

Apriori Algoritması ve Türkiye'deki Örnek Uygulamaları

M. Emin Eker¹, Recai Oktaş², Gökhan Kayhan³

¹ Ondokuz Mayıs Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Samsun

² Ondokuz Mayıs Üniversitesi, Bilgisayar Mühendisliği Bölümü, Samsun

³ Ondokuz Mayıs Üniversitesi, Bilgisayar Mühendisliği Bölümü, Samsun

emineker@bil.omu.edu.tr, roktas@bil.omu.edu.tr, gokhan@bil.omu.edu.tr

Özet: Veri madenciliği daha önceden bilinmeyen, geçerli ve uygulanabilir bilgilerin geniş veri tabanlarından elde edilmesi ve bu bilgilerin işletme kararları verirken kullanılmasıdır. Veri madenciliğinde birliktelik kuralı çıkarım algoritmalarından biri Apriori algoritmasıdır. Apriori algoritması birlikte olma ilişki analizlerinin yapıp birliktelik kurallarının ortaya çıkartılması konusunda en çok bilinen ve kullanılan algoritmadır. Geniş nesne kümelerinin ortaya çıkartılması işlemleri için kullanılır. Bu çalışmada veri madenciliğine giriş yapılacak, Apriori algoritmasının yapısı ve uygulamaları anlatılacaktır.

Anahtar Sözcükler: Veri Madenciliği, Birliktelik Kuralları, Apriori Algoritması, Örnek Apriori Algoritması Uygulamaları.

Apriori Algorithm and Sample Applications in Turkey

Abstract: Data mining using for obtaining and discovering valid and applicable information from large databases and also using these information while making business decisions. One of the extraction of association rules algorithm on data mining is the Apriori algorithm. Apriori algorithm is the most known and used algorithm about making the connection analysis and exposing the connection rules. It is used for exposing the large itemset operations. In this paper, data mining will be introduced, structure and applications of Apriori algorithm will be described.

Keywords: Data Mining, Association Rules, Apriori Algorithm, Example Apriori Algorithm Applications.

1. Giriş

Bilgi sistemlerinin, her geçen gün ucuzlaması ve güçlerinin giderek artması nedeniyle yaşamın her alanına hızla girmektedir. İşlemcilerin hızlanması, disk kapasitelerinin artması, bilgisayar ağlarındaki ilerleme sonucu her bir bilgisayarın başka bilgisayarlardaki verilere ulaşması olanağı, bilgisayarların çok büyük miktardaki verileri saklayabilmesine ve daha kısa sürede işleyebilmesine olanak sağlamaktadır.

Büyük miktarlardaki anlamsız verilerin anlamlı hale getirilmesi için istatistiki yöntemlerle çeşitli algoritmalar geliştirilmiş, ve bu algoritmalar uygulanarak anlamlı sonuçlar elde edilmeye çalışılmaktadır. Elde edilen anlamlı veriler ile de konu ile ilgili kişi veya kurumlara katkı sağlamak hedeflenmektedir. Çıkarılan bu anlamlı sonuçların ilgili kurumlara katkı sağlanması beklenmektedir.

2. Veri Madenciliği

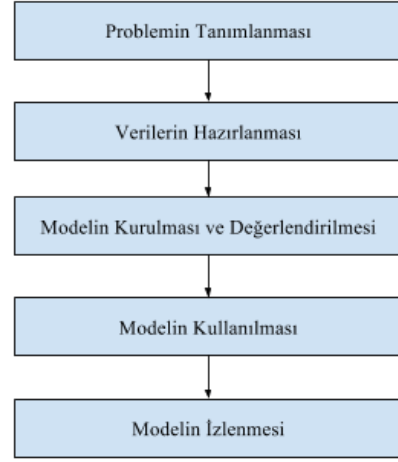
Teknolojinin büyük hızla gelişmesi sonucu bu şekilde durmadan büyüyen ve işlenmediği sürece değersiz gibi görünen veri yığınları oluşmaktadır. Bu veri yığınlarını, içlerinde altın madenleri bulunan dağlara benzetmek mümkündür. Bu madenlere ulaşmak için kullanılan yöntem ise “*Veri Madenciliği*” dir.

Veri madenciliği, önceden bilinmeyen, veri içinde gizli, anlamlı ve yararlı örüntülerin büyük ölçekli veritabanlarından otomatik biçimde elde edilmesini sağlayan bilgi keşfi süreci içindeki bir adımdır.

Verileri kaydetmek, yalnızca maden yataklarını oluşturmaktır. Bu veriler sadece veri takibi gibi operasyonel amaçlarla kullanılırsa müşteri ilişkileri açısından sadece bir kayıt olmaktan ileriye gidemezler. Verileri sadece kayıt olmaktan çıkarıp değerli bir veri madenine sahip olmak için elimizdeki bu bilgileri değerlendirmemiz şarttır.

Günümüzde bankacılık, sigorta ve borsa gibi birçok alandaki bütün bilgiler bilgisayar sistemleri sayesinde çeşitli veri tabanlarına kaydedilmektedir. Veri madenciliği, eldeki yapısız veriden, anlamlı ve kullanışlı bilgiyi çıkarmaya yönelik çalışmalarının bütünü olmuştur. Sonuç olarak yıllar ilerledikçe ortaya çıkan veri yığınlarından bir anlam çıkarma, başka bir anlamda potansiyel olarak kullanışlı bilgi haline getirme amacıyla Veri Madenciliği ortaya çıkmıştır.

Veri madenciliği rekabetin oldukça güçlendiği piyasalarda firmalar, iş sürekliliğini devam ettirmek ve daha ileriye taşıyabilmek adına birtakım değerlerinin yönetilmesinde büyük rol oynamaktadır. Bu değerlerin başında müşteri bilgileri gelmektedir. Yapılacak işlemler ile var olan müşterilerin elde tutulması ve müşteri davranışlarının tespit edilerek yeni müşteri kazanımı yapılabilir. Örneklenen müşteri davranışlarının bu şekilde inceleniyor olması kısmen müşteri mahremiyetine zarar vermektedir.



Şekil 1. Veri Madenciliği Süreci

Veri tabanı, sistematik erişim imkanı olan, yönetilebilir, güncellenebilir, taşınabilir, birbirleri arasında tanımlı ilişkiler bulunabilen bilgiler kümesidir. Belirli bir amaca yönelik düzen verilmiş kayıt ve dosyaların tümüdür. Veri tabanında bilgi keşfi, verilerin doğru, faydalı ve anlaşılır modeller ve kalıplar elde etmede kullanılan özel bir süreçtir.

3. Birliktelik Kuralları ve İlişki Analizi

Veritabanlarındaki bilgi miktarı arttıkça birçok kurum ve kuruluş sahip oldukları bilgiler arasındaki ilişkileri ortaya çıkarma çabası içerisine girmiştir. Böylesi yığınlar halindeki bilgiler arasındaki ilişkiler kurumlar için altın değerinde sonuçlar doğrulabilecek kararlarının alınmasında önemli rol oynamaktadır. İlişki analizi veritabanındaki bir dizi bilgi ya da kaydın diğer kayıtlarla olan bağlantısını açıklayan işlemler dizisidir. Yani bir kayıt varken, diğer bir üçüncü hatta dördüncü kaydın var olma olasılığı nedir? Ya da bu iki kayıt varken, diğer bir üçüncü hatta dördüncü kaydın veritabanına girme olasılığı nedir? İlişki analizi bu tür soruların yanıtını verir ve verilerin birlikte olan kurallarını ortaya çıkarır.

Birliktelik kuralları matematiksel olarak aşağıdaki gibi ifade edilebilir [5]:

$I = \{ I_1, I_2, \dots, I_m \}$ bir dizi – nesne kümesi olsun.

$T = \{ t_1, t_2, \dots, t_n \}$ ise veritabanındaki işlemleri (alışverişi) gösterebilir.

Herbir t_k nın alacağı değer 0 veya 1 dir. Eğer $t_k = 0$ ise I_k satın alınmamış, eğer $t_k = 1$ ise I_k satın alınmış demektir. Herbir işlem için veritabanında ayrı bir kayıt vardır. Şimdi $X \subseteq I$ için X 'teki herbir I_k ya karşılık gelen t_k değeri, $t_k = 1$ dir.

Bu birliktelik kuralıyla şu şekilde ifade edilmektedir:

$X \Rightarrow I_J$, X , I nın bir alt kümesidir. I_J ise I içindeki herhangi bir elemandır ve bu eleman X içinde yer almamaktadır. $X \Rightarrow I_J$ kuralının T için uygun olduğunun söylenebilmesi için belli bir güven seviyesinden söz etmek gerekecektir. Yani, T için deki tüm X 'lerin ne kadarının I_k yı sağladığı $\%c$ değeriyle ifade edilmektedir. Bu durumda, birliktelik kuralını $0 \leq c \leq 1$ güven seviyesiyle birlikte şöyle ifade edebiliriz. $X \Rightarrow I_J$ $X \Rightarrow I_J | c$. Güven seviyesi, kuralın gücünü ifade etmektedir.

Bunun dışında, kuralın destek seviyesinden de söz edilir. Destek seviyesi ise T içindeki işlemlerin ne kadarının X 'i sağladığıdır. Verilerin bir veritabanında, birliktelik analizinin yapıp birliktelik kurallarının ortaya çıkartılması, kullanıcının vereceği en küçük destek seviyesi ve en küçük güven seviyesinden daha büyük güven ve destek seviyelerine sahip kuraların tespit edilmesidir. En küçük destek seviyesini sağlayan nesne – kümelerine geniş nesne kümesi, diğerlerine ise küçük nesne kümesi denilir [5].

Veriler üzerinde birliktelik analizi yapıldıktan sonra ortaya çıkan sonuç genellikle aşağıdaki gibi ifade edilir.

Yaş (kişi, “20 – 30”)

→

satın alır(Kişi, “LCD TV”)

[DESTEK = %2, Güven = %13]

Yukarıdaki ifade yaşı 20 – 30 arasındaki kişilerin LCD TV almaya yatkın olduğunu ve bugüne kadar alışveriş yapan kişilerin içinden %2 sinin bu durumda olduğunu ve yaşı 20 – 30 arasında değişen müşterilerin %13'ünün LCD TV aldığını ifade eder. Örneğin;

Yaş (kişi, “20 – 30”) ^ Cinsiyet(Kişi, “erkek”)

→

satın alır(Kişi, “LCD TV”)

[DESTEK = %1, Güven = %36]

ifadesiye, yaşı 20 – 30 arasında değişip aynı zamanda erkek olan müşterilerden LED TV alanların tüm müşterilere oranının %1 olduğunu ve yaşının 20 – 30 arasında değişip aynı zamanda erkek olanların %60 ının LED TV aldığını ifade eder. İlk kural tek boyutludur, çünkü sadece yaş ve değişkeni verilmiştir. İkinci kurala iki boyutludur., çünkü burada değişken sayısı, yaş ve cinsiyet olmak üzere ikidir. Çok boyutluya bir başka örnekse aşağıdaki gibi verilebilir.

Yaş (kişi, “20 – 30”) ^ Cinsiyet(Kişi, “erkek”)

^ satın alır(Kişi, “LCD TV”)

→

satın alır(Kişi, “DVD Kaydedici”)

[DESTEK = %1, Güven = %65]

Burada da yaşı 20 – 30 arasında değişen erkek müşterilerden LED TV satın alanların %65'inin aynı zamanda DVD Kaydedici satın aldığını ifade etmektedir [2].

Bu gibi ilişki çıkarma işlemlerin yapıp kuralların çıkartılabilmesi için çeşitli algoritmalar geliştirilmiştir. AIS algoritması 1993'te ortaya çıkmış, daha sonra SETM algoritması ve 1994'te ortaya atılan ve bu günlerde çok kullanılan Apriori ve AprioriTid algoritmaları geliştirilmiştir.

4. Apriori Algoritması

Apriori algoritması birliktelik analizlerinin yapıp birliktelik kurallarının ortaya çıkartılması konusunda en çok bilinen ve kullanılan algoritmadır. Geniş nesne kümelerinin ortaya çıkartılması işlemleri için kullanılır.

Geniş nesne kümelerini ortaya çıkartan algoritmalar eldeki tüm verileri birçok kez tararlar. İlk taramada, herbir nesnenin destek seviyesi, hesaplanarak kullanıcı tarafından başlangıçta girilen minimum destek seviyesi ile karşılaştırılır ve herbir nesnenin geniş olup olmadığına bakılır. Bundan sonraki her tarama bir önceki taramada geniş olarak tespit edilmiş nesnelerden başlar ve geniş nesne kümeleri oluşturulur. Bu geniş nesne kümelerine aday nesne kümeleri denir. Taramanın sonunda ise hangi aday nesne kümesinin gerçekten geniş olduğu kontrol edilir. Daha önce de belirtildiği gibi bir nesne kümesinin geniş olarak adlandırılabilmesi için o nesne kümesinin kullanıcı tarafından verilen *minimum* destek seviyesinin üzerinde bir destek seviyesine sahip olması gerekir. Bir sonraki taramada, yine bir önceki taramada geniş olarak seçilen nesne kümelerinden başlanır ve veritabanının sonuna kadar bu nesne kümelerinin destekleri hesaplanır. Bu işlem, başka yeni geniş nesne kümeleri bulunamayana kadar sürer [5].

AIS algoritması geniş nesne kümeleri üretmek için geliştirilmiş bir algoritmadır. SETM ise sadece iki elemanlı kümeler oluşturur. Bu algoritmalar birçok kez veri tabanı tarama işlemi uyguladılar. Apriori algoritması ise daha önceden ortaya atılmış olan AIS ve SETM algoritmalarından herbir geçişte aday nesne kümelerinin sayılma ve bu aday kümelerinin üretilme şekliyle ayrılır. Hem AIS algoritmasında hem de SETM algoritmasında, tarama esnasında, veriler okunurken aday nesne kümeleri üretilir. Bir işlem (*T - transaction*) yapılır ve geniş nesne kümelerinin bu işlemlerde olup olmadığına da bakılır. Yeni aday nesne kümelerinin üretilmesi ise işlemlerdeki diğer nesnelerle

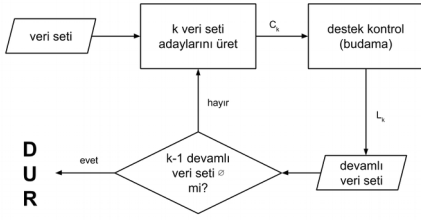
elde edilen geniş nesne kümelerinin birleştirilmesiyle üretilir [5]. Tabii bu da, gereksiz yere, aslında küçük nesne kümesi olan birçok aday nesne kümesinin sanki geniş nesne kümesiymiş gibi üretilmesi ve sayılması sonucunu doğurur. Bu da algoritmanın zaman karmaşıklığını artırır.

Apriori algoritması ise aday nesneleri üretirken veritabanındaki işlemleri hiç işin içine sokmadan, yalnızca bir önceki taramada geniş olduğu tespit edilmiş nesne kümelerini kullanarak oluşturur. Apriori algoritması geniş bir nesne kümesinin her-hangi bir alt kümesinin de geniş olacağı kabulüne dayanır. Böylece k adet nesneden oluşmuş bir nesne kümesi, $k-1$ adet nesneye sahip geniş nesne kümelerinin birleştirilmesi ve ait kümeleri geniş olmayanların silinmesiyle elde edilebilir. Bu birleşme ve silme işlemi sonunda daha az sayıda aday nesne kümeleri oluşacaktır.

Agrawal ve Srikant tarafından geliştirilen Apriori algoritması 1994 yılında 20. VLDB (Very Large Database Endowment) konferansında sunulmuştur. Bu bildiri de, Agrawal ve Srikant algoritmasının çalışma ayrıntılarını ve algoritmanın laba kodunu şu şekilde sunar [5];

- Verilerin ilk taraması esnasında, geniş nesne kümelerinin tespiti için, tüm nesneler sayılır.
- Bir sonraki tarama, k ıncı tarama olsun, iki aşamadan oluşur;
- Apriori-gen fonksiyonu kullanılarak, $(k-1)$ inci taramada elde edilen, L_{k-1} nesne kümeleriyle, C_k aday nesne kümeleri oluşturulur,
- Sonra veritabanı taranarak, C_k daki adayların desteği sayılır.
- Hızlı bir sayım için, verilen bir I işlemindeki, C_k yı oluşturan adayların çok iyi bilinmesi gerekir.

İkinci Tarama



Tablo 1. Apriori Algoritması Akış Diagramı

Örnek:

Aşağıda bulunan bir alışveriş tablosu örneğindeki verilere minimum destek %25 ve güven %55 olacak şekilde Apriori algoritmasını uygulayalım.

ID	SEPET
1	Ayran, Tuz, Şeker, Simit
2	Ayran, Tuz, Simit
3	Makarna, Simit
4	Makarna, Sucuk
5	Ayran, Tuz
6	Ayran, Tuz, Makarna

Tablo 2. Veri Tablosu

Birinci Tarama

Ürün	Miktar	Destek
Ayran	4	%67
Tuz	4	%67
Şeker	1	%17
Simit	3	%50
Makarna	3	%50
Sucuk	1	%17

Tablo 3. Birinci Tarama Sonucu

Ürün	Miktar	Destek
Ayran, Tuz	4	%67
Ayran, Simit	2	%33
Ayran, Makarna	1	%17
Tuz, Simit	2	%33
Tuz, Makarna	1	%17
Simit, Makarna	1	%17

Tablo 4. İkinci Tarama Sonucu

Üçüncü Tarama

Ürün	Miktar	Destek
Ayran, Tuz, Simit	2	%33

Tablo 5. Üçüncü Tarama Sonucu

En Geniş Kümeler ve Yüzdelikleri:

- Ayran alanlar (4 kayıt), Tuz ve Simit alır (2 Kayıt) [Destek %33, Güven = %50]
- Tuz alanlar (4 kayıt), Ayran ve Simit alır (2 Kayıt) [Destek %33, Güven = %50]
- Simit alanlar (3 kayıt), Ayran ve Tuz alır (2 Kayıt) [Destek %33, Güven = %67]
- Ayran ve Tuz alanlar (4 kayıt), Simit alır (2 Kayıt) [Destek %33, Güven = %50]
- Ayran ve Simit alanlar (2 kayıt), Tuz alır (2 Kayıt) [Destek %33, Güven = %100]

Görüldüğü gibi 3 ve 5 numaralı (en geniş) nesne kümeler, başlangıçta verilen %25 destek ve %55 güven seviyeleri ölçütünü karşılamaktadır.

Simit alanlar (3 kayıt), Ayran ve Tuz alır (2 Kayıt) [Destek %33, Güven = %67]

Ayran ve Simit alanlar (2 kayıt), Tuz alır (2 Kayıt) [Destek %33, Güven = %100]

Elimizdeki veritabanına göre Ayran – Tuz – Simit en geniş nesne kümesidir. Güven seviyesi ise bakış açısına göre değişmektedir.

Örneğin her ayran ve simit alan, mutlaka tuz da satın almıştır; bununla beraber her ayran ve tuz alan, mutlaka simit almıştır diyemiyoruz.

5. Apriori Algoritması Uygulamaları

Geleneksel kullanım alanı marketlerde ürünler arası ilişkileri tanımlamaktır. Benzer şekilde hızlı tüketim mamulleri üreticisi firmalarda depo sistemlerinin optimizasyonunda da kullanılabilir. Genel olarak birlikte sevk edilen ürünlerin yakın raflara yerleştirilmesi depo içindeki hareketi ve taşıma miktarını azaltıcı sonuçlar sağlayacaktır. Yöntem restoranlarda servis hızının artırılması için de çözümler sağlayabilir. Müşterilerin sipariş etme ihtimali olan ürünleri önceden tahmin ederek hazırlamak veya ilişkili ürünlerden mönüler oluşturmak gibi çözümler üretilebilir.

Verilen bu örnek ile daha bir çok alanda Apriori algoritması kullanılmaktadır. Bu bölümde Apriori algoritması ile yapılan araştırmalar, bildiriler ve makaleler listelenmiştir. Bu derleme ilgili algoritmanın yerelde hangi farklı alanlarda kullanıldığını tespit etmek için yapılmıştır.

Ulaş(2001) tarafından yılında yapılan bir yüksek lisans çalışmasında, sepet analizi gerçekleştirilmiştir. Büyük süpermarket zinciri olan Gima Türk A.Ş.'nin verileri üzerine Apriori algoritması uygulanmış ve ortaya çıkan sonuçlar incelenmiştir. Ayrıca mal satışları arasındaki ilişkileri bulmak amacıyla da, bileşen analizi ve k-ortalama öbeklemesi metotları kullanılmıştır [10].

Toprak(2004) tarafından ilişkisel veri tabanları üzerinde çoklu ilişkisel yapıdaki ortak kuralları bulmayı sağlayan bir uygulama geliştirilmiştir. Uygulama altyapısı olarak ilişkisel veri tabanlarındaki desenleri tanımlayabilen, bu desenleri eklerle geliştirebilen ve bu desenlerin çeşitli ölçmeleri için gerekli sayımları veri tabanından temel yetilerle alan bir yapı

kullanılmıştır. Bu altyapı, veri tabanının tanımında yer alan bilgileri kullanarak arama alanının daraltılmasını sağlamıştır. Bu çalışma, Apriori algoritmasını arama alanını daha da küçültmek için kullanarak ve altyapı tarafından desteklenmeyen özyinelemeli desenlerin bulunmasını sağlayarak altyapıya yenilikler getirmiştir. Apriori algoritması her tablo üzerinde sık karşılaşılan desenleri bulmak için kullanılmış ve bu algoritmanın gerekli destek değerini bulma yöntemi değiştirilmiştir. Veri tabanındaki özyinelemeli ilişkileri belirlemek için bir yöntem sunulmuş ve uygulama bu durumlar için tablo kısaltmalarının kullanıldığı bir çözüm sağlamıştır. Veri tabanı alanlarında saklanan sürekli değerleri bölümleyebilmek için eşit derinlik yöntemi kullanılmıştır. Uygulama bir veri madenciliği yarışması olan KDD Cup 2001'den alınan örnek genlerde yer tahmini problemi ile test edilmiş ve ortaya çıkan sonuçlar yarışmayı kazanan yaklaşımın sonuçlarıyla karşılaştırılmıştır [11].

Sıramkaya(2005) tarafından hazırlanan bir uygulamada internet üzerinden ulaşılabilen basın-yayın kaynaklarında yer alan görsel ve metinsel verilerin hızlı ve etkin bir şekilde erişimi ve bu kaynaklardan anlamlı ve önemli bilgilerin çıkarılması hedeflenmiştir. Çalışmalar istihbarat açısından önem taşıyan kişi ve örgütlerle ilgili haberler üzerinde yoğunlaşmıştır. Sunucu bilgisayarda internet üzerinde yer alan haber kaynaklarından toplanmış ve işlenmiş metinsel belgelerden oluşan veritabanı ile bu bilgileri işleyen uygulama yazılımları bulunmaktadır. Bir arayüz ile kullanıcının bu bilgileri sorgulaması sağlanmıştır. Çalışma, Birliktelik Kural Madenciliği tekniği ile uygulanmıştır. Bu teknik uygulanırken Apriori Algoritması kullanılmıştır. Yapılan veri madenciliği çalışmasında Bulanık Mantık çalışması, kişi-kişi ilişkilerini bulmakta uygulanmıştır. Bu uygulamadaki amaç kullanıcıların arama yapmak istedikleri kişilerin isimlerini yazarken yapabilecekleri yazım hatalarını elemektir. İsimlerdeki harflerin konumlarının birbirlerine göre uzaklıklarını temel olarak

bulanık mantık kurallarının uygulandığı bir algoritma kullanılmıştır [12].

Toprak(2005) tarafından yeni bir melez çok ilişkili veri madenciliği tekniği 2005 yılında gerçekleştirilmiştir. Bu çalışmada kavram öğrenme, kavram ile kavramı gerçekleştirme önkoşulları arasındaki eşleştirme olarak tanımlanmış ve ilişkisel kural madenciliği alanında buluşsal yöntem olarak kullanılan Apriori kuralı örüntü uzayını küçültmek amacı ile kullanılmıştır. Önerilen sistem, kavram örneklerinden ters çözümlülük operatörü kullanılarak genel kavram tanımlarını oluşturan ve bu genel örüntüleri Apriori kuralını temel alan bir operatör yardımı ile özelleştirerek güçlü kavram tanımlamaları elde eden melez bir öğrenme sistemi olarak tanımlanmıştır. Sistemin iki farklı sürümü, üç popüler veri madenciliği problemi için test edilmiş ve sonuçlar önerilen sistemin, en gelişkin ilişkisel veri madenciliği sistemleri ile karşılaştırılabilir durumda olduğunu göstermiştir [13].

Kılınç(2009) tarafından yapılan bir çalışmada birliktelik kuralları için bir yöntem sunmuştur. Apriori algoritmasının ürettiği kurallar elenerek bir elektronik firmasında üretim ve mal giriş kalite verileri üzerinde uygulanmıştır. Ortaya çıkarılan kurallar test verileri ile doğrulanmış ve sonuçlar analiz edilmiştir [14].

Yıldız(2010) tarafından yapılan bir çalışmada sık kümelerin bulunması için gizliliği koruyan bir yaklaşım önermiştir. Ayrıca bu çalışmada Matrix Apriori algoritması üzerinde değişiklikler yapılmış ve sık küme gizleme çerçevesi de geliştirilmiştir [15].

6. Sonuç

Yapılan araştırmalar dahilinde öncelikle veri madenciliğinde birliktelik kuralları analizi uygulamaları yapan yerel araştırmalar derlenmiştir. Derleme sonucunda en çok kullanılan algoritmanın Apriori algoritması olduğu sonucuna varılmıştır. Algoritmanın

diğer bazı benzer algoritmalarla olan farklılıkları ve benzerlikleri açıklanmıştır. Apriori algoritması dünyada bir çok kurumsal ve ticari kaynak üzerinde kullanılmıştır. Tüm dünyada olduğu gibi Türkiye'de de bir çok alanda Apriori algoritması kullanılmıştır. Konu kapsamı gereğince öncelikle algoritma örnekle açıklanmış ve özellikle Türkiye'de yapılan çalışmalara ulaşılmaya çalışılmıştır. Farklı konularda farklı örnekler listelenmeye çalışılmıştır.

7. Kaynaklar

- [1] Argüden, Y., Erşahin, B., "Veri Madenciliği" ISBN: 978-975-93641-9-9 (2010).
- [2] Silahtaroglu, G., "Veri Madenciliği" ISBN: 978-975-6797-81-5 (2014).
- [3] Özçakır, F.C., Çamurcu, A.Y., "Birliktelik Kuralı Yönetimi için Bir Veri Madenciliği Yazılımı Tasarımı ve Uygulaması", **İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi** Yıl: 6 Sayı:12 Güz 2007/2 s. 21-37.
- [4] Savaş, S., Topaloğlu, N., Yılmaz, M., "Veri Madenciliği ve Türkiye'deki Uygulama Örnekleri", **İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi** Yıl:11 Sayı: 21 Bahar 2012 s. 1-23.
- [5] Agrawal, R., Srikant, R., "Fast algorithms for mining association rules", **conference on Veri Large databases, Santiago, Chile** (1993).
- [6] Mishra, R., Choubey, A., "Comparative Analysis of Apriori Algorithm and Frequent Pattern Algorithm for Frequent Pattern Mining in Web Log Data", Rahul Mishra et al, / (IJCSIT) **International Journal of Computer Science and Information Technologies**, Vol. 3 (4) , 2012,4662 – 4665, ISSN: 0975-9646.
- [7] Bansal, D., Bhambhu, L., "Execution of APRIORI Algorithm of Data Mining Directed

Towards Tumultuous Crimes Concerning Women”, **International Journal of Advanced Research in Computer Science and Software Engineering**, ISSN: 2277 128.

[8] Tanna, R., Ghodasara, Y., “Using Apriori with WEKA for Frequent Pattern Mining”, **International Journal of Engineering Trends and Technology (IJETT)** – Volume 12 Number 3 - Jun 2014.

[9] Çöllüoğlu, Ö., Özdemir, S., “Analysis of Gifted Students’ Interest Areas Using Data Mining ”, **Journal of Gifted Education Research**, 2013, 1(3), 213-226.

[10] Ulaş, M. A, "Market Basket Analysis For Data Mining", **Computer Engineering, Boğaziçi University Master Thesis** (1999).

[11] Toprak, S., "Data Mining For Rule Discovery in Relational Databases”, **Middle East Technical University, Computer Engineering**, (2004).

[12] Sıramkaya, E., "Veri Madenciliğinde Bulanık Mantık Uygulaması", **Selçuk Üniversitesi, Fen Bilimleri Enstitüsü Yüksek Lisans Tezi**, (2005).

[13] Toprak, S.D., "A New Hybrid Multi-Relational Data Mining Technique", **Middle East Technical University, Computer Engineering, Master Thesis**, (2005).

[14] Kılınç, Y., "Mining Association Rules For Quality Related Data In An Electronics Company", **Middle East Technical University, Industrial Engineering**, (2009).

[15] Yıldız, B., "Impacts Of Frequent Itemset Hiding Algorithms On Privacy Preserving Data Mining", **İzmir Institute of Technology, Computer Engineering**, (2010).