

1906003022015

Veritabanı Yönetim Sistemleri

BAİBÜ Bilgisayar Müh.

Dr. Öğr. Üyesi İsmail Hakkı Parlak

ismail.parlak@ibu.edu.tr

Oda: 335

Ayrıştırma ile Tasarım

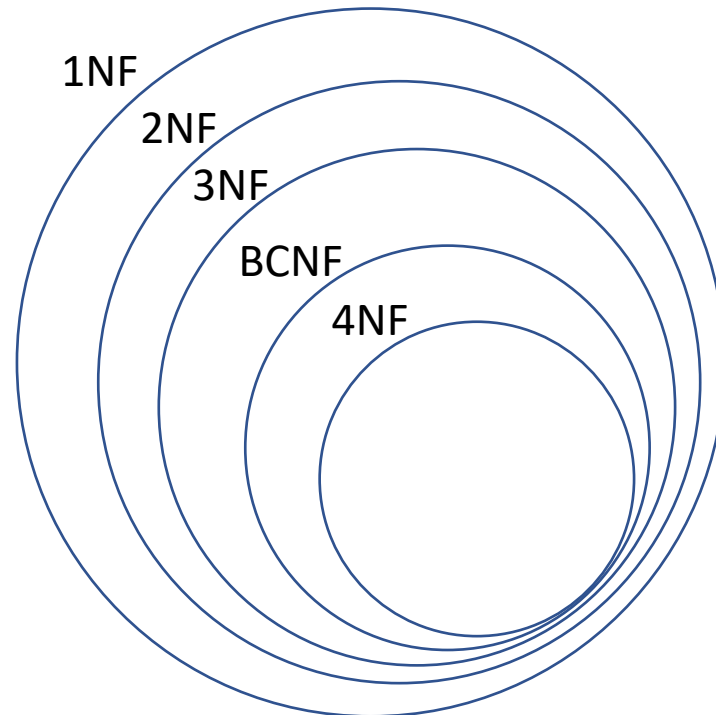
- Her niteliği barındıran "dev" ilişkiler ile başlıyoruz.
- Dev ilişkiyi, aynı bilgileri barındırabileceğimiz daha küçük ilişkilere doğru ayrıştırıyoruz.
- Ayrıştırma işlemini otomatik gerçekleştirebiliriz:
 - Dev ilişkiler + veriye ait özellikler belirlenir.
 - Sistem veriye ait özellikler kullanılarak ayrıştırılır.
 - En son elde edilen ilişkiler ***normal formları*** sağlar.
 - Bu sayede anomali oluşma ihtimali ortadan kalkar ve veri kaybı yaşanmaz.

Veri Özellikleri ve Normal Formlar

Fonksiyonel bağılıklar → Boyce-Codd Normal Form

+

Birden çok değerli bağılıklar → 4. Normal Form



Örnek

- Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)
- Başvurular(TCNo, üniAdı, üniŞehri, tarih, anaDal)
- "öncelik ortalamaya göre belirlenir."
 - $\text{ortalama} > 4.5$ ise öncelik = 1
 - $4 < \text{ortalama} \leq 4.5$ ise öncelik = 2
 - $\text{ortalama} \leq 4$ ise öncelik = 3
- Aynı ortalamaya sahip olan 2 öğrenci aynı önceliğe sahip olur.

Örnek

- Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)
- Aynı ortalamaya sahip olan 2 öğrenci aynı önceliğe sahip olur.
- $\forall t, u \in \text{Öğrenciler}$:
 $t.ortalama = u.ortalama \rightarrow t.öncelik = u.öncelik$
- ortalama \rightarrow öncelik

Fonksiyonel Bağımlılık (Functional Dependency)

- $\forall t, u \in R: t.A = u.A \rightarrow t.B = u.B$
- $A \rightarrow B$

FB Genel Notasyonu

- $\forall t, u \in R: t[A_1, A_2, \dots, A_n] = u[A_1, A_2, \dots, A_n] \rightarrow t[B_1, B_2, \dots, B_m] = u[B_1, B_2, \dots, B_m]$
- $A_1, A_2, \dots, A_n \rightarrow B_1, B_2, \dots, B_m$

Kısaltma

- $A_1, A_2, \dots, A_n = \bar{A}$
- $B_1, B_2, \dots, B_m = \bar{B}$

Fonksiyonel Bağımlılık (Functional Dependency)

- $\bar{A} \rightarrow \bar{B}$
- $R(\bar{A}, \bar{B}, \bar{C})$

\bar{A}	\bar{B}	\bar{C}
\bar{a}	\bar{b}	\bar{c}_1
...
\bar{a}	\bar{b}	\bar{c}_2
...

Örnek

- Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)
- FB'ler neler olabilir?

FB (FD) ve Anahtar (Key) Kavramı

1. Kopyası olmayan (tekrarsız) kayıtlardan oluşan bir R olsun.
2. R içinde $\bar{A} \rightarrow \text{Tüm nitelikler}$ olsun.

Bu 2 şart sağlanıyorsa \bar{A} bir anahtardır (key).

FD Türleri

- Önemsiz (trivial) FB:

$$\bar{A} \rightarrow \bar{B} \text{ ve } \bar{B} \subseteq \bar{A}$$

- Önemsiz olmayan (nontrivial) FB:

$$\bar{A} \rightarrow \bar{B} \text{ ve } \bar{B} \subsetneq \bar{A}$$

- Tamamen önemsiz olmayan (completely nontrivial) FB:

$$\bar{A} \rightarrow \bar{B} \text{ ve } \bar{B} \cap \bar{A} = \emptyset$$

FD Kuralları

- Ayırma (splitting) kuralı:

$\bar{A} \rightarrow B_1, B_2, \dots, B_m$ ise:

$$\bar{A} \rightarrow B_1$$

$$\bar{A} \rightarrow B_2$$

...

$$\bar{A} \rightarrow B_m$$

- Sol tarafı parçalayabilir miyiz?

$A_1, A_2, \dots, A_n \rightarrow B$ ise: $A_1 \rightarrow B???$

FD Kuralları

- Birleştirme (combining) kuralı:

$$\left. \begin{array}{l} \bar{A} \rightarrow B_1 \\ \bar{A} \rightarrow B_2 \\ \dots \\ \bar{A} \rightarrow B_m \end{array} \right\} \bar{A} \rightarrow B_1, B_2, \dots, B_m$$

FD Kuralları

- Önemsiz (trivial) FD kuralları:
- $\bar{A} \rightarrow \bar{B}$ ve $\bar{B} \subseteq \bar{A}$

1. $\bar{A} \rightarrow \bar{B}$ ise $\bar{A} \rightarrow \bar{A} \cup \bar{B}$

2. $\bar{A} \rightarrow \bar{B}$ ise $\bar{A} \rightarrow \bar{A} \cap \bar{B}$

FD Kuralları

- Geçişlilik (transitive) kuralı:
- $\bar{A} \rightarrow \bar{B}$ ve $\bar{B} \rightarrow \bar{C}$ ise $\bar{A} \rightarrow \bar{C}$

\bar{A}	\bar{B}	\bar{C}	\bar{D}

Nitelik Kapanımı (Closure of Attributes)

- Bir ilişki için FB'ler ve bir nitelik kümesi \bar{A} verilmiş olsun.
- $\bar{A} \rightarrow B$ 'yi sağlayan tüm B 'lerin kümesi \bar{A} 'nın kapanımını oluşturur.
- \bar{A} 'nın kapanımı \bar{A}^+ sembolü ile gösterilir.

Nitelik Kapanımı Örnek

- Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)
- TCNo \rightarrow ad, adres, ortalama
- ortalama \rightarrow öncelik
- liseKodu \rightarrow liseAdı, liseŞehri
- {TCNo, liseKodu}⁺ = ?

Kapanım ve Anahtar Kavramı

- \bar{A} , R için bir anahtar mıdır?
 - Eğer $\bar{A}^+ = R$ 'nin tüm nitelikleri ise \bar{A} anahtardır.
- R üzerinde tanımlı bir FD kümesi verildiğinde tüm anahtarlar nasıl bulunabilir?
 - Tüm nitelik alt kümelerinin kapanımlarının tek tek tüm nitelikleri kapsayıp kapsamadığı test edilebilir. Ancak çok verimli bir yöntem değildir.
 - Daha verimli çalışmak için alt küme boyutları küçükten büyüğe test edilebilir. Eğer $AB \rightarrow$ tüm nitelikler ise AB 'nin tüm süper setleri \rightarrow tüm nitelikler olacaktır.

Takipçi FB Kavramı

- S1 ve S2 iki FB kümesi olsun.
- S1'i sağlayan bütün ilişkiler S2'yi de sağlıyorsa, S2, S1'in takipçisidir denir.
- Ör:
 - S2: {TCNo \rightarrow öncelik}
 - S1: {TCNo \rightarrow ortalama, ortalama \rightarrow öncelik}
- Takipçilik testi: $\bar{A} \rightarrow \bar{B}$ FB'si S'in takipçisi midir?
 - S üzerinde \bar{A}^+ hesaplanır, eğer \bar{B} hesaplanan \bar{A}^+ içindeyse $\bar{A} \rightarrow \bar{B}$, S'in takipçisidir.

İlişkisel Şemayı Ayırıştırmak

- $R(A_1, A_2, \dots, A_n)$

Ayrıştırma Örneği 1

Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)

S1(TCNo, ad, adres, liseKodu, ortalama, öncelik)

S2(liseKodu, liseAdı, liseŞehri)

S1'in nitelikleri \cup S2'nin nitelikleri = Öğrenciler'in nitelikleri

$S1 \bowtie S2 = \text{Öğrenciler}$

Ayrıştırma Örneği 2

Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)

S1(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri)

S2(ad, liseAdı, ortalama, öncelik)

S1'in nitelikleri \cup S2'nin nitelikleri = Öğrenciler'in nitelikleri

S1 \bowtie S2 = Öğrenciler ???

Boyce-Codd Normal Form (BCNF)

- Dev ilişkileri ayırıştırma doğru yapılmalıdır (oluşan parçalar birleştirilince dev ilişki kayıpsız olarak tekrar oluşturulabilmeli).
- Ayırıştırma sonucu oluşan ilişkiler BCNF'ye uygun olmalıdır.
- **BCNF:** FB'leri bulunan bir R, BCNF'i şu şartlar altında sağlar:
 - Tüm $\bar{A} \rightarrow B$ için \bar{A} anahtar olmalıdır.

\bar{A}	B	XYZ

} BCNF ihlali örneği

BCNF Örneği 1

- Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)
- TCNo \rightarrow ad, adres, ortalama
- ortalama \rightarrow öncelik
- liseKodu \rightarrow liseAdı, liseŞehri
- Anahtarlar = {TCNo, liseKodu}
- Tüm FB'lerin sol tarafında anahtar bulunuyor mu?

BCNF Örneği 2

- Başvurular(TCNo, üniAdı, üniŞehri, tarih, anaDal)
- TCNo, üniAdı, üniŞehri \rightarrow tarih, anaDal (her öğrenci her üniversiteye sadece 1 kere ve 1 ana dala başvurabilir)
- Anahtarlar =
- Tüm FB'lerin sol tarafında anahtar bulunuyor mu?

BCNF Ayırıştırma Algoritması

- Girdi: $R + R'$ 'deki FB'ler.
 - Çıktı: R 'nin geri kayıpsız birleştirilebilen BCNF'e uyan ayrışimleri.
1. R 'nin anahtarlarını bul.
 2. Tüm ayrışım ilişkileri BCNF'e uyana kadar tekrar et:
 - $A \rightarrow B$ 'ye sahip BCNF'i ihlal eden bir R' seç.
 - R' ilişkisini $R_1(A, B)$ ve $R_2(A, \text{geriye kalanlar})$ diye ayırıştır.
 - R_1 ve R_2 için FB'leri hesapla.
 - R_1 ve R_2 'nin anahtarlarını hesapla.

BCNF Ayırıştırma Örneği

- Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)
- TCNo \rightarrow ad, adres, ortalama
- ortalama \rightarrow öncelik
- liseKodu \rightarrow liseAdı, liseŞehri
- Anahtar = {TCNo, liseKodu}

BCNF Ayırıştırma Örneği

- Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)
- TCNo \rightarrow ad, adres, ortalama
- ortalama \rightarrow öncelik
- liseKodu \rightarrow liseAdı, liseŞehri
- Anahtar = {TCNo, liseKodu}

BCNF Ayırıştırma Örneği

- Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)
- TCNo \rightarrow ad, adres, ortalama
- ortalama \rightarrow öncelik
- liseKodu \rightarrow liseAdı, liseŞehri
- Anahtar = {TCNo, liseKodu}

BCNF Ayırıştırma Örneği

- Öğrenciler(TCNo, ad, adres, liseKodu, liseAdı, liseŞehri, ortalama, öncelik)
- TCNo \rightarrow ad, adres, ortalama
- ortalama \rightarrow öncelik
- liseKodu \rightarrow liseAdı, liseŞehri
- Anahtar = {TCNo, liseKodu}