ARC '16

مؤتمر مؤسسة قطر
السنوي للبحوث

QATAR FOUNDATION
ANNUAL RESEARCH
CONFERENCE

Towards World-class
Research and Innovation

Information Communications Technology Pillar

http://dx.doi.org/10.5339/qfarc.2016.ICTPP3084

Conceptual-based Functional Dependency Detection Framework

Fahad Islam

Qatar University, QA

Email: fahadi@qu.edu.qa

Nowadays, knowledge discovery from data is one of the challenging problems, due to its importance in different fields such as; biology, economy and social sciences. One way of extracting knowledge from data can be achieved by discovering functional dependencies (FDs). FD explores the relation between different attributes, so that the value of one or more attributes is determined by another attribute set [1]. FD discovery helps in many applications, such as; query optimization, data normalization, interface restructuring, and data cleaning. A plethora of functional dependency discovery algorithms has been proposed. Some of the most widely used algorithms are; TANE [2], FD_ MINE [3], FUN [4], DFD [5], DEP-MINER [6], FASTFDS [7] and FDEP [8]. These algorithms extract FDs using different techniques, such as; (1) building a search space of all attributes combinations in an ordered manner, then start searching for candidate attributes that are assumed to have functional dependency between them, (2) generating agreeing and difference sets, where the agreeing sets are acquired through applying cross product of all tuples, the difference sets are the complement of the agreeing sets, both sets are used to infer the dependencies, (3) generating one generic set of functional dependency, in which each attribute can determine all other attributes, this set is then updated and some dependencies are removed to include more specialized dependencies through records pairwise comparisons.

Huge efforts have been dedicated to compare the most widely used algorithms in terms of runtime and memory consumption. No attention has been paid to the accuracy of resultant set of functional dependencies represented. Functional dependency accuracy is defined by two main factors; being complete and minimal.

In this paper, we are proposing a conceptual-based functional dependency detection framework. The proposed method is mainly based on Formal Concept Analysis (FCA); which is a mathematical framework rooted in lattice theory and is used for conceptual data analysis where data is represented in the form of a binary relation called a formal context [9]. From this formal context, a set of implications is extracted, these implications are in the same form of FDs. Implications are proven to be semantically equivalent to the set of all functional dependencies available

Cite this article as: Islam F. (2016). Conceptual-based Functional Dependency Detection Framework. Qatar Foundation Annual Research Conference Proceedings 2016: ICTPP3084 http://dx.doi.org/10.5339/qfarc.2016.ICTPP3084.

in the certain database [10]. This set of implications should be the smallest set representing the formal context which is termed the Duquenne–Guigues, or canonical, basis of implications [11]. Moreover, completeness of implications is achieved through applying Armstrong rules discussed in [12].

The proposed framework is composed of three main components; they are:

1. Data transformation component: it converts input data to binary formal context.
2. Reduction component: it applies data reduction on tuples or attributes.
3. Implication extraction component: this is responsible for producing minimal and complete set of implications.

The key benefits of the proposed framework:

1. It works on any kind of input data (qualitative and quantitative) that is automatically transformed to a formal context of binary relation,
2. A crisp Lukasiewicz data reduction technique is implemented to remove redundant data, which positively helps reducing the total runtime,
3. The set of implications produced are guaranteed to be minimal; due to the use of Duquenne–Guigues algorithm in extraction,
4. The set of implications produced are guaranteed to be complete; due to the use of Armstrong rules.

The proposed framework is compared to the seven most commonly used algorithms listed above and evaluated based on runtime, memory consumption and accuracy using benchmark datasets.

Acknowledgement

This contribution was made possible by NPRP-07-794-1-145 grant from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.