

Veri Madenciliğinde Birliktelik Kuralları ve Ardışık Zamanlı Örüntüler

Hazırlayan: Melike Akalan

İçerik

- ❑ Birliktelik Kuralları Nedir ?
- ❑ Ardışık Zamanlı Örüntüler Nedir ?
- ❑ Uygulama Alanları
- ❑ Kullanılan Algoritmalar
- ❑ Algoritmaların Avantajları/Dezavantajları
- ❑ Sonuç ve Yorum

Birliktelik Kuralları ve Ardışık Zamanlı Örüntüler

- Veri madenciliği modelleme yöntemlerindendir.
- Birliktelik kuralları ilişkisel veri tabanlarında veya veri kaynaklarında **eş zamanlı olarak gerçekleşen** hareketlerin birbirleri arasındaki ilişkileri bulmaya çalışırken;
- Ardışık zamanlı örtüler ise halihazırda birbiriyle ilişkisi var olan verilerin **belirli bir zaman aralığındaki** ilişkilerini ortaya koyar.

Birliktelik Kuralları ve Ardışık Zamanlı Örüntüler



Birliktelik Kuralları ve Ardışık Zamanlı Örüntüler

- «**Market Sepeti Analizi**» birliktelik kurallarının ve ardışık zamanlı örüntülerin kullanıldığı en yaygın örneklerden biridir. Bu örneğin amacı müşterinin daha fazla alışveriş yapacağı şekilde ilgili ürün reyonlarını düzenlemektir. Şimdilerde ise daha da gelişmiş haliyle birçok pazarlama firması tarafından tercih edilir. Özellikle sosyal medyada karşımıza çıkan birçok reklam da bu mantıkta oluşturulur.

Birliktelik Kuralları ve Ardışık Zamanlı Örüntüler

- Basitçe bir örnekle açıklayacak olursak; kitapyurdunda alışveriş yapan bir müşterinin seçmiş olduğu kitaba göre, karşısına sepete eklemesi için önerilen kitaplar çıkmaktadır. Birliktelik kuralları da bu öneri ürünleri bulmaya çalışır. Yani o an sepete eklenen ürünün yanında başkalarının hangi ürünleri tercih ettiği analiz edilerek bunlar üzerinden «**birliktelik kuralları**» çıkarılır. Bu sayede müşteriye daha fazla ürün satışı sağlanır.

Birliktelik Kuralları ve Ardışık Zamanlı Örüntüler

- Aynı örneği bu sefer de «**ardışık zamanlı örüntüler**» için inceleyecek olursak:
Müşteriye sepete eklemesi için önerilen ürünler başkalarının alışverişlerine göre değil kişinin yapmış olduğu daha önceki alışverişleri esas alınarak yapılır.
- Yani kişinin hangi sıklıkta alışveriş yaptığı?
hangi zamanlarda hangi ürünleri tercih ettiği?
gibi faktörler kullanılarak bir tavsiye verilir.

Birliktelik Kuralları

- Birlikte gerçekleşen durumların nitelikleri arasındaki ilişkiyi bulmayı amaçlar.
- İlk olarak gerçekleşmiş olan verilerin analizi yapılır. Bu veriler içerisindeki birliktelik hareketleri tespit edilir. Elde edilen birliktelik hareketlerine göre de yapılacak olan çalışmalar şekillenir.
- Ardışık zamanlı örüntülerden farkı ise veriler arasındaki ilişkiyi bulmaya çalışırken herhangi bir **zaman bilgisini kullanmaz**. Birlikte gerçekleşen durumlara, olaylara odaklanır.

Uygulama Alanları

- İnönü Üniversitesinin geliştirmiş olduğu «birliktelik kuralları madenciliği yazılımı.»
- Çapraz Satış, örneğin vodafone kullanan birinin yemeksepetindeki indirimlerden yararlanabilmesi.
- Trafik Kazalarında birliktelik kuralı analizi,
- Dağıtık Sistemlerde birliktelik kuralları ile sepet analizi,
- E-Ticaret Satışlarının, birliktelik kuralları ile analizi gibi makale çalışmaları da bulunmaktadır.

Algoritmalar

- **Apriori**
- AIS
- SETM
- CARMA
- Sampling
- Partitioning gibi daha pek çok farklı algoritma birliktelik kurallarının hesaplanmasında kullanılmaktadır.

Apriori Algoritması

- 1994'te Agrawal ve Srikant adlı kişiler tarafından geliştirilmiştir.
- Veri Madenciliğinde birliktelik kuralları algoritmaları içerisinde en çok bilinen ve de uygulanan bir algoritmadır.
- Elde ettiği bilgileri bir önceki adımdan sağladığı için algoritmanın adı «prior (önceki)» anlamından gelmektedir.
- Veri tabanlarında sıkça geçen ifadelerin keşfedilmesinde kullanılır.

Apriori Algoritması Parametreleri

- Birliktelik kuralında, öğeler arasındaki birliktelik, destek(support) ve güven(confidence) kavramlarıyla hesaplanır.
- A ve B iki farklı ürün olsun.
 $\text{Destek}(A) = A \text{ sayısı} / \text{Toplam alışveriş sayısı}$
 $\text{Destek}(A,B) = (A,B) \text{ sayısı} / \text{Toplam alışveriş sayısı}$
- Destek, alınan ürünün ya da öğenin orda bulunma olasılığını ya da kaç kere geçtiğini hesaplar.

Apriori Algoritması Parametreleri

- $\text{Güven}(A,B) = (\text{A,B sayısı} / \text{A'nın olduğu alışveriş sayısı})$
 $\text{Güven}(A \Rightarrow B) = \text{Destek}(A,B) / \text{Destek}(A)$
- Güven, B'nin A ile beraber alınma ya da bulunma olasılığını hesaplar.
- Oluşturulacak olan kuralların her biri güven ve destek değerleri ile tanımlanır. $A \Rightarrow B$ için (destek=5% güven=55%) gibi.

Apriori Algoritması

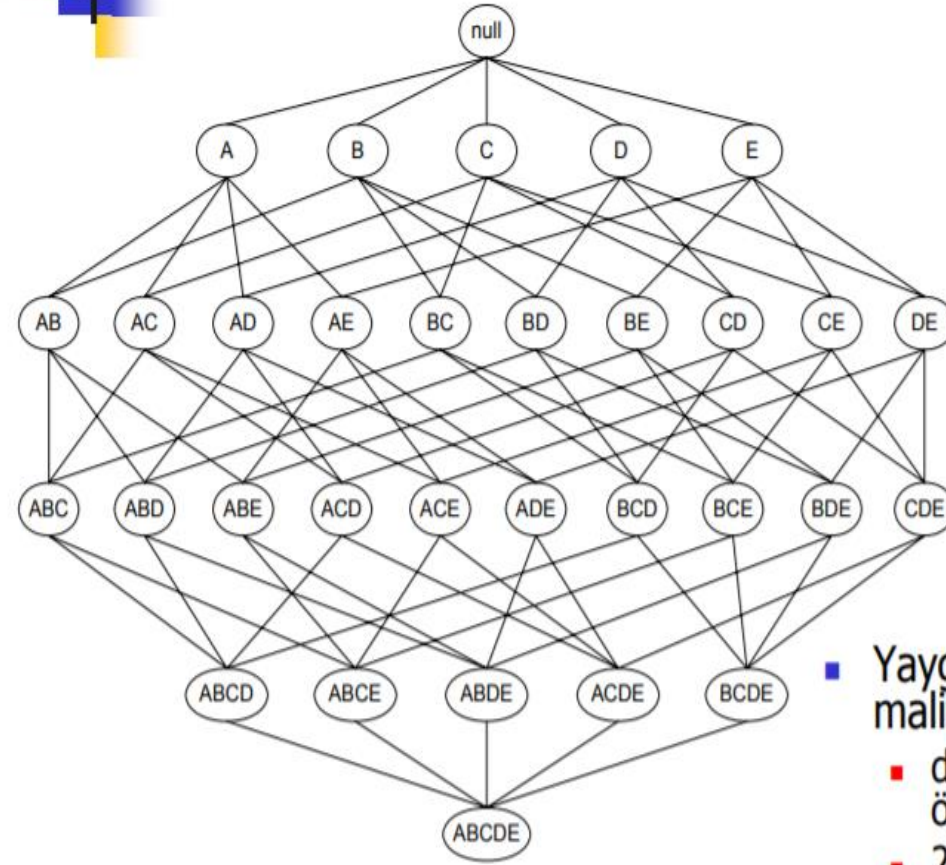
- Bir önceki slayttaki 5% destek oranı, yapılan tüm alışverişlerin 5%'inde A ile B'nin birlikte alındığını gösterir.
- %55 güven olasılığı ise A'yı alan müşterilerin 55%'inin aynı alışverişte B'yi de aldığını ifade eder.

Apriori Algoritması

- Algoritma temel mantığı;
«eğer k-öğeler kümesi minimum destek şartına uyuyorsa (yaygın öğeler ise), bu kümenin alt kümeleri de minimum destek şartına uyar.»
(onlar da sık tekrar eden yaygın öğelerdir.)
şeklindedir.
- K-öğeler kümesi, incelenecek veri setindeki k tane öğedir.

Apriori Algoritması

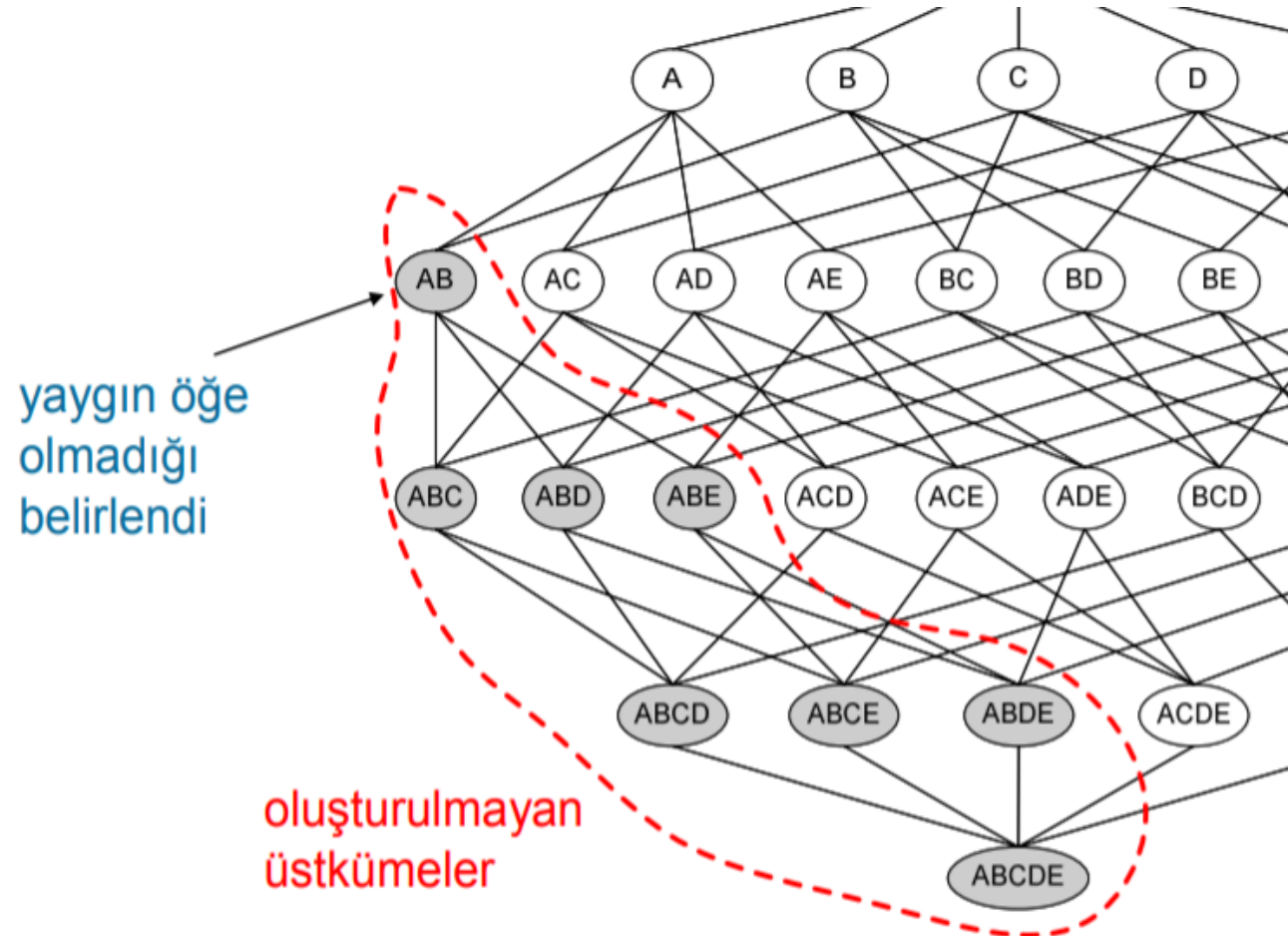
- Her yaygın öge adayı için veri kümesini taranarak hareketlerde yaygın öge adayı olup/olmadığına bakılır.



- Yaygın ögeleri bulmak maliyetli
 - d öge için $2^d - 1$ yaygın öge oluşabilir
 - $2^d - 1$ yaygın öge adayı

Apriori Algoritması

- Yaygın öge olmayan bir kümenin üst kümeleri de yaygın öge adayı oluşturamaz. (destek değeri hesaplanmaz)



Apriori Algoritması Aşamaları

1. Minimum destek sayısı ve minimum güven değerleri belirlenir.
2. Öğeler kümesi içerisindeki her bir öğenin destek değeri bulunur.
3. Minimum destek değerinden daha düşük desteği olan öğeler algoritmadan çıkarılır.
4. Elde edilen tekli birliktelikler ile ikili birliktelikler oluşturulur.

Apriori Algoritması Aşamaları

5. Minimum destek değerinden düşük olan öge kümeleri çıkarılır.
6. Üçlü birliktelikler oluşturulur.
7. Üçlü birlikteliklerden minimum destek değerini geçemeyenler çıkarılır.
8. Üçlü birlikteliklerden birliktelik kuralları oluşturulur.

Apriori Avantaj/Dezavantaj

- Algoritma satış, ticaret gibi alanlar için yararlı analizler yapabilmektedir.
- Sonuçların anlaşılması ve değerlendirilmesi oldukça kolaydır.
- **Dezavantajları:**
Çıkarılan kural sayısı fazla olabilir.
Yaygın öge bulmak için tüm veri setinde dolaştığından büyük veri setlerinde zaman sorunu yaşayabilir.

Ardışık Zamanlı Örüntüler

- Birbirleri ile ilişkisi olan fakat birbirini izleyen zamanlarda gerçekleşen ilişkileri ifade etmek için kullanılır.
- Ardışık zamanlı örüntü örnekleri:
bir memurun işe gidiş-gelişi,
telefon görüşme kayıtları,
maaş alma veya para yatırma gibi banka hesap hareketleri,
deprem gibi doğal afetlerin meydana geliş sırası vb.

Uygulama Alanları

- **Metin madenciliğinde** cümle içerisindeki kelimelerin sırasının elde edilmesinde,
- **Web madenciliğinde** müşteri profilinin çıkarılmasında,
- **Hastalık tespitinde** DNA sıra diziliminin analizinde,
- **Adli bilişimde** kişinin telefon görüşmeleri, banka hesap hareketleri, web geçmişinin incelenmesi gibi birçok alanda kullanılır.

GSP Algoritması (Generalized Sequential Pattern)

- Çalışma mantığı Apriori algoritmasından çok da farklı değildir. Tek farkı her öge için tek tek işlem yapmak yerine dizi sırası için işlem yapmasından kaynaklanır.
- Apriori 'de yaygın olmayan öğeleri çıkartıyorduk fakat burada örüntü(dizi) odaklı çalışıldığından onları da hesaplamalara dahil ediyoruz.

GSP Parametreleri

- Sıralı dizinin uzunluğu $|s|$: Sıralı diziyi oluşturan öğeler sayısıdır.
- k -sıralı dizi: k öğesi olan sıralı dizidir.
- Bir altdizinin destek değeri: Alt dizinin bulunduğu sıralı dizilerin toplam sıralı dizi sayısına oranıdır.
- Sıralı dizi örüntüsü: Destek değeri en küçük destek değerine eşit ya da büyük olan alt dizilerlerdir.

GSP Aşamaları

1. Veritabanını bir kez tarayarak 1 öğeden oluşan sıralı dizi örüntülerini bulunur.
2. Yeni bir sıralı dizi örüntüsü bulamayana kadar tekrarla
Aday oluşturma: $(k-1)$. adımda bulunan sıralı dizi örüntüleri birleştirilerek k uzunluklu sıralı dizi örüntü adayları bulunur.
Aday azaltma: $(k-1)$ alt dizileri en küçük destek değerini sağlamayan adaylar elenir.

GSP Aşamaları

2. **Destek değeri hesaplama:** Kalan sıralı dizi örüntü adayları için veritabanı taranarak destek değeri hesaplanır.
Aday eleme: En küçük destek değerini sağlamayan sıralı dizi örüntü adayları elenir. Böylece algoritma sonlanmış olur.

GSP

5. adım: 1 aday, 1 adet 5-sıralı dizi örüntüsü

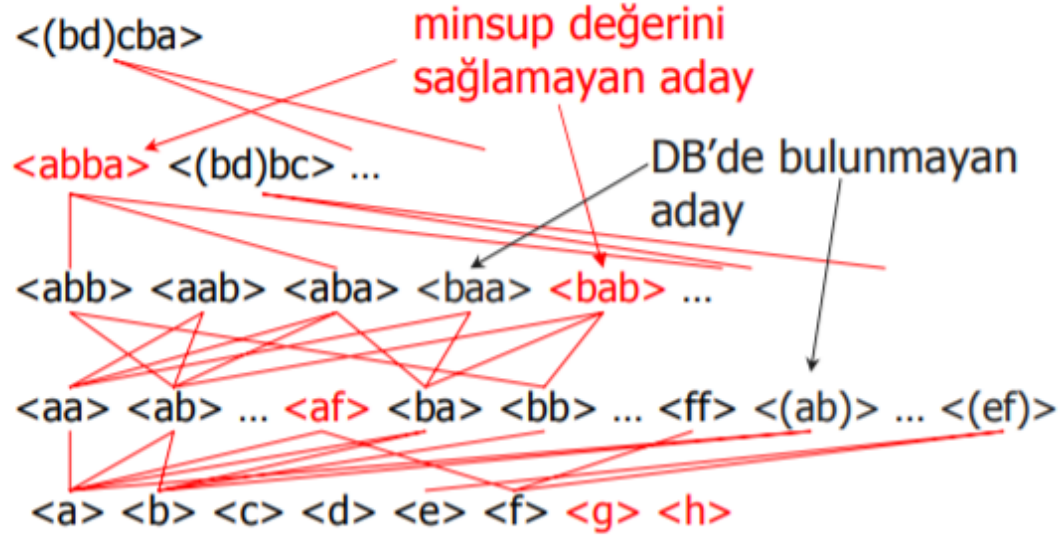
4. adım: 8 aday, 6 adet 4-sıralı dizi örüntüsü

3. adım: 46 aday, 19 adet 3-sıralı dizi örüntüsü. 20 aday DB'de değil

2. adım: 51 aday, 19 adet 2-sıralı dizi örüntüsü. 10 aday DB'de değil

1. adım: 8 aday, 6 adet 1-sıralı dizi örüntüsü.

$minsup = 2$



ID	Sıralı dizi
10	$\langle (bd)cb(ac) \rangle$
20	$\langle (bf)(ce)b(fg) \rangle$
30	$\langle (ah)(bf)abf \rangle$
40	$\langle (be)(ce)d \rangle$
50	$\langle a(bd)bcb(ade) \rangle$

GSP Avantaj/Dezavantaj

- Aday öğeleri elememesi onları da değerlendirmesi, öğeler arasındaki zaman ilişkisini de hesaba katması gibi **avantajları** bulunur.
- Veritabanını çok fazla taraması,
- Çok fazla sıralı örüntü oluşturabilmesi
- Uzun sıralı örüntülerin belirlenmesinin zorluğu, gibi **dezavantajları** bulunur.

Sonuç ve Yorum

- Giderek artan veri miktarı ve de tüketim bunlar üzerinden bir kural oluşturma, ilişki kurma ihtiyacı ortaya çıkarmıştır. Ortaya çıkarılan ilişkilerdeki amaç verileri daha anlamlı daha kurallı daha yararlı hale getirerek daha fazla kolaylık daha fazla kazanç sağlamaktır.
- Bu gibi işlemler de bahsedilen Apriori, Gsp gibi algoritmalar aracılığıyla sağlanır hem birliktelik hem de zamana bağlı birliktelik ilişkileri beraber incelendiğinde ise daha gerçekçi sonuçlara ulaşılmaktadır.

Kaynakça

- http://biostatapps.inonu.edu.tr/?page_id=847
- <https://dergipark.org.tr/en/download/article-file/561376>
- <https://dergipark.org.tr/tr/download/article-file/773189>
- <https://ab.org.tr/ab16/bildiri/46.pdf>
- <https://ab.org.tr/ab14/bildiri/134.pdf>
- <https://web.itu.edu.tr/~sgunduz/courses/verimaden/slides/d8.pdf>

- Teşekkürler.