Web Madenciliği
(Web Mining)

Hazırlayan: M. Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü

Bu dersin sunumları, "Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data, Bing Liu, Springer, 2011." kitabı kullanılarak hazırlanmıştır.

Konular

- Sıralı Örüntülerin Temelleri
- GSP Tabanlı Sıralı Örüntü Madenciliği
  - Algoritma
- Sıralı Örüntülerden Kural Oluşturulması
  - Sıralı Kurallar
  - Etiket Sıralı Kurallar
  - Sınıf Sıralı Kurallar

Sıralı Örüntülerin Temelleri

- Birliktelik kural madenciliğinde transaction'ların sırası gözönüne alınmaz.
- Bazı uygulamalarda item'ların sırası da önemlidir.
- Kullanıcıların Web sayfalarındaki ziyaret sırası Web kullanım madenciliğinde faydalıdır.
- Metin madenciliğinde cümle içerisindeki kelimelerin sırası dilsel örüntülerin elde edilmesi için önemlidir.
- Bir müşteri üç ay içerisinde, önce bilgisayar, sonra CD-ROM, ardından video kamera alıyor.
- Sağlıkla ilgili verilerde, deprem gibi doğal afetlerde, telefon çağrı örüntülerinin analizinde, DNA sıra analizi ve gen analizinde kullanılır.

Sıralı Örüntülerin Temelleri

- $I = \{i_1, i_2, \ldots, i_m\}$ item'lar kümesidir. Bir **sequence** X ise, itemset'lerin sıralı listesidir ($X \subseteq I$).
- Bir sequence $s=\langle a_1 a_2 \ldots a_r \rangle$ şeklinde gösterilir. Burada, a_i **element** olarak adlandırılır ve bir itemset'tir.
- Bir $a_i = \{x_1, x_2, \ldots, x_k\}$ olarak gösterilir. Burada, $x_j \in I$ dir.

Sıralı Örüntülerin Temelleri

- Bir sequence içerisindeki element'teki item'lar **lexicographic (sözlüksel)** sıralıdır.
- Bir item bir element içerisinde sadece bir kez bulunur. Ancak, farklı element'lerde birden fazla bulunabilir.
- Bir sequence için **size (boyut)** içindeki **element sayısı** ile ifade edilir. **Length (uzunluk)** ise içerisindeki **item sayısı** ile ifade edilir.
- Uzunluğu k olan sequence ise **k-sequence** olarak gösterilir (Farklı element içinde aynı item'ların tekrarı uzunluğu etkiler).
- Eğer, $s_1=\langle a_1 a_2 \ldots a_r \rangle$ ve $s_2=\langle b_1 b_2 \ldots b_v \rangle$ şeklinde iki sequence için $1 \leq j_1 < j_2 < \ldots < j_{r-1} < j_r \leq v$ şeklindeki sayılar için $a_1 \subseteq b_{j1}$, $a_2 \subseteq b_{j2}$, …, $a_r \subseteq b_{jr}$ olursa, s_2 **sequence'i** s_1 **sequence'ini kapsar** (s_2 supersequence, s_1 subsequence).

5

Sıralı Örüntülerin Temelleri

Örnek

- I = {1, 2, 3, 4, 5, 6, 7, 8, 9}.
- ⟨{3} {4, 5} {8}⟩ sequence'i ⟨{6} {3, 7} {9} {4, 5, 8} {3, 8}⟩ sequence'inin **subsequence'idir.**
- {3} ⊆ {3, 7}, {4, 5} ⊆ {4, 5, 8}, ve {8} ⊆ {3, 8}.
- Ancak, ⟨{3}{8}⟩ sequence'i, ⟨{3, 8}⟩ sequence'i tarafından kapsanmaz.
- ⟨{3}{4, 5}{8}⟩ sequence'inin size'ı 3, ve length'i ise 4'tür.

6

Sıralı Örüntülerin Temelleri

Örnek

- Örnekte minsup = 2 transaction için **<{ab}{c}>** örnek bir sıralı örüntüdür.

Müşteri No	sequence
10	<{a}{abc}{ac}{d}{cf}>
20	<{ad}{c}{bc}{ae}>
30	<{ef}{ab}{df}{cb}>
40	<{eg}{af}{cbc}>

Sıralı Örüntülerin Temelleri

Örnek

- Aşağıdaki tabloda bir market sepeti transaction'ları verilmiştir.

Customer ID	Transaction Time	Transaction (items bought)
1	July 20, 2005	30
1	July 25, 2005	90
2	July 9, 2005	10, 20
2	July 14, 2005	30
2	July 20, 2005	10, 40, 60, 70
3	July 25, 2005	30, 50, 70, 80
4	July 25, 2005	30
4	July 29, 2005	30, 40, 70, 80
4	August 2, 2005	90
5	July 12, 2005	90

Sıralı Örüntülerin Temelleri

Örnek

- Aşağıda müşteri sıralaması görülmektedir.

Customer ID	Data Sequence
1	⟨{30} {90}⟩
2	⟨{10, 20} {30} {10, 40, 60, 70}⟩
3	⟨{30, 50, 70, 80}⟩
4	⟨{30} {30, 40, 70, 80} {90}⟩
5	⟨{90}⟩

- Aşağıda minsup = %25 (en az 2 müşteri) transaction için sıralı örüntüler görülmektedir.

	Sequential Patterns with Support ≥ 25%
1-sequences	⟨{30}⟩, ⟨{40}⟩, ⟨{70}⟩, ⟨{80}⟩, ⟨{90}⟩
2-sequences	⟨{30} {40}⟩, ⟨{30} {70}⟩, ⟨{30}, {90}⟩, ⟨{30, 70}⟩, ⟨{30, 80}⟩, ⟨{40, 70}⟩, ⟨{70, 80}⟩
3-sequences	⟨{30} {40, 70}⟩, ⟨{30, 70, 80}⟩

9

Konular

- Sıralı Örüntülerin Temelleri
- GSP Tabanlı Sıralı Örüntü Madenciliği
  - Algoritma
- Sıralı Örüntülerden Kural Oluşturulması
  - Sıralı Kurallar
  - Etiket Sıralı Kurallar
  - Sınıf Sıralı Kurallar

10

GSP Tabanlı Sıralı Örüntü Madenciliği

- GSP (Generalized Sequential Pattern) algoritması Apriori algoritmasıyla aynı şekilde çalışır.
- F_k frequent k-sequence'leri, C_k ise tüm aday k-sequence'leri tutar.
- Temel farklılık aday oluşturma fonksiyonu olan **candidate-gen-SPM()** fonksiyonudur. (SPM-Sequential Pattern Mining)
- İki sequence'in birleşiminde yeni bir aday sequence elde edilir. **Ancak ardarda eklenebilir özellikte olmaları gereklidir.**

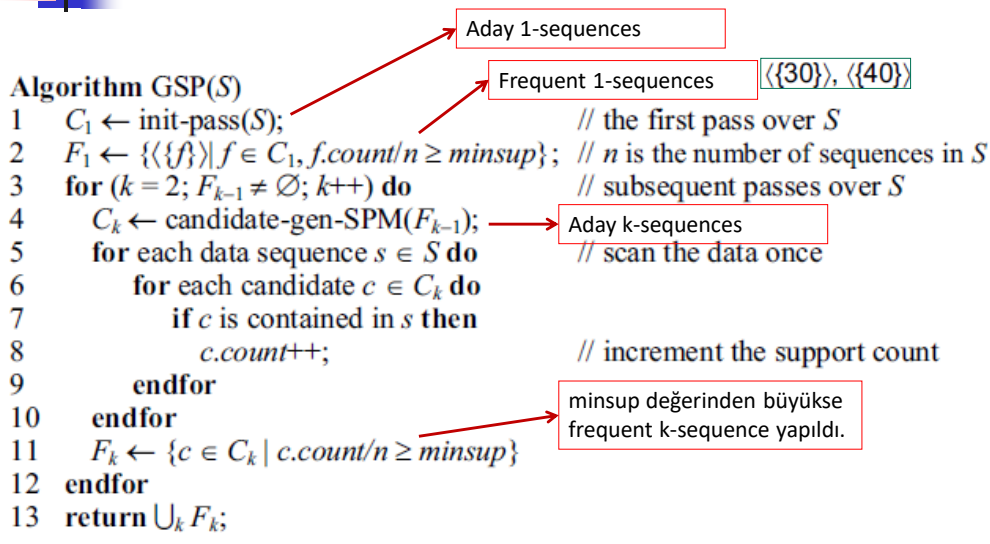
11

Konular

- Sıralı Örüntülerin Temelleri
- GSP Tabanlı Sıralı Örüntü Madenciliği
  - Algoritma
- Sıralı Örüntülerden Kural Oluşturulması
  - Sıralı Kurallar
  - Etiket Sıralı Kurallar
  - Sınıf Sıralı Kurallar

12

Algoritma



Algorithm GSP(S)
1 $C_1 \leftarrow$ init-pass(S); // the first pass over S
2 $F_1 \leftarrow \{\langle\{f\}\rangle | f \in C_1, f.count/n \geq minsup\}$; // n is the number of sequences in S
3 **for** ($k = 2$; $F_{k-1} \neq \emptyset$; k++) **do** // subsequent passes over S
4 $C_k \leftarrow$ candidate-gen-SPM(F_{k-1});
5 **for** each data sequence $s \in S$ **do** // scan the data once
6 **for** each candidate $c \in C_k$ **do**
7 **if** c is contained in s **then**
8 $c.count$++; // increment the support count
9 **endfor**
10 **endfor**
11 $F_k \leftarrow \{c \in C_k \mid c.count/n \geq minsup\}$
12 **endfor**
13 **return** $\bigcup_k F_k$;

Algoritma

Function candidate-gen-SPM(F_{k-1}) // SPM: Sequential Pattern Mining

1. Join adımı: Aday sequence'ler F_{k-1} (**s_1**) ile F_{k-1} (**s_2**) birleştirilerek yeni sequence elde edilir.

Eğer **s_1 sequence'inin ilk item'ını çıkardıktan sonra kalan sequence** ile, **s_2 sequence'inin son item'ını çıkardıktan sonra kalan sequence aynı ise**, s_1 sequence'ine s_2 sequence'i eklenir (s_1={1, 2, 3}, s_2={2, 3, 4}). s_{yeni}={1, 2, 3, 4}

- Eğer eklenen item **ayrı bir element ise** yeni sequence'in sonuna element olarak eklenir.

- Diğer durumlarda **s_1 sequence'inin** son elementine **s_2 sequence'inin** son item'ı eklenir.

2. Prune adımı: Eğer aday sequence'in (k-1) subsequence'lerinden birisi frequent değilse aday kümeden silinir.

Algoritma

- Aşağıdaki tabloda F_3, C_4 ve prune adımından sonra elde edilen sequence'ler görülmektedir.

Frequent 3-sequences	Candidate 4-sequences	
	after joining	after pruning
⟨{1, 2} {4}⟩	⟨{1, 2} {4, 5}⟩	⟨{1, 2} {4, 5}⟩
⟨{1, 2} {5}⟩	⟨{1, 2} {4} {6}⟩	
⟨{1} {4, 5}⟩		
⟨{1, 4} {6}⟩		
⟨{2} {4, 5}⟩		
⟨{2} {4} {6}⟩		

- ⟨{1, 2}{4}⟩ ile ⟨{2}{4, 5}⟩ birleştirilir ve ⟨{1,2}{4, 5}⟩ oluşur.
- ⟨{1, 2}{4}⟩ ile ⟨{2}{4}{6}⟩ birleştirilir ve ⟨{1,2}{4}{6}⟩ oluşur.
- **⟨{1}{4, 5}⟩ birleştirilemez.** ⟨{4, 5}{x}⟩ veya ⟨{4, 5, x}⟩ gereklidir.
- Pruning adımında ⟨{1, 2}{4}{6}⟩ silinir. Çünkü ⟨{1}{4}{6}⟩, ⟨{1, 2}{6}⟩ F_3'te yoktur.

15

Konular

- Sıralı Örüntülerin Temelleri
- GSP Tabanlı Sıralı Örüntü Madenciliği
  - Algoritma
- Sıralı Örüntülerden Kural Oluşturulması
  - Sıralı Kurallar
  - Etiket Sıralı Kurallar
  - Sınıf Sıralı Kurallar

16

Sıralı Örüntülerden Kural Oluşturulması

- Elde edilen sıralı örüntülerden, **sıralı kurallar, etiket sıralı kurallar** ve **sınıf sıralı kurallar** oluşturulabilir.
- Elde edilen kurallar özellikle Web kullanım madenciliğinde ve Web içerik madenciliğinde kullanılır.
- Web kullanım madenciliğinde **clickstream örüntülerinin elde edilmesi amacıyla kullanılır.**
- Elde edilen örüntüler, kullanıcıların sık eriştikleri sayfaların, nesnelerin veya kaynakların gösteriminde kullanılır.
- Web içerik madeciliğinde Web sayfalarındaki bilgilerin elde edilmesi amaçlanır.
- İçeriğe göre **kategori oluşturma**, **öbekleme** ve **sorguyla ilişki düzeyi belirleme** yapılabilir.

17

Konular

- Sıralı Örüntülerin Temelleri
- GSP Tabanlı Sıralı Örüntü Madenciliği
  - Algoritma
- Sıralı Örüntülerden Kural Oluşturulması
  - Sıralı Kurallar
  - Etiket Sıralı Kurallar
  - Sınıf Sıralı Kurallar

18

Sıralı Kurallar

- **Bir sıralı kural $X \rightarrow Y$ şeklinde ifade edilir.**
- Burada, **Y bir sequence'dir** ve **X ise Y'nin subsequence'dir ($X \subseteq Y$).**
- Y sequence'inin length'i X'in length'inden büyüktür.
- $X \rightarrow Y$ şeklindeki bir sıralı kuralın **support** değeri, bir sıralı veritabanı **S içerisinde Y'nin bulunma oranıdır.**
- $X \rightarrow Y$ şeklindeki bir sıralı kuralın **confidence** değeri, bir sıralı veritabanı **S içerisinde Y'nin X ile birlikte bulunma oranıdır.**

19

Sıralı Kurallar

Örnek

- Aşağıdaki tabloda minsup = %30 (>=2/5) ve minconf = %60 olsun.

	Data Sequence
1	⟨{1}{3}{5}{7, 8, 9}⟩
2	⟨{1}{3}{6}{7, 8}⟩
3	⟨{1, 6}{7}⟩
4	⟨{1}{3}{5, 6}⟩
5	⟨{1}{3}{4}⟩

- Bir sıralı kural aşağıdaki gibi olabilir:

⟨{1}{7}⟩ → ⟨{1}{3}{7, 8}⟩ [sup = 2/5, conf = 2/3]

- 1. ve 2. sequence'lerde ⟨{1}{3}{7,8}⟩ vardır (support için).
- 1., 2. ve 3. sequence'lerde ⟨{1}{7}⟩ vardır (confidence için).

20

Konular

- Sıralı Örüntülerin Temelleri
- GSP Tabanlı Sıralı Örüntü Madenciliği
  - Algoritma
- Sıralı Örüntülerden Kural Oluşturulması
  - Sıralı Kurallar
  - Etiket Sıralı Kurallar
  - Sınıf Sıralı Kurallar

21

Etiket Sıralı Kurallar

- **Bir etiket sıralı kural $X \rightarrow Y$ şeklinde ifade edilir.**
- Burada, Y **bir sequence'dir** ve X **ise** Y**'de bazı item'ların yerine wildcard (*) konularak oluşturulan bir sequence'dir.**
- Wildcard konulan item'ler genellikle önemlidir ve etiket olarak adlandırılır.
- Aşağıdaki tabloda minsup = %30 (>=2/5) ve minconf = %60 olsun.

	Data Sequence
1	⟨{1}{3}{5}{7, 8, 9}⟩
2	⟨{1}{3}{6}{7, 8}⟩
3	⟨{1, 6}{7}⟩
4	⟨{1}{3}{5, 6}⟩
5	⟨{1}{3}{4}⟩

22

Etiket Sıralı Kurallar

Örnek

	Data Sequence
1	⟨{1}{3}{5}{7, 8, 9}⟩
2	⟨{1}{3}{6}{7, 8}⟩
3	⟨{1, 6}{7}⟩
4	⟨{1}{3}{5, 6}⟩
5	⟨{1}{3}{4}⟩

- Bir etiket sıralı kural aşağıdaki gibi olabilir:

  ⟨{1}{*}{7, *}⟩ → ⟨{1}{3}{7, 8}⟩ [sup = 2/5, conf = 2/2].

- Burada, {*} sadece bir item'a sahip bir element'i gösterir. {1} ile veya {7, *} ile kendi arasında başka element'ler olabilir.
- {7, *} da ise 7 ile başlayan en az iki item'a sahip tüm element'ler ifade edilmektedir.
- 1. ve 2. sequence'lerde ⟨{1}{*}{7, *}⟩ vardır.
- 1. ve 2. sequence'lerde ⟨{1}{3}{7,8}⟩ vardır.
- Burada, **item 3 ve 8 etiket olarak adlandırılır.**

23

Konular

- Sıralı Örüntülerin Temelleri
- GSP Tabanlı Sıralı Örüntü Madenciliği
  - Algoritma
- Sıralı Örüntülerden Kural Oluşturulması
  - Sıralı Kurallar
  - Etiket Sıralı Kurallar
  - Sınıf Sıralı Kurallar

24

Sınıf Sıralı Kurallar

- Sınıf sıralı kurallar (class sequential rules - CSR), sınıf birliktelik kurallarına benzer.
- S bir **sequence kümesi** olsun. Buradaki **her sequence** bir y **sınıfına etiketlensin.** I kümesi S kümesindeki tüm item'ların kümesi ve Y ise tüm sınıfların kümesi olsun **($I \cap Y = \emptyset$).**
- Giriş verisi $D = \{(s_1, y_1), (s_2, y_2), \ldots, (s_n, y_n)\}$ olsun ($s_i \in S$ ve $y_i \in Y$).
- **Bir sınıf sıralı kural $X \rightarrow y$ şeklinde ifade edilir.** Burada, X bir sequence'dir ve $y \in Y$ dir.
- Eğer, X sequence'i s_i'nin subsequence'i ise, **(s_i, y_i) verisi $X \rightarrow y$ kuralını kapsar (cover).**
- Eğer, X sequence'i s_i'nin subsequence'i ve $y_i = y$ ise, **(s_i, y_i) verisi $X \rightarrow y$ kuralını karşılar (satisfy).**

25

Sınıf Sıralı Kurallar

Örnek

- Aşağıdaki tabloda minsup = %30 ve minconf = %60 olsun.

	Data Sequence	Class
1	⟨{1}{3}{5}{7, 8, 9}⟩	c_1
2	⟨{1}{3}{6}{7, 8}⟩	c_1
3	⟨{1, 6}{9}⟩	c_2
4	⟨{3}{5, 6}⟩	c_2
5	⟨{1}{3}{4}{7, 8}⟩	c_2

- Bir sınıf sıralı kural aşağıdaki gibi olabilir:

  ⟨{1}{3}{7, 8}⟩ → c_1 [sup = 2/5, conf = 2/3].

- **1., 2. ve 5. sequence'ler bu kuralı kapsar (cover).**
- **1. ve 2. sequence'leri bu kuralı karşılar (satisfy).**

26

Ödev

- Sıralı kural madenciliği ile ilgili bir uygulama alanı hakkında detaylı araştırma ödevi hazırlayınız.