**Washington State University  
CPT\_S 415 – Big Data Online**

Srinivasulu Badri

**Assignment 1**

Name: Nam Jun Lee

Student Number: 11606459

1. [**Big Data concept**]Give one example of Big Data application you know. Use the detailed example to explain each of the five Big V’s.

The big data application I know is Facebook. So let me explain how 5V’s of big data are applied on Facebook.

Five Big V's:

1. Volume: Facebook has a huge number of users. According to Google, monthly active users are 2.934 billion. As such, the amount of data Facebook has is enormous.
2. Velocity: Facebook speeds up scrolling posts and playing videos, so users don't feel uncomfortable using the service
3. Variety: Facebook contains text such as users' personal information and messages. There are also pictures and countless videos. As such, Facebook includes various data source types.
4. Veracity: Facebook contains the quality and accuracy of its data. They remove dirty data for the smooth operation of the service so that users do not feel unpleasant emotions while using the service.
5. Value: Facebook includes a 'Like' feature. Through the information obtained by this function, the product advertisement for each user's area of ​​interest is displayed intensively, thereby arousing the desire of users to purchase. As such, it helps to create business value by analyzing users' interests.
6. [**Relational Data Model**]As of January 2017, the OpenFlights Airports Database (https://openflights.org/data.html) contains over 10,000 airports, train stations and ferry terminals spanning the globe. Each entry in the Airport table contains the following:
   1. Consider the following terms: *relation schema, relational database schema, domain, attribute, attribute domain, relation instance.* Give what these terms are with the above Airport database. Give one small (4-5 tuples) instance of the Airport table.

**Airport**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Airport ID | Name | City | Country | IATA | ICAO | Latitude | Longitude | Altitude | Time  zone | DST | Tz database time zone | Type | Source |
| 1 | Goroka Airport | Goroka | Papua New Guinea | GKA | AYGA | -6.0816  89834590001 | 145.391998291 | 5282 | 10 | U | Pacific/Port\_Moresby | airport | OurAirports |
| 2 | Madang Airport | Madang | Papua New Guinea | MAG | AYMD | -5.2070  7988739 | 145.789001465 | 20 | 10 | U | Pacific/Port\_Moresby | airport | OurAirports |
| 3 | Mount Hagen Kagamuga Airport | Mount Hagen | Papua New Guinea | HGU | AYMH | -5.826  789855957031 | 144.29600524902344 | 5388 | 10 | U | Pacific/Port\_Moresby | airport | OurAirports |
| 4 | Nadzab Airport | Nadzab | Papua New Guinea | LAE | AYNZ | -6.569  803 | 146.725977 | 239 | 10 | U | Pacific/Port\_Moresby | airport | OurAirports |

**Relation schema**: Airport(AirportID, Name, City, Country, IATA, ICAO, Latitude, Longitude,

Altitude, Time zone, Tz database time zone, Type, Source)

**Relational database schema**: Database schema is the collection of relation schema. Hence, one relational database schema.

**Domain:** A data type with optional constraints. For example, the domain of Name, City, and Country in the Airport relation should be a string. It cannot have numeric values.

**Attribute:** Column Headers. Hence, Airport table has 14 attributes: AirportID, Name, City, Country, IATA, ICAO, Latitude, Longitude, Altitude, Time zone, Tz database time zone, Type, Source.

**Attribute domain**: All possible values in a Column. For example, the domain of IATA in the Airport relation is 3-letter IATA code.

**Relation instance**: A relation instance is a tuple/row in a relation. Hence, the Airport table has one particular combination of attribute values.

* 1. There are three databases in the OpenFlight dataset: Airport, Airline, and Route. Give the schema of these three databases and mark the primary keys, foreign keys and provide examples of functional dependencies you identified over the three tables. [You may draw a diagram to illustrate the schema, PKs, FKs and FDs]

**Airport**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Airport ID | Name | City | Country | IATA | ICAO | Latitude | Longitude | Altitude | Timezone | DST | Tz database time zone | Type | Source |

**Airline**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Airline ID | Name | Alias | IATA | ICAO | Callsign | Country | Active |

**Route**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Airline | Airline ID | Source airport | Source airport ID | Destination airport | Destination airport ID | Codeshare | Stops | Equipment |

Primary Key: Each schema is underlined ( \_\_\_\_ )

Foreign Key: Directed arrow from FK to referenced schema ( )

Functional Dependency: Directed arrow from FD to related attributes ( )

Partial Dependency: Directed arrow from partial dependency to related attributes ( )

Transitive Dependency: Directed arrow from transitive dependency to related attributes ( )

1. [**Functional Dependencies**]Recall Armstrong’s axioms.

• Reflexivity rule: if Y ⊆ X then X → Y  
• Augmentation rule: if X → Y then XZ → YZ  
**•** Transitivity rule: if X → Y and Y → Z then X → Z

1. Give two examples for using Armstrong’s inference rules to induce new FDs from the set of FDs you designed in question 2 (b).
   * + 1. Augmentation rule: Airport Schema

If Airport ID Country then,

Airport ID + Name = Country + Name

* + - 1. Transitivity rule: Airline Schema

If Alias Name and Name Callsign then,

Alias Callsign

1. Prove the following inference rules also hold, using FD definition and Armstrong’s Axioms.
2. decomposition rule: **if X** → **YZ then: X** → **Y and X** → **Z**

Proof:

X → YZ ----- given above

YZ → Y ----- Reflexivity Rule

YZ → Z ----- Reflexivity Rule

X → Y ----- Transitivity Rule on X → YZ and YZ → Y

X → Z ----- Transitivity Rule on X → YZ and YZ → Z

Hence, Proved.

1. Pseudo transitivity: **if X** → **Y and YW** → **Z then: XW** → **Z**

Proof:

X → Y ----- given above

YW → Z ----- given above

XW → YW ----- Augmentation Rule on X → Y by augmenting with W

XW → Z ----- Transitivity Rule on XW → YW and YW → Z

Hence, Proved.

1. [**Normalization]** Given a relation R(A1, A2, A3, A4), with three FDs A2, A3 → A4; A3, A4 → A1; A1, A2→ A3. Provide the 3NF and BCNF form of the schema and explain why.

First, find the candidate keys.

Check FDs: are included in both attributes, but is included only in the left attribute. So, calculate the closing of the power set combination of :

---- Candidate key

---- Candidate key

lead only to superkeys, so are candidate keys.

Hence, the Set of key attributes: .

Now, provide the 3NF and BCNF form of this relation R:

* 3NF: For each FD, determine whether the left-hand side is a super key or the right-hand side is all key attributes, **this relation is in 3NF** because there is no transitive dependency here.
* BCNF: Check each FD: { is non-trivial but is not a super key. If this relation is in BCNF, each non-trivial FD must be a super key. Hence, **it violates BCNF**. It means **this relation is not in BCNF.**

.