

Assignment 5: Dependency Theory

Hien Tu - tun1

November 16, 2021

D1 Reasoning with dependencies

1.
 - Assume $AB \rightarrow C, A \rightarrow D$ and $CD \rightarrow EF$
 - Apply Augmentation on $AB \rightarrow C$ with A to derive $AB \rightarrow CA$
 - Apply Augmentation on $A \rightarrow D$ with C to derive $CA \rightarrow CD$
 - Apply Transitivity on $AB \rightarrow CA$ and $CA \rightarrow CD$ to derive $AB \rightarrow CD$
 - Apply Transitivity on $AB \rightarrow CD$ and $CD \rightarrow EF$ to derive $AB \rightarrow EF$
 - Apply Reflection on $F \subseteq EF$ to derive $EF \rightarrow F$
 - Apply Transitivity on $AB \rightarrow EF$ and $EF \rightarrow F$ to derive $AB \rightarrow F$
 - Hence, $AB \rightarrow F$

2.
 - Assume we have rows $r_1, r_2 \in I$ of instance I such that $r_1[XW] = r_2[XW]$
 - By $r_1[XW] = r_2[XW]$, we have $r_1[X] = r_2[X]$ and $r_1[W] = r_2[W]$
 - Using $X \rightarrow Y$ and $r_1[X] = r_2[X]$, we conclude that $r_1[Y] = r_2[Y]$
 - By $r_1[Y] = r_2[Y]$ and $r_1[W] = r_2[W]$, we have $r_1[YW] = r_2[YW]$
 - Using $YW \rightarrow Z$ and $r_1[YW] = r_2[YW]$, we conclude that $r_1[Z] = r_2[Z]$

- Hence, $r_1[Z] = r_2[Z]$ holds.
- 3.
- Assume we have $r_1[X]$ for every instance I_1 of R and every row $r_1 \in I_1$
 - Using $R[X] \subseteq S[Y]$ and $r_1[X]$, there exists a row in instance I_2 of S with $r_1[X] = r_2[X]$
 - Using $S[Y] \subseteq T[Z]$ and $r_2[Y]$, there exists a row in instance I_3 of T with $r_2[Y] = r_3[Z]$
 - Thus, for every instance I_1 of R and every row $r_1 \in R$, there exists a row in instance I_3 of T such that $r_1[X] = r_3[Z]$
 - Hence, $R[X] \subseteq T[Z]$
- 4.
- Assume $X \twoheadrightarrow Y$ and $XY \longrightarrow Z$
 - Apply Complementation on $X \twoheadrightarrow Y$ to derive $X \twoheadrightarrow Z$ (with Z all attributes of \mathbf{R} not in X and Y)
 - Apply Reflexivity on $Z \setminus (X \cup Y) \subseteq Z$ to derive $Z \longrightarrow Z \setminus (X \cup Y)$
 - Apply Transitivity on $XY \longrightarrow Z$ and $Z \longrightarrow Z \setminus (X \cup Y)$ to derive $XY \longrightarrow Z \setminus (X \cup Y)$
 - Since Z is all attributes of \mathbf{R} not in X and Y , $Z \cap XY = \emptyset$
 - Apply Coalensence on $X \twoheadrightarrow Z$, $XY \longrightarrow Z \setminus (X \cup Y)$, $Z \cap XY = \emptyset$ and $Z \setminus (X \cup Y) \subseteq Z$, we conclude that $X \longrightarrow Z \setminus (X \cup Y)$
 - Hence, $X \longrightarrow Z \setminus (X \cup Y)$

5. Consider the following table of the schema

person(name, number, birthdate, age)

name	number	birthdate	age
Alice	1	2001-01-01	20
Alice	2	2005-09-05	16

Since name, number are the primary keys, they determine all attributes. Thus, “name, number \rightarrow birthdate”.

We know from the lecture that birthdate determines age since people who have the same birthdate would have the same age, and so “birthdate \rightarrow age”.

Since $\{\text{birthdate}\} \subseteq \{\text{name, birthdate}\}$, by applying Reflexivity, we get “name, birthdate \rightarrow birthdate”.

Applying Transitivity on “name, birthdate \rightarrow birthdate” and “birthdate \rightarrow age”, we derive “name, birthdate \rightarrow age”

Hence, we have “name, number \rightarrow birthdate” and “name, birthdate \rightarrow age”.

However, from the table, we can see that “name \rightarrow age” does not hold.

Let $X = \text{name}$, $W = \text{number}$, $Y = \text{birthdate}$, $Z = \text{age}$, we have shown that the inference rule from the question is not sound.

6. TODO: Do Question 6

7. The attribute closure of set of attributes C :

- Initially, $\text{closure} = \{C\}$
- From $C \rightarrow A$, since $C \subseteq \text{closure}$ and $A \not\subseteq \text{closure}$, $\text{closure} = \{C, A\}$
- From $AC \rightarrow E$, since $AC \subseteq \text{closure}$ and $E \not\subseteq \text{closure}$, $\text{closure} = \{C, A, E\}$
- From $E \rightarrow B$, since $E \subseteq \text{closure}$ and $B \not\subseteq \text{closure}$, $\text{closure} = \{C, A, E, B\}$
- From $AB \rightarrow D$, since $AB \subseteq \text{closure}$ and $D \not\subseteq \text{closure}$, $\text{closure} = \{C, A, E, B, D\}$
- From $BC \rightarrow D$, since $BC \subseteq \text{closure}$ and $D \subseteq \text{closure}$, we don't need to add D into closure more (as D is already in closure)
- From $D \rightarrow A$, since $D \subseteq \text{closure}$ and $A \subseteq \text{closure}$, we don't need to add A into closure more (as A is already in closure)
- Therefore, $C^+ = \{A, B, C, D, E\}$

The attribute closure of set of attributes (EA) :

- Initially, $closure = \{E, A\}$
- From $E \rightarrow B$, since $E \subseteq closure$ and $B \not\subseteq closure$, $closure = \{E, A, B\}$
- From $AB \rightarrow D$, since $AB \subseteq closure$ and $D \not\subseteq closure$, $closure = \{E, A, B, D\}$
- From $AC \rightarrow E$, since $AC \not\subseteq closure$, we don't need to add E into $closure$
- From $BC \rightarrow D$, since $BC \not\subseteq closure$, we don't need to add D into $closure$
- From $C \rightarrow A$, since $C \not\subseteq closure$, we don't need to add A into $closure$
- From $D \rightarrow A$, since $D \subseteq closure$ and $A \subseteq closure$, we don't need to add A into $closure$ more (as A is already in $closure$)
- Therefore, $(EA)^+ = \{A, B, D, E\}$

8. Compute X^+ for every $X \subseteq \{A, B, C, D, E\}$

- $A^+ = \{A\}$ since there is no $A \rightarrow \dots$ in \mathfrak{S}
- $B^+ = \{B\}$ since there is no $B \rightarrow \dots$ in \mathfrak{S}
- $C^+ = \{A, B, C, D, E\}$ from question 7
- $D^+ = \{A, D\}$ since there is only $D \rightarrow A$ that would satisfy the closure algorithm
- $E^+ = \{B, E\}$ since there is only $E \rightarrow B$ that would satisfy the closure algorithm
- $(AB)^+ = \{A, B, D\}$ since there is only $AB \rightarrow D$ that would satisfy the closure algorithm
- $(AC)^+ = \{A, B, C, D, E\}$ since AC includes C and we can reach others from only C
- $(AD)^+ = \{A, D\}$ since there is no dependency that would satisfy the closure algorithm

- $(AE)^+ = \{A, B, E\}$ since there is only $E \rightarrow B$ that would satisfy the closure algorithm
- $(BC)^+ = \{A, B, C, D, E\}$ since BC includes C and we can reach others from only C
- $(BD)^+ = \{A, B, D\}$ since there is only $D \rightarrow A$ that would satisfy the closure algorithm
- $(BE)^+ = \{B, E\}$ since there is no dependency that would satisfy the closure algorithm
- $(CD)^+ = \{A, B, C, D, E\}$ since CD includes C and we can reach others from only C
- $(CE)^+ = \{A, B, C, D, E\}$ since CE includes C and we can reach others from only C
- $(DE)^+ = \{A, B, D, E\}$ there are $D \rightarrow A$, $E \rightarrow B$ satisfy the closure algorithm
- $(ABC)^+ = \{A, B, C, D, E\}$ since ABC includes C
- $(ABD)^+ = \{A, B, D\}$ since there is no dependency that would satisfy the closure algorithm
- $(ABE)^+ = \{A, B, D, E\}$ since there is only $AB \rightarrow D$ that would satisfy the closure algorithm
- $(ACD)^+ = \{A, B, C, D, E\}$ since ACD includes C
- $(ACE)^+ = \{A, B, C, D, E\}$ since ACE includes C
- $(ADE)^+ = \{A, B, D, E\}$ since there is only $E \rightarrow B$ that would satisfy the closure algorithm
- $(BCD)^+ = \{A, B, C, D, E\}$ since BCD includes C
- $(BCE)^+ = \{A, B, C, D, E\}$ since BCE includes C
- $(BDE)^+ = \{A, B, D, E\}$ since there is only $D \rightarrow A$ that would satisfy the closure algorithm
- $(CDE)^+ = \{A, B, C, D, E\}$ since CDE includes C
- $(ABCD)^+ = \{A, B, C, D, E\}$ since $ABCD$ includes C
- $(ABCE)^+ = \{A, B, C, D, E\}$ since $ABCE$ includes C
- $(ABDE)^+ = \{A, B, D, E\}$ since there is no dependency that would satisfy the closure algorithm

- $(ACDE)^+ = \{A, B, C, D, E\}$ since $ACDE$ includes C
- $(BCDE)^+ = \{A, B, C, D, E\}$ since $BCDE$ includes C
- $(ABCDE)^+ = \{A, B, C, D, E\}$ since $ABCDE$ includes C
- Any combination that contains C is the superkey
- C is the (candidate) key

- 9.
- Starting with $\{AB \rightarrow D, AC \rightarrow E, BC \rightarrow D, C \rightarrow A, D \rightarrow A, E \rightarrow B\}$
 - From $C \rightarrow A$, we can use Augmentation with C to derive $C \rightarrow AC$. We also have $AC \rightarrow E$. Then, by Transitivity, we conclude $C \rightarrow E$. Thus, we can add $C \rightarrow E$ to our set.
 $\{AB \rightarrow D, AC \rightarrow E, BC \rightarrow D, C \rightarrow A, D \rightarrow A, E \rightarrow B, C \rightarrow E\}$
 - From $C \rightarrow E$, we can use Augmentation with A to derive $AC \rightarrow AE$. Then, we can use Decomposition to get $AC \rightarrow A$ and $AC \rightarrow E$. $AC \rightarrow A$ is trivial since $A \subseteq AC$. This means that, from $C \rightarrow E$, we can get $AC \rightarrow E$. So, we can get rid of $AC \rightarrow E$.
 $\{AB \rightarrow D, BC \rightarrow D, C \rightarrow A, D \rightarrow A, E \rightarrow B, C \rightarrow E\}$
 - From $C \rightarrow E$ and $E \rightarrow B$, by Transitivity, we get $C \rightarrow B$. Then, we get apply Augmentation on $C \rightarrow B$ with C to derive $C \rightarrow BC$. We also have $BC \rightarrow D$. So, by Transitivity, we derive $C \rightarrow D$. Thus, we can add $C \rightarrow D$ to our set.
 $\{AB \rightarrow D, BC \rightarrow D, C \rightarrow A, D \rightarrow A, E \rightarrow B, C \rightarrow E, C \rightarrow D\}$
 - From $C \rightarrow D$, we can apply Augmentation with B to get $BC \rightarrow BD$. Then, we can use Decomposition to get $BC \rightarrow B$ and $BC \rightarrow D$. $BC \rightarrow B$ is trivial since $B \subseteq BC$. This means that from $C \rightarrow D$, we can get $BC \rightarrow D$. So, we can get rid of $BC \rightarrow D$.
 $\{AB \rightarrow D, C \rightarrow A, D \rightarrow A, E \rightarrow B, C \rightarrow E, C \rightarrow D\}$
 - By Transitivity on $C \rightarrow D$ and $D \rightarrow A$, we can derive $C \rightarrow A$. Thus, we can get rid of $C \rightarrow A$.
 $\{AB \rightarrow D, D \rightarrow A, E \rightarrow B, C \rightarrow E, C \rightarrow D\}$