another.

I have observed that my approach is not a generalization of the type of fuzzy functional dependencies defined in [1], [2], [3], [5], [6]).

My main motivation is to capture the integrity arising out of neutrosophic constraints, and so I need to define a
new type of functional dependency known as rank neutrosophic functional dependency (RNd).

Rank Neutrosophic Functional Dependency (RNd)

Let \( X, Y \subseteq \{A_1, A_2, \ldots, A_n\} \). Choose two parameters \( \alpha, \beta \) in \([0, 1]\) and \( R \) as defined earlier. An Rank Neutrosophic functional dependency (RNd) \( X \rightarrow_{(\alpha, \beta)} Y \) is said to exist if, whenever \( t_1[X] \models_{(\alpha, \beta), R} t_2[X] \) is true it is also the case that \( t_1[Y] \models_{(\alpha, \beta), R} t_2[Y] \) is true.

We say that the set \( X \) of attributes if-functional defines the set \( Y \) of attributes at \((\alpha, \beta)\)-level of choice. In another terminology, the set \( Y \) of attributes is if-functionally defined by the set \( X \) of attributes at \((\alpha, \beta)\)-level of choice. Denote it by the notation \( X \rightarrow_{(\alpha, \beta)} Y \) or, simply by the notation \( X \rightarrow Y \) because of the fact that \( R \) is already fixed and the choices \( \alpha, \beta \) may be left varied during analysis. Call it simply an \((\alpha, \beta) \) — RNd.

The following propositions are straightforward from the above definition of \((\alpha, \beta) \) – RNd.

**Proposition 5.1**

(i) For any subset \( X \) of \( R \) and for any \( \alpha, \beta \) \([0, 1]\),
\[
X \rightarrow_{(\alpha, \beta)} X.
\]

(ii) Suppose that \( 0 \leq \beta_1 \leq \beta_2 \leq 1 \). Then,
\[
X \rightarrow_{(\alpha, \beta_1)} Y \Rightarrow X \rightarrow_{(\alpha, \beta_2)} Y.
\]

(iii) Suppose that \( 0 \leq \alpha_1 \leq \alpha_2 \leq 1 \). Then,
\[
X \rightarrow_{(\alpha_1, \beta)} Y \Rightarrow X \rightarrow_{(\alpha_2, \beta)} Y.
\]

**5.1. Validation of Armstrong’s Axioms with RNd’s**

In this section it is checked up whether Armstrong’s axioms are also true with Rank Neutrosophic functional dependencies. Armstrong’s axioms are the most important base of the theory of relational databases. The results which are true on crisp environment may not be true on fuzzy or neutrosophic environment. Consequently it is necessary now to study the Armstrong’s axioms with RNd’s and to explore the relevant results on the Armstrong’s axioms of neutrosophic nature. Let us call them Rank Neutrosophic Armstrong’s Axioms (RN-N-Armstrong’s-axioms).

Choose any pair of values of the choice parameters \( \alpha, \beta \) in \([0, 1]\). Then the following Propositions are true.

**Proposition (RN-Armstrong’s Axioms) 5.1.1**

(A1) : If \( Y \subseteq X \), then \( X \rightarrow_{(\alpha, \beta)} Y \) RN-Reflexive rule

(A2) : If \( X \rightarrow_{(\alpha, \beta)} Y \), then \( XZ \rightarrow_{(\alpha, \beta)} YZ \) RN-Augmentation rule

(A3) : If \( X \rightarrow_{(\alpha, \beta)} Y \), and \( Y \rightarrow_{(\alpha, \beta)} Z \), then \( X \rightarrow_{(\alpha, \beta)} Z \) RN-Transitive rule

Proof (A1):

Since \( Y \subseteq X \), therefore whenever \( t_1[X] \models_{(\alpha, \beta), R} t_2[X] \) is true, the result \( t_1[Y] \models_{(\alpha, \beta), R} t_2[Y] \) is also true. Hence \( X \rightarrow_{(\alpha, \beta), R} Y \).

Proof (A2):

Suppose that \( t_1[X] \models_{(\alpha, \beta), R} t_2[X] \) ................. (i)

Therefore we have \( t_1[Y] \models_{(\alpha, \beta), R} t_2[Y] \) ................. (ii)

Now suppose that \( t_1[XZ] \models_{(\alpha, \beta), R} t_2[XZ] \) ................. (iii)

From (i) and (iii) we get \( t_1[Z] \models_{(\alpha, \beta), R} t_2[Z] \) ................. (iv)

From (ii) and (iv) we get \( t_1[YZ] \models_{(\alpha, \beta), R} t_2[YZ] \) ................. (v)

From (iii) and (v) we deduce that \( X \rightarrow_{(\alpha, \beta)} Y \).

Hence Proved.

Proof (A3): Assume that both the RNd’s \( X \rightarrow_{(\alpha, \beta)} Y \) and \( Y \rightarrow_{(\alpha, \beta)} Z \) hold in the relation \( R \). Therefore whenever \( t_1[X] \models_{(\alpha, \beta), R} t_2[X] \) is true, \( t_1[Y] \models_{(\alpha, \beta), R} t_2[Y] \) is also true. But whenever \( t_1[Y] \models_{(\alpha, \beta), R} t_2[Y] \) is true, \( t_1[Z] \models_{(\alpha, \beta), R} t_2[Z] \) is also true. Combining these two we see that whenever \( t_1[X] \models_{(\alpha, \beta), R} t_2[X] \) is true, the result \( t_1[Z] \models_{(\alpha, \beta), R} t_2[Z] \) is also true.

Therefore \( X \rightarrow_{(\alpha, \beta)} Z \). Hence Proved.

The axiom (A1) is called RN-Reflexive rule, axiom (A2) is called RN-Augmentation rule and axiom (A3) is called RN-Transitive rule.

**Proposition (RN-Union rule) 5.1.3**

Axiom (A2) : If \( X \rightarrow_{(\alpha, \beta)} YZ \), then \( X \rightarrow_{(\alpha, \beta)} Y \)

Proof :

Given that \( X \rightarrow_{(\alpha, \beta)} YZ \) ............... (i)

Now \( Y \subseteq YZ \). Therefore by (A1), we have
\( Y \rightarrow_{(\alpha, \beta)} Y \) ............... (ii)

Applying (A1) on (i) and (ii) we get \( X \rightarrow_{(\alpha, \beta)} Y \)

Hence Proved.

**Proposition (RN-Union rule) 5.1.3**

Axiom (A3) : If \( X \rightarrow_{(\alpha, \beta)} Y \) and \( X \rightarrow_{(\alpha, \beta)} Z \), then
\( X \rightarrow_{(\alpha, \beta)} YZ \)

Proof:

Given that \( X \rightarrow_{(\alpha, \beta)} Y \) ............... (i)

and \( X \rightarrow_{(\alpha, \beta)} Z \) ............... (ii)

From (i) we can write \( X \rightarrow_{(\alpha, \beta)} XY \) .......(iii)

From (ii) we can write \( XY \rightarrow_{(\alpha, \beta)} YZ \) .... (iv)
From (iii) and (iv) we get $X \xrightarrow{(\alpha, \beta) R} YZ$

Hence Proved.

The following propositions can be proved similarly.

**Proposition (RN-pseudotransitive rule) 5.1.4**

$(A_d)$ If $X \xrightarrow{(\alpha, \beta) R} Y$ and $WY \xrightarrow{(\alpha, \beta) R} Z$, then $WX \xrightarrow{(\alpha, \beta) R} Z$.

**Proposition 5.1.5**

$(A_f)$ If $X \xrightarrow{(\alpha, \beta) R} Y$, then $XZ \xrightarrow{(\alpha, \beta) R} Y$.

Till here we have mentioned the rank neutrosophic integrity constraints in relational databases with the notion of rank neutrosophic functional dependency (RNfd). In the logical design of a relational database, integrity constraints play a vital role.

4. **CONCLUSION**

Among the integrity constraints, data dependencies constitute an important class. We can not ignore the integrity constraints of neutrosophic nature which are very common in database designs. For example, an neutrosophic integrity constraint like, “salary of two lecturers of almost same age will be approximately equal” is certainly a constraint to a database designer which he can not accommodate in his design with the help of crisp type of constraints. It is clear from the derivation that the notion of RNfd is not an extension of any existing model of fuzzy functional dependency (ffd). Armstrong’s axioms are the most important base of the theory of relational databases has also been worked on here. It is true that the results which are true on crisp environment may not be true on neutrosophic environment. So, I have studied the Armstrong’s axioms with RNfds and I have verified few results on the Armstrong’s axioms of neutrosophic nature. The Armstrong’s axioms dealing with RNfd are called by RN-Armstrong’s-axioms. In another terminology, RNfd is not just an extension of the existing notion of ffd([2], [3], [4], [5], [8]). The major drawback of these existing concepts is that the comparison of two data of a domain is done with the help of one type of fuzzy equality relations, which are not equivalence relations. This problem is solved with the help of RNfd.

**REFERENCES**


