Fırat Üniv. Müh. Bil. Dergisi
28(2), 207-216, 2016

Science and Eng. J of Fırat Univ.
28(2), 207-216, 2016

Maskelenmiş Veriler için Kümeleme-Tabanlı Şilin Atak Tespit Yöntemi

Alper Bilge, Zeynep Batmaz ve Hüseyin Polat*
Anadolu Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Eskişehir
*polath@anadolu.edu.tr

(Geliş/Received: 18.02.2016; Kabul/Accepted: 20.09.2016)

Özet

İnternet'in yaygınlaşması ile beraber hem ortak filtreleme hem de mahremiyetin korunması artan ilgi görmektedir. Mahremiyeti koruyarak doğru önerileri hızlı bir şekilde kullanıcıya sunmak üzere mahremiyet-tabanlı ortak filtreleme algoritmaları geliştirilmiştir. Ortak filtreleme algoritmaları gibi mahremiyet-tabanlı ortak filtreleme sistemleri de şilin ataklarına maruz kalabilir. Şilin atakları hedef ürünlerin popülaritesini yükseltmek veya düşürmek amacıyla kullanılan ataklardır. Bu ataklar filtreleme sisteminin veri tabanına birbirine benzeyen belli miktarda sahte kullanıcı profilinin eklenmesi ile gerçekleştirilir. Sahte profillerin tespit edilmesi gerekmektedir. Bu çalışmada maskelenmiş veri içeren mahremiyet-tabanlı ortak filtreleme sistemleri için kümeleme temelli bir şilin atak yöntemi tasarlanmıştır. Önerilen yöntemin başarısı gerçek veri ile yapılan deneylerle ölçülmüştür. Deney sonuçları önerilen metodun başarılı bir şekilde şilin atakları tespit ettiğini göstermiştir.

Anahtar Kelimeler: Şilin Atak, Ortak Filtreleme, Şilin Atak Tespit Etme, Mahremiyet, Doğruluk.

Clustering-based Shilling Attack Detection Method for Masked Data

Abstract

Collaborative filtering and privacy protection are receiving increasing attention with the widespread use of the Internet. Privacy-preserving collaborative filtering schemes have been proposed to provide accurate recommendations efficiently while preserving privacy. Like collaborative filtering algorithms, privacy-preserving collaborative filtering methods might be subjected to shilling attacks. These attacks are used to increase or decrease the popularity of target items. Shilling attacks are conducted by inserting fake profiles into filtering systems' databases. It is imperative to detect these fake profiles. In this study, we propose a clustering-based shilling attack detection method for privacy-preserving collaborative filtering schemes holding masked data. We perform real data-based experiments to evaluate the proposed scheme. Our empirical outcomes show that the method is able to successfully detect shilling attacks.

Keywords: Shilling Attack, Collaborative Filtering, Shilling Attack Detection, Privacy, Accuracy.

1. Giriş

İnternet'in gelişmesi ile beraber çoğu kullanıcı günlük işlerinin büyük bir bölümünü İnternet üzerinden gerçekleştirmektedir. İnternet'in yaygınlaşması e-ticaret sitelerinin sayısının hızla artmasına neden olmuştur. Günümüzde çoğu müşteri günlük alışverişlerini İnternet üzerinden e-ticaret sitelerini kullanarak yapmaktadır. Hem e-ticaret sitelerini kullanan kişi sayısı hem de bu sitelerde pazarlanan ürün sayısı hızla artmaktadır. Bu hızlı artış İnternet'teki veri miktarının artmasına neden olmaktadır. Bir başka deyişle, müşteriler kısa zamanda hoşlandıkları ürünlere erişmek ve satın almak istemektedir. Bu problem "bilgi bombardımanı" olarak adlandırılabilir. E-ticaret siteleri bu problemi çözmek ve müşterilerine hoşlanacakları ürünleri kısa zamanda önermek için ortak filtreleme (OF) sistemlerini kullanmaktadır.

OF terimi ilk defa 1992 yılında Tapestry isimli e-mail filtreleme sistemi ile ortaya konmuştur [1]. Öneri sunmak amacıyla geliştirilen OF algoritmaları temelde üç ana aşamadan oluşmaktadır [2]. Öncelikle n kullanıcıdan m ürün hakkındaki değerlemeleri toplanarak $n \times m$ boyutunda bir kullanıcı-ürün matrisi oluşturulur. Kullanıcılar bütün ürünler hakkında değerleme vermediklerinden bu matris

Bu çalışma TÜBİTAK tarafından desteklenmiştir (Proje No: 111E218).

genelde çok boşluklu bir matristir. Daha sonra öneri isteyen kullanıcının (aktif kullanıcı) komşuları belirlenir. Bunun için aktif kullanıcı ile kullanıcı-ürün matrisindeki her kullanıcı arasındaki benzerlik hesaplanır. Bu benzerliklere göre en benzer k kullanıcı komşu olarak belirlenir. Bir diğer komşu seçme yöntemi eşik temelli metottur. Bu yöntemde benzerliği belli bir eşik değerin üzerinde olan kullanıcılar komşu olarak seçilir. Son aşamada ise bir öneri algoritması kullanarak komşuların verisine dayalı olarak aktif kullanıcıya öneri hesaplanır. OF sistemleri iki temel hizmet sunmaktadır. Birinci servis tek bir ürün için öneri sunulması olarak tanımlanabilir. İkincisi ise aktif kullanıcıya en çok hoşlanacağı N ürünün yer aldığı üst-N öneri listesi sunmaktır.

Kötü niyetli kullanıcılar veya firmalar OF sistemlerinin sunduğu servisleri kendi avantajları doğrultusunda manipüle etmek isteyebilir. Bazı ürünleri hedef ürün olarak seçerek bu ürünlerin popülaritesini artırmak isteyebilirler. Veya bazı hedef ürünlerin popülaritesini azaltmak isteyebilirler. Bu amaçlarını gerçekleştirmek için OF sistemlerinin kullanıcı-ürün matrislerine sahte profiller ekleyebilirler. Bu tür saldırılara şilin ataklar veya profil enjeksiyon saldırıları adı verilir [3]. Bu saldırılar iki ana gruba ayrılır [3]. Ürün popülaritesini arttırmak amacıyla yapılan şilin ataklara *itme* (*push*) atakları adı verilir. Hedef ürünlerin popülaritesini azaltmak için gerçekleştirilen saldırılara ise *çekme* (*nuke*) atakları denir. En yaygın olarak kullanılan itme atakları arasında *rasgele* (*random*), *ortalama* (*average*), *yoklayıcı* (*probe*), *sürü* (*bandwagon*), *melez* (*hybrid*), *favori ürün* (*favorite item*), *bölüm* (*segment*) ve *mükemmel bilgi* (*perfect knowledge*) atakları sayılabilir. Popüler çekme atakları arasında ise *ters sürü* (*reverse bandwagon*) ve *sevme/nefret* (*love/hate*) atakları yer alır. Rasgele, ortalama, yoklayıcı, melez, favori ürün ve mükemmel bilgi atağı gibi ataklar hem popülariteyi yükseltmek hem de azaltmak amacıyla kullanılabilir [3].

Kullanıcılarına öneri hizmetleri sunan OF sistemleri şilin ataklara maruz kalabilir [4]. Müşteriler hoşlanacağı ürünleri satın alma eğiliminde olurlar. OF sistemlerinin sunduğu öneri hizmetleri bu açıdan önemlidir. Kötü niyetli kullanıcı veya firma kendi ürünlerinin popülaritesini artırarak daha fazla satılmasını hedefleyebilir. Veya rakip kişi veya firmaların ürünlerinin popülaritesini azaltarak bu ürünlerin daha az satılmasını amaçlar. Bu iki hedef doğrultusunda OF sistemlerinin kullanıcı-ürün matrislerine belli sayıda sahte profiller eklerler Sahte kullanıcı profiline örnek Şekil 1'de verilmiştir [5]. Bir sahte profil dört ana parçadan oluşur. Birinci kısım I_S olarak adlandırılır ve atağın karakteristiğini belirleyen K üründen oluşur. İkinci kısım I_F olarak isimlendirilir ve atağın tespit edilmesini zorlaştıracak şekilde doldurulacak L üründen oluşur. Üçüncü parça ise I_O olarak adlandırılır ve değerlendirme atanmamış O üründen oluşur. Son parça ise popülaritesi manipüle edilmek istenen hedef bir üründen oluşur ve I_T olarak adlandırılır.

I_S	I_F	I_O	I_T
S ürün: Atak karakteristiğini belirler	F ürün: Atak tespitini zorlaştırır	O ürün: Değerlendirme sunulmamış ürün	Hedef ürün

Şekil 1. Örnek bir sahte profil

Kullanıcılarına öneri hizmeti sunan OF sistemleri kullanıcılarının mahremiyetini korumalıdır. Doğru öneriler üretmek için doğru ve yeterli miktarda veriye ihtiyaç vardır. Eğer kullanıcı mahremiyeti korunmasa kullanıcılar ürünler hakkındaki değerlemelerini OF sistemleri ile paylaşmak istemeyebilir veya yanlış değerlemeler sunabilir. Bu durumda doğru öneriler sunmak zorlaşır. Müşteri mahremiyeti korunursa kullanıcılar doğru değerlemelerini paylaşmak için daha fazla motive olur. Bu amaçla mahremiyet-tabanlı ortak filtreleme (MOF) algoritmaları önerilmiştir [6, 7]. Önerilen bu algoritmalar kullanılan değerleme türüne göre sayısal değerleme ve ikili değerleme temelli algoritmalar olarak iki ana gruba ayrılmıştır. Değerlemelerin paylaşılma şekline göre merkezi sunucu- ve dağıtık veri-tabanlı olarak iki sınıfa ayrılabilir. Dağıtık veri temelli sistemler ise iki partili, çok partili ve eşten eşe olmak üzere üç alt sınıfa bölünebilir.

OF sistemleri gibi MOF sistemleri de şilin ataklara karşı savunmasızdır. MOF sistemleri şilin ataklarına maruz kalabilir ve bu sistemlere karşı şilin ataklar tasarlanabilir [8, 9]. MOF sistemlerinde mahremiyeti korumak için genelde rasgele karıştırma ve cevap yöntemleri kullanılır.

Bu sistemlerde mahrem veriler kullanıcıların değerlemeleri ve hangi ürünleri değerleyip değerlemedikleri kabul edilir. Bunun dışındaki veriler genel olarak kabul edilir. Mahremiyeti korumak için rasgele karıştırma yöntemi kullanıldığında, mahrem verilere rasgele sayılar eklenerek maskelenir. Bu nedenle geleneksel şilin atak tasarımları MOF sistemlerine karşı başarılı olmayabilir. Daha başarılı atak tasarımı için mahremiyeti koruyan yöntemler ve maskelenmiş veri göz önüne alınır [8, 9]. MOF sistemlerine karşı başarılı şilin atakların tasarımı yanında, sahte kullanıcı profilleri sisteme eklendiğinde gürbüz olarak çalışacak filtreleme sistemleri geliştirmek de önemlidir [10]. Bunlara ek olarak, MOF sistemlerindeki sahte kullanıcı profillerini tespit etmek gerekmektedir [11].

Sahte profiller MOF sistemlerinin sunacağı önerileri etkileyeceğinden, bu profillerin tespit edilerek kullanıcı-ürün matrisinden çıkarılması MOF sistemlerinin sağlıklı sonuçlar üretmesi için önemlidir. OF sistemlerindeki şilin atakların tespit edilmesi için değişik yöntemler önerilmiştir [4]. Fakat MOF sistemlerindeki veriler maskelenmiş veriler olduğundan, bu sistemlerdeki şilin atakları tespit etmek için farklı yöntemler geliştirilmelidir. Bu çalışmada, MOF sistemlerindeki sahte profilleri tespit etmek için ikiye ayırma kümeleme temelli bir şilin atak tespit etme yöntemi önerilmiştir. Maskelenmiş veri temelli kullanıcı profillerinden ikiye ayırma kümeleme algoritması kullanılarak bir ikili karar ağacı oluşturulur. Her bir yaprak düğüm bir küme olarak kabul edilir. Bu kümeler için küme içi ilgileşim değeri hesaplanır. Bu değere göre kümelerin sahte profiller içerip içermediklerine karar verilir. Sahte profiller birbirine çok benzediğinden aynı kümede yer alma olasılıkları fazladır. Sahte profiller içeren küme diğerlerine göre daha sıkı bir kümedir ve küme içi ilgileşim değeri en büyüktür. Bu bilgiler kullanılarak şilin atak kümesi tespit edilir ve bu kümedeki profiller kullanıcı-ürün matrisinden çıkarılır.

Bu çalışmada ikinci bölümde şilin atakların tespit edilmesi konusunda yapılan çalışmalar incelenmiştir. Üçüncü bölümde temel bilgiler verilmiştir. Dördüncü bölümde önerilen şilin atak tespit yöntemi açıklanmıştır. Önerilen yöntemin başarısını ölçmek için gerçek verilerle yapılan deneyler ve sonuçları beşinci bölümde sunulmuştur. Son bölümde ise makale sonuçları açıklanmış ve olası araştırma konuları verilmiştir.

2. İlgili Çalışmalar

OF sistemlerindeki şilin atakları tespit etmek için kullanılan yöntemler genelde sınıflandırma, kümeleme ve istatistiksel temelli yöntemlerdir [4]. En yaygın kullanılan tespit yöntemleri sınıflandırma ve kümeleme temelli metotlardır. Zhou ve ark. [12] rasgele ve ortalama ataklarla üretilen sahte profilleri tespit etmek için sınıflandırma-tabanlı bir yöntem kullanmıştır. Önerilen yöntem hedef ürünlerin analizine dayanmaktadır. Deneysel olarak başarısı ölçülen yöntemin rasgele ve ortalama ataklarla üretilen şilin atakları başarıyla tespit ettiği gözlenmiştir. Burke ve ark. [13] sahte profillerden hesaplanan öznitelikleri kullanarak şilin atakları tespit etmeyi önermiştir. Sınıflandırma temelli bu yöntemin genel şilin atak tespit etme yöntemlerinden daha başarılı olduğu görülmüştür. Öznitelik olarak hem genel hem de modele özgü öznitelikler kullanılmıştır. Cao ve ark. [14] basit Bayes sınıflandırıcı-tabanlı bir şilin atak tespit yöntemi geliştirmiştir. Bu yöntemde öncelikle sahte profil olduğu bilinen küçük bir sahte profil seti eğitim seti olarak kullanılarak basit Bayes sınıflandırıcı eğitilir. Bu sınıflandırıcı kullanılarak sınıfı bilinmeyen diğer profiller sınıflandırılmaya çalışılır.

Kümeleme-tabanlı şilin atak tespit yöntemleri de yaygın olarak kullanılmaktadır. O'Mahony ve ark. [15] sahte profilleri gerçek profillerden ayırt etmek için kümeleme algoritması kullanmıştır. Bunun için profillere komşuluk belirlemeye çalışılmış ve bu komşuluklardan sahte profiller içerenler tespit edilerek komşuluktan çıkarılmıştır. Mehta ve Nejdl [16] olasılık temelli bir kümeleme kullanarak sahte profilleri tespit etmeyi amaçlamıştır. Gerçek profilleri sahte profillerden ayırt etmek için profiller arasındaki benzerlikler hesaplanmıştır. Sahte profiller birbirine çok benzediğinden dolayı sahte profiller arasındaki benzerliklerin gerçek profiller arasındaki benzerliklere göre daha yüksek çıkması beklenmektedir. Deney sonuçları önerilen yöntemin eğitim setine ihtiyaç duymadan başarılı bir şekilde sahte profilleri tespit ettiğini göstermiştir. Zhang ve Kulkarni [17] şilin atak

profilleri arasındaki yüksek ilgileşimi kullanarak sahte profilleri tespit etmeye çalışmıştır. Bunun için spektral kümeleme kullanılmıştır. Bu kümeleme algoritması aralarında yüksek ilgileşim olan profilleri gruplandırmaktadır. Gerçek verilerle yapılan deneyler bu yöntemin başarılı bir şekilde sahte profilleri tespit ettiğini göstermiştir. Bilge ve ark. [18] ikiye ayırma kümeleme temelli bir şilin atak tespit etme yöntemi önermiştir. Bu metot sahte profilleri aynı kümeye gruplandırma eğilimindedir. Eğer bu küme tespit edilirse, sahte profiller dolayısıyla tespit edilmiş olur. Chakraborty ve Karforma [19] aykırı değer analizi kullanarak sahte profilleri tespit etmeye çalışmıştır. Bu analiz için değişik metotlar kullanılmıştır. Bunlardan biri PAM kümeleme algoritmasıdır. Eğer doldurma büyüklüğü yüksek ise bu metot başarılı bir şekilde şilin atak profillerini tespit etmektedir. Doldurma büyüklüğü küçüldükçe başarı da kötüleşmektedir.

OF sistemlerinde yer alan sahte profilleri tespit etmek için yapılan değişik çalışmalara rağmen, maskelenmiş veriler içeren MOF sistemlerindeki sahte profilleri tespit etmek için yapılan çalışmalar kısıtlıdır. Gunes ve Polat [11] MOF sistemlerindeki şilin atak profillerini tespit etmek için hiyerarşik kümeleme-tabanlı bir tespit algoritması önermiştir. Hiyerarşik olarak maskelenmiş kullanıcı profilleri kümelenmektedir. OF sistemlerince hesaplanan önerilerin manipüle edilmesi için sahte profillerin birbirine çok benzer olarak tasarlanması gerekmektedir. Bu nedenle bu profiller birbirine çok benzer. Kullanıcı profilleri maskelenmiş olsa bile, sahte profiller büyük olasılıkla aynı kümeye düşecektir. Gunes ve Polat [11] bu yaklaşımdan yola çıkarak sahte profilleri kümeleyip tespit etmeyi amaçlamıştır. Bizim çalışmamız ilgili çalışmalardan farklı olarak ikiye ayırma *k*-ortalama kümeleme algoritmasına dayalı bir şilin atak tespit yöntemi geliştirmeyi hedeflemiştir. Bu yöntem maskelenmiş veri içeren MOF sistemleri için geliştirilmiştir. Şilin ataklar hem OF hem de MOF sistemlerinde artan ilgi görmektedir. Fakat yapılan çalışmalar genelde etkili atak tasarımı ve gürbüz algoritma geliştirme üzerine yoğunlaşmıştır. Ayrıca OF sistemlerinde bu atakların nasıl tespit edileceği konusu yaygın olarak çalışılmıştır. Fakat MOF sistemlerinde atak tespit yöntemleri konusunda çalışmalar nispeten çok kısıtlıdır. Bu çalışma bu eksikliği gidermek için gerçekleştirilmiştir.

3. Veri Maskeleme ve Maskelenmiş Veri-Tabanlı Şilin Ataklar

OF sistemleri kullanıcıların mahremiyetini ihlal edebilir. Kullanıcılar sınıflandırılabilir, istenmeyen pazarlama yapılabilir, fiyat ayrımcılığına maruz kalınabilir ve takip edilebilir. Bu tür riskleri ortadan kaldırmak için MOF sistemleri önerilmiştir [20]. Polat ve Du [20] aşağıda açıklanan rasgele karıştırma yöntemi ile mahremiyetin korunabileceğini göstermiştir:

1. Kullanıcılar tekdüze veya normal dağılım kullanacaklarına karar verir.
2. Her kullanıcı 0 ile σ_{max} aralığından rasgele bir σ değeri seçer. Burada σ_{max} rasgele sayıların standart sapmasının üst sınırını, σ ise standart sapmasını temsil eder.
3. Her kullanıcı 0 ile β_{max} aralığından rasgele bir β değeri seçer. β_{max} doldurulacak boş hücrelerin üst sınırını yüzde değeri, β ise doldurulacak boş hücre yüzdesidir. Boş hücrelerin β yüzdesi kadarı rasgele seçilir ve rasgele sayı ile doldurulur.
4. Kullanıcılar değerlemelerini z-skor değerlerine dönüştürür ve bu değerlere rasgele sayı ekleyerek maskeler. Rasgele sayılar seçilen dağılım ve standart sapma değerleri ile üretilir. Maskelenmiş veri öneri sistemine gönderilir.

Maskelenmiş veri durumunda şilin ataklar sistemi etkili manipüle edecek şekilde tasarlanmalıdır. Popülariteyi artırmak için geliştirilen rasgele, ortalama, bölüm ve sürü şilin atakları maskelenmiş veri durumunda aşağıdaki gibi tasarlanmıştır [9]: Rasgele atak modelinde doldurulacak ürünler rasgele sayılarla doldurulur. Hedef ürüne en büyük rasgele sayı atanır. Ortalama şilin atak modelinde ürün değerlemelerinin ortalamaları hesaplanır. Tekdüze dağılımla üretilen rasgele sayılar bu ortalamalara eklenir doldurulacak ürünlerin hücresine eklenir. Hedef ürün yine en büyük rasgele sayı ile doldurulur. Bölüm atak modelinde popüler ürünler, doldurulacak ürünler

ve hedef ürün tespit edilir. Bunların sayısı kadar rasgele sayı üretilir. En büyük rasgele sayı hedef ürüne atanır. Geri sayılar büyükten küçüğe sıralanır ve sırayla popüler ürünlere ve doldurulacak ürünlere bu sayılar atanır. Sürü atak modeli bölüm şilin atağına benzer. Popüler ürünler yerine seçilmiş ürünler kullanılır ve aynı şekilde tasarlanır.

4. İkiye Ayırma *k*-Ortalama Kümeleme Temelli Şilin Atak Tespit Yöntemi

MOF sistemleri herkese açık olduğundan şilin ataklara maruz kalabilir. Sistemin doğru ve güvenilir öneriler üretebilmesi için sahte profillerin tespit edilmesi gerekir. Bu tür profillerin maskelenmiş veri içeren MOF sistemlerinde tespit edilmesi için ikiye ayırma *k*-ortalama kümeleme temelli bir tespit algoritması önerdik. Önceki çalışmamızda [18] ikiye ayırma *k*-ortalama kümeleme temelli şilin atak tespit etme metodu OF sistemleri için geliştirilmiştir. Fakat maskelenmiş veri durumunda hem atak tasarımları değişmekte hem de atak tespit yönteminin maskelenmiş veriden çıkarım yapması gerekmektedir.

Önerilen yöntemde öncelikle ikili karar ağacı oluşturulması gerekir. Bu ağaç Bilge Ve Polat [21] tarafından şu şekilde oluşturulmuştur: En uygun komşu sayısı (*N*) durma kıstası olarak seçilir. Her aşamada *k*-ortalama kümeleme algoritması kullanıcı-ürün matrisindeki kullanıcıları iki gruba ayırır. Ara düğümler küme kabul edilerek merkezleri hesaplanır. Her bir yaprak düğümde en fazla *N* kullanıcı kalıncaya dek kümelemeye devam edilir. Elde edilen bu ikili karar ağacında her bir yaprak düğümde en fazla *N* kullanıcı yer alır.

Şilin atak profilleri birbirine çok benzer olduğundan bu tür profillerin aynı yaprak düğümde veya aynı kümede toplanmaları beklenir. Bu nedenle sahte profilleri veya sahte profillerin çoğunu içeren bu küme veya yaprak düğümün belirlenmesi gerekir. Bunun için her kümenin küme içi ilgileşim değeri hesaplanır [18]. Kümenin merkezi hesaplanır. Her kullanıcının bu merkeze uzaklığı hesaplanır. Bu uzaklıkların ortalaması alınır. En düşük ortalamanın olduğu küme sahte profilleri içeren küme kabul edilir. Çünkü sahte profiller birbirine çok benzediğinden bu ortalamanın en düşük çıkması beklenir.

Mahremiyet endişeleri olmadığı zaman kullanıcılar gerçek değerlemelerini OF sistemleri ile paylaşır. Gerçek veriden ikili karar ağacı oluşturmak, küme merkezlerini bulmak ve benzerliklerin ortalamalarını hesaplamak kolaydır. Mahremiyet endişeleri olduğunda MOF sistemlerinde artık maskelenmiş veriler vardır. Bu durumda şilin atak tespit zorlaşmaktadır. Bütün işlemlerin maskelenmiş veriler üzerinden yapılması gerekmektedir. Atak tespit sırasında gerçekleştirilen işlemler maskelenmiş veriler üzerinde aşağıdaki gibi gerçekleştirilir. Boyu *m* olan **A** ve **B** vektörlerinin sırasıyla herhangi bir yaprak düğümde yer alan bir kullanıcıyı ve o kümenin merkezini temsil ettiğini varsayalım. Benzer şekilde, **A'** ve **B'** ise bu vektörlerin maskelenmiş halidir. Bu durumda maskelenmiş bu iki vektör arasındaki benzerlik veya uzaklık şöyle hesaplanır:

$$A' \cdot B' = \sum_{j=1}^{m}(a_j + r_j)\,(b_j + v_j) = \sum_{j=1}^{m}(a_j b_j + a_j v_j + r_j b_j + r_j v_j) \approx \sum_{j=1}^{m} a_j b_j \quad (1)$$

Formül 1'de r ve v değerleri rasgele sayıları temsil etmektedir. Bu rasgele sayılar ortalaması 0 olan sayılar olduğundan, formül 1'de yer alan son üç parçalı toplamın beklenen değeri sıfırdır. Bu nedenle formül 1 ile maskelenmiş iki vektör arasındaki benzerlik hesaplanır. Formül 1 ikiye ayırma *k*-ortalama kümeleme algoritması kullanarak ikili karar ağacı oluşturma sırasında herhangi iki maskelenmiş kullanıcı vektörleri arasındaki benzerliği bulmak için kullanılabilir. Bütün kullanıcı vektörleri (sahte vektörler dâhil) rasgele karıştırma yöntemi ile maskelenmiş olduğundan bunların yer aldığı kümenin merkezini temsil eden vektör de maskelenmiştir. Bu nedenle formül 1 ayrıca bir düğüm veya kümedeki herhangi bir maskelenmiş kullanıcı vektörünün bu kümenin merkezine olan uzaklığı veya benzerliğini hesaplamak için kullanılabilir.

Bir düğümdeki kullanıcıların küme merkezine olan uzaklıkları veya küme merkezi ile olan benzerlikleri formül 1 ile hesaplandıktan sonra bu benzerliklerin ortalamasının hesaplanması gerekmektedir. W_{CM} değeri *C* kümesi ile *M* kullanıcı vektörü arasındaki

benzerliği ifade eder. Bu benzerlikler maskelenmiş veriler üzerinden hesaplandığından, rasgele değerlerden dolayı bu benzerlik değerleri de maskelenmiş durumdadır. Bu nedenle herhangi bir kümede yer alan kullanıcıların küme merkezi ile olan benzerliklerinin ortalaması (*OB'*-maskelenmiş veriye dayalı ortalama benzerlik) aşağıdaki gibi hesaplanır:

$$OB' = \frac{1}{s}\sum(W_{CM} + R_{CM}) = \frac{1}{s}\sum W_{CM} + \frac{1}{s}\sum R_{CM} \approx \frac{1}{s}\sum W_{CM} \quad (2)$$

Formül 2'de yer alan *s* değeri bir kümede yer alan kullanıcı sayısını gösterir. R_{CM} değeri ise benzerlik değerindeki rasgele sayılarından dolayı oluşan rasgele değeri ifade eder. Maskelenmiş verilerle hesaplanan benzerliklerden dolayı oluşacak rasgele değerler pozitif veya negatif olabilir. Bu nedenle formül 2'de hesaplanan ikinci toplamın beklenen değeri sıfır olur. Ayrıca büyüyen *s* değerlerine bağlı olarak bu toplamın ortalaması gerçekte sıfıra yaklaşır.

Formül 2 yukarıda açıklandığı şekilde kümedeki kullanıcıların küme merkezine olan uzaklıklarının ortalamasını hesaplamak için kullanılabilir. Buna ek olarak, küme merkezlerinin belirlemesinde de kullanılabilir. Çünkü küme merkezleri o kümede yer alan maskelenmiş kullanıcı vektörlerinin ortalamasının hesaplanması ile bulunur. Kullanıcılar gerçek z-skor değerleri yerine rasgele karıştırma yöntemi ile maskelenmiş z-skor değerlerini MOF sistemine iletir. Herhangi bir *C* kümesinin merkezi (*C'*-maskelenmiş küme merkezi) aşağıdaki gibi hesaplanır:

$$C' = \left[\frac{\sum_{i=1}^{s} z'_{i1}}{s}, \frac{\sum_{i=1}^{s} z'_{i2}}{s}, \cdots, \frac{\sum_{i=1}^{s} z'_{im}}{s}\right] = \left[\frac{\sum_{i=1}^{s}(z_{i1}+r_{i1})}{s}, \frac{\sum_{i=1}^{s}(z_{i2}+r_{i2})}{s}, \cdots, \frac{\sum_{i=1}^{s}(z_{im}+r_{im})}{s}\right] = \left[\frac{\sum_{i=1}^{s} z_{i1}}{s} + \frac{\sum_{i=1}^{s} r_{i1}}{s}, \frac{\sum_{i=1}^{s} z_{i2}}{s} + \frac{\sum_{i=1}^{s} r_{i2}}{s}, \cdots, \frac{\sum_{i=1}^{s} z_{im}}{s} + \frac{\sum_{i=1}^{s} r_{im}}{s}\right] \approx \left[\frac{\sum_{i=1}^{s} z_{i1}}{s}, \frac{\sum_{i=1}^{s} z_{i2}}{s}, \cdots, \frac{\sum_{i=1}^{s} z_{im}}{s}\right] \quad (3)$$

Formül 3'de yer alan ikinci toplamların ortalaması rasgele sayıların toplamlarının ortalamasıdır. Bu rasgele sayılar ortalaması sıfır olan rasgele sayılar olarak üretildiğinden formül 3 kullanılarak maskelenmiş z-skor vektörlerinin ortalaması veya küme merkezleri hesaplanabilir.

Son olarak, küme içi ilgileşim değerlerinin (*ICC'*-maskelenmiş veriden hesaplanan küme içi ilgileşim değeri) maskelenmiş verilerden hesaplanması gerekmektedir. Bu değerler herhangi bir *C* kümesi için aşağıdaki formülle hesaplanır. Burada *n* ve *m* değerleri sırasıyla toplam kullanıcı ve ürün sayısını göstermektedir:

$$ICC'_C = \frac{\sum_{i=1}^{s}\sum_{j=1}^{m}\left(z'_{ij}z'_{Cj}\right)}{s} = \frac{\sum_{i=1}^{s}\sum_{j=1}^{m}(z_{ij}+r_{ij})(z_{Cj}+r_{Cj})}{s} \frac{\sum_{i=1}^{s}\sum_{j=1}^{m}(z_{ij}z_{Cj}+z_{ij}r_{Cj}+z_{Cj}r_{ij}+r_{ij}r_{Cj})}{s} \approx \frac{\sum_{i=1}^{s}\sum_{j=1}^{m} z_{ij}z_{Cj}}{s} \quad (4)$$

Yukarıda açıklandığı gibi maskelenmiş veriler ikiye ayırma *k*-ortalama kümeleme algoritması ile kümelere ayrılır. Bunun için benzerlikler maskelenmiş verilerden hesaplanabilir. Küme merkezleri yine maskelenmiş kullanıcı vektörlerinden hesaplanabilir. Maskelenmiş kullanıcı vektörlerinin küme merkezlerine uzaklıkları ve bu uzaklıklardan küme içi ilgileşim değerleri yine maskelenmiş veriden hesaplanabilir. Şilin atakların etkili olabilmesi için aynı strateji ile birbirine çok benzeyen sahte profiller üretilir. Bu profiller gerçek profillere göre birbirine daha çok benzer. Sahte profillerin aynı kümeye gruplandırılma olasılıkları çok fazladır. Sahte profiller birbirine çok benzediğinden küme içi ilgileşim değeri bu küme için en büyük olacaktır. Bu değer ne kadar doğru hesaplanırsa, sahte profilleri tespit etme başarısı o kadar iyi olacaktır. Yukarıda açıklandığı gibi maskelenmiş veriden gerekli değerlerin hesaplanabileceği gösterilmiştir. İkili karar ağacının kök düğümünden başlayarak *ICC'* değerlerine göre sağa veya sola dallanarak ağaç üzerinde hareket edilir. İlgileşim değerine göre yön belirlenir. Hangi düğümde durulacağı ise farklı seviyeler

arasındaki ilgileşim değerlerine göre karar verilir. Eğer farklı iki seviyede yer alan kümelerin ilgileşim değerleri diğerlerine göre daha küçük sapma gösteriyorsa, üst seviyedeki düğümün atak profilleri içerdiği kabul edilir. Bir kümede yer alan sahte profiller kümeleme ile iki alt kümeye ayrılır. Bu durumda küme içi ilgileşim değerlerindeki değişim (ρ) çok düşük olacaktır. Bu değişim miktarına göre üst seviyedeki düğüm ve dolayısıyla alt iki kümede sahte profiller olduğuna karar verilir.

5. Deneyler

Önerilen şilin atak tespit yönteminin başarısını ölçmek için gerçek veriler kullanılarak deney yapılmıştır. Bunun için MovieLens (MLP) veri seti kullanılmıştır. MLP 943 kullanıcının 1.682 film için verdiği 1 ile 5 arasında değişen kesikli değerlemelerini içerir ve toplam 100.000 değerleme vardır. Her kullanıcı en az 20 filmi değerlemiştir. Önerdiğimiz metodun başarısını ölçmek için *kesinlik* (*precision*), *bulma* (*recall*) ve *F1* ölçütü kullanılmıştır. Kesinlik sahte profil olarak doğru sınıflandırılan profil sayısının sınıflandırılan toplam profil sayısına oranıdır. Bulma ise doğru bir şekilde sahte profil olarak sınıflandırılan profil sayısının toplam sahte profil sayısına oranıdır. *F1* ölçütü ise $F1 = 2 \times Kesinlik \times Bulma / (Kesinlik + Bulma)$ olarak hesaplanır.

5.1. Deney sonuçları

Amacımız önerilen tespit algoritmasının sahte profil tespit etme (sahte profil sınıflandırma) başarısını ölçmektir. Bu nedenle mahremiyet kontrol parametre değerlerini sabitledik. Bunun için σ_{max} ve β_{max} değerlerini sırasıyla 2 ve 25 olarak seçip kullandık. İlk deneyde ρ parametresinin değişen değerlerine göre genel performansın nasıl değiştiğini inceledik. Diğer kontrol parametreleri olan atak büyüklüğü ve doldurma büyüklüğü 25 olarak seçildi. Rasgele karıştırmadan dolayı deneylerimiz 100 defa çalıştırıp sonuçları kesinlik ve bulma cinsinden Tablo 1'de gösterilmiştir. Tabloda **RA** rasgele atağı, **OA** ortalama atağı, **SA** sürü atağı ve **BA** bölüm atağını temsil etmektedir.

Tablo 1. Değişen ρ değerlerine göre tespit etme başarısının değişimi

	Kesinlik					*Bulma*				
ρ	1	2	4	7	10	1	2	4	7	10
RA	0,004	0,005	0,007	0,011	**0,017**	0,005	0,010	0,017	0,035	**0,095**
OA	0,840	0,844	0,840	0,841	**0,848**	0,764	0,770	0,766	0,766	**0,774**
SA	**0,708**	0,683	0,582	0,470	0,382	0,634	0,677	0,716	0,781	**0,809**
BA	**0,835**	0,812	0,762	0,714	0,686	0,663	0,702	0,825	0,866	**0,877**

Tablo 1'de verilen sonuçlar incelendiği kesinlik açısından değişim değeri 10 olduğunda rasgele ve ortalama atakları için en iyi sonuçlar elde edilmiştir. Diğer ataklar için en iyi başarı değişim 1 olduğunda elde edilmiştir. Bulma ölçütü açısından değişim 10 olduğunda yöntemimiz en başarılı sonuçları elde eder. Değişim değerleri büyüdükçe, kesinlik değerleri sürü ve bölüm atakları için kötüleşmektedir. Rasgele ve ortalama ataklar için değişim parametresinin değişen değerleri çok fazla etkili olmamaktadır. Bulma değerleri değişimin artan değerleriyle beraber bölüm ve sürü atakları için iyileşmektedir. Kesinlik ve bulma değerleri yanında *F1* ölçütü değerlerini de hesaplayıp Tablo 2'de gösterdik. Tablo 2'de verilen sonuçlar incelendiğinde Tablo 1'de görülen benzer eğilimler elde edilmiştir.

Tablo 2. Değişen ρ değerlerine göre *F1* ölçütü değerlerinin değişimi

ρ	1	2	4	7	10
RA	0,004	0,006	0,010	0,016	**0,028**
OA	0,800	0,806	0,801	0,802	**0,809**
SA	0,665	**0,670**	0,626	0,551	0,481
BA	0,737	0,751	**0,789**	0,776	0,753

İkinci deneyimizde atak büyüklüğü parametresinin değişen değerlerine göre yöntemimizin başarısını inceledik. Bunun için Tablo 2'de en iyi sonucu veren ρ değerlerini kullandık. Doldurma büyüklüğü 25 olarak sabitlenmiştir. Atak büyüklüğü ise 3 ile 25 arasında değiştirilmiştir. Rasgele karıştırmadan dolayı deneylerimiz 100 defa tekrarlanmış ve genel ortalamalar kesinlik ve bulma cinsinden Tablo 3'de sunulmuştur. Tabloda **AB** atak büyüklüğünü temsil etmektedir.

Tablo 3. Değişen atak büyüklüğüne göre tespit etme başarısının değişimi

	Kesinlik					*Bulma*				
AB	3	5	10	15	25	3	5	10	15	25
RA	0,014	0,021	**0,025**	0,020	0,004	**0,218**	0,188	0,118	0,061	0,007
OA	0,903	**0,944**	0,881	0,937	0,913	0,805	0,836	0,770	**0,843**	0,781
SA	0,401	0,579	0,564	0,512	**0,980**	0,336	0,508	0,541	0,822	**0,842**
BA	0,247	0,345	0,682	**0,964**	0,949	0,225	0,508	0,810	0,884	**0,993**

Tablo 3'deki sonuçlara göre ortalama, sürü ve bölüm ataklarıyla üretilen şilin atak profillerini önerdiğimiz yöntemin başarıyla tespit ettiği söylenebilir. Kesinlik ölçütü açısından en başarılı sonuçlar sürü atağı için elde edilmiştir. Bulma ölçütü açısından en başarılı sonuçlar bölüm atağı için gözlenmiştir. Rasgele atak için yöntemimizin başarısı düşüktür. Atak büyüklüğünün artan değerlerine bağlı olarak performansın genelde iyileştiği söylenebilir. Ortalama atak için atak büyüklüğünün yöntemimizin performansı açısından çok fazla etkili olmadığı gözlenmiştir. Genel performansı göstermek için *F1* ölçütünü de hesaplayıp Tablo 4'de gösterdik. Tablo 4'deki sonuçlara göre atak büyüklüğünden bağımsız olarak yöntemimizin ortalama atak profillerini başarıyla tespit ettiği gözlenmiştir. Rasgele atak profillerinin tespit edilme başarısı düşüktür.

Tablo 4. Değişen atak büyüklüğüne göre *F1* ölçütü değerlerinin değişimi

AB	3	5	10	15	25
RA	0,027	0,037	**0,041**	0,030	0,006
OA	0,851	0,887	0,821	**0,888**	0,842
SA	0,365	0,541	0,552	0,631	**0,906**
BA	0,236	0,411	0,741	0,922	**0,971**

Atak büyüklüğü yanında bir diğer kontrol parametresi doldurma büyüklüğüdür. Bu nedenle son olarak, doldurma büyüklüğünün değişen değerlerine göre yöntemimizin sınıflandırma başarısı incelenmiştir. Bunun için atak büyüklüğü 25 seçilip doldurma büyüklüğü 3 ile 25 arasında değiştirilmiştir. Rasgele karıştırmadan dolayı deneyler 100 defa tekrar edilip genel ortalamalar kesinlik ve bulma cinsinden Tablo 5'de sunulmuştur. **DB** doldurma büyüklüğünü temsil etmektedir.

Tablo 5. Değişen doldurma büyüklüğüne göre tespit etme başarısının değişimi

	Kesinlik					*Bulma*				
DB	3	5	10	15	25	3	5	10	15	25
RA	0,001	**0,007**	0,006	0,003	0,005	0,004	**0,022**	0,019	0,008	0,008
OA	**0,973**	0,965	0,951	0,936	0,911	**0,940**	0,923	0,882	0,852	0,792
SA	0,768	0,842	0,966	**0,987**	0,981	0,666	0,769	**0,851**	0,845	0,833
BA	**0,973**	0,973	0,966	0,956	0,950	**1,000**	0,999	0,993	0,986	0,988

Doldurma büyüklüğünün artan değerleriyle beraber atakların tespit edilme başarısı düşmektedir. Tablo 5'de listelenen sonuçlara göre genel olarak yöntemimizin başarısı küçük doldurma büyüklüklerinde daha iyi olarak gözlenmiştir. Büyük doldurma büyüklükleri atak profillerine daha fazla rasgele değer eklenmesi anlamına gelir. Bu ise tespit edilme başarısını olumsuz etkiler. En başarılı sonuçlar bölüm atağı için gözlenmiştir. Rasgele atak profillerinin tespit edilmesi önerilen algoritma başarılı çalışmamaktadır. Diğer ataklar için genelde

başarılı sonuçlar vermektedir. Genel başarıyı *F1* cinsinden göstermek için *F1* değerleri hesaplanıp tablo 6'da sunulmuştur. Tablo 5'de gözlenen benzer eğilimler Tablo 6'da da gözlenmiştir.

Tablo 6. Değişen doldurma büyüklüğüne göre *F1* ölçütü değerlerinin değişimi

DB	3	5	10	15	25
RA	0,002	**0,011**	0,009	0,004	0,006
OA	**0,957**	0,944	0,915	0,892	0,847
SA	0,713	0,804	0,905	**0,910**	0,901
BA	0,986	0,986	0,979	0,971	0,969

Bu çalışmada önerilen şilin atak tespit yöntemi Gunes ve Polat [11] tarafından önerilen hiyerarşik kümeleme-tabanlı yöntem ile karşılaştırılmıştır. Hiyerarşik kümeleme-tabanlı yöntem rasgele atakla üretilen şilin profillerini küçük doldurma büyüklüğünde daha başarılı olarak tespit etmektedir. Fakat doldurma büyüklüğü 25 seçildiğinde değişen atak büyüklüğüne göre başarılar karşılaştırıldığında bizim yöntemimizin nispeten daha başarılı olduğu görülmektedir. Ortalama atakla üretilen şilin profillerinin tespit edilmesi göz önüne alındığında, bizim algoritmamız daha başarılı sonuç vermektedir. Sürü ve bölüm ataklarla üretilen sahte profillerin tespit edilmesinde her iki algoritma da benzer başarılar göstermektedir. Genel olarak sürü ve bölüm atakları çok özel ataklar olduğundan, sahte profil tespit etme başarısı yüksek olmaktadır. Her iki yöntem de birbirine çok yakın başarı göstermektedir. Ama bizim algoritmamız çok küçük atak büyüklüklerinde daha başarılı çalışmaktadır.

6. Sonuçlar ve Gelecekteki Çalışmalar

Ortak filtreleme algoritmaları gibi mahremiyet-tabanlı ortak filtreleme algoritmaları da şilin ataklara maruz kalabilir. Şilin ataklar bazı hedef ürünlerin popülaritesini yükseltmek için tasarlanabilir. Bu manipülasyonlar müşterileri rahatsız edebilir. Bu nedenle şilin ataklarla sisteme eklenen sahte profillerin tespit edilmesi çok önemlidir.

Bu çalışmada maskelenmiş veri içeren mahremiyet-tabanlı ortak filtreleme sistemlerindeki sahte profilleri tespit etmek için ikili ayrıma *k*-ortalama kümeleme temelli bir şilin atak yöntemi tasarlanmıştır. Önerilen yöntem maskelenmiş veriden sahte profilleri tespit etmek için gerekli bilgileri hesaplayıp başarılı bir şekilde sahte profilleri tespit edebilmektedir. Özellikle sürü ve bölüm ataklarına karşı çok başarılı sonuçlar elde edilmiştir. Benzer şekilde algoritmamız ortalama atakla üretilen şilin profilleri de başarıyla tespit etmektedir. Fakat rasgele atağa karşı başarılı olduğu söylenemez. Değişik kontrol parametrelerin değişen değerlerine göre önerilen yöntemin genelde başarılı sonuçlar gösterdiği gözlenmiştir. Benzer yöntemle karşılaştırıldığında çoğu durumda daha başarılı çalıştığı görülmüştür.

Kümeleme-tabanlı tespit yöntemlerine ek olarak sınıflandırma-tabanlı yöntemler de başarılı sonuçlar verebilir. Bu nedenle planlanan bir çalışmamız sınıflandırma temelli şilin atak tespit etme metotları geliştirmek olarak adlandırılabilir. Bir diğer çalışma ise önerilen yöntemlerin değişik mahremiyet kontrol parametrelerinin değişen değerlerine göre başarılarının nasıl değiştiğinin incelenmesidir.

7. Kaynaklar

1. Goldberg, D., Nichols, D., Oki, B.M. ve Terry, D. (1992). Using collaborative filtering to weave an information Tapestry. *Communications of the ACM*, **35** (**12**): 61-70.

2. Herlocker, J.L., Konstan, J.A., Borchers, A. ve Riedl, J.T. (1999). An algorithmic framework for performing collaborative filtering. *The 22nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, Berkeley, CA, USA, 230-237.

3. Mobasher, B., Burke, R.D., Bhaumik, R. ve Williams, C.A. (2007). Towards trustworthy recommender systems: An analysis of attack models and algorithm robustness. *ACM Transactions on Internet Technology*, **7** (**4**): 23-60.

4. Gunes, I., Kaleli, C., Bilge, A. ve Polat, H. (2014). Shilling attacks against recommender systems: A comprehensive survey. *Artificial Intelligence Review*, **42** (**4**): 767-799.

5. Mobasher, B., Burke, R.D., Bhaumik, R. ve Sandvig, J.J. (2007). Attacks and remedies in collaborative recommendation. *IEEE Intelligent Systems*, **22** (**3**): 56-63.

6. Bilge, A., Kaleli, C., Yakut, I., Gunes, I. ve Polat, H. (2013). A survey of privacy-preserving collaborative filtering schemes. *International Journal of Software Engineering and Knowledge Engineering*, 23 (8): 1085-1108.

7. Ozturk, A. ve Polat, H. (2015). From existing trends to future trends in privacy-preserving collaborative filtering. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, **5** (**6**): 276-291.

8. Gunes, I., Bilge, A., Kaleli, C. ve Polat, H. (2013). Shilling attacks against privacy-preserving collaborative filtering. *Journal of Advanced Management Science*, **1** (**1**): 54-60.

9. Gunes, I., Bilge, A. ve Polat, H. (2013). Shilling attacks against memory-based privacy-preserving recommendation algorithms. *KSII Transactions on Internet and Information Systems*, **7** (**5**), 1272-1290.

10. Bilge, A., Gunes, I. ve Polat, H. (2013). A robust privacy-preserving recommendation algorithm. *The 2nd Asian Conference on Information Systems*, Phuket, Thailand.

11. Gunes, I. ve Polat, H. (2015). Hierarchical clustering-based shilling attack detection in private environments. *The 3rd International Symposium on Digital Forensics and Security*, Ankara, Turkey, 1-7.

12. Zhou, W., Wen, J., Koh, Y. S., Alam, S., ve Dobbie, G. (2014). Attack detection in recommender systems based on target item analysis. *The 2014 International Joint Conference on Neural Networks*, Beijing, China, 332-339.

13. Burke, R. D., Mobasher, B., Williams, C. A. ve Bhaumik, R. (2006). Classication features for attack detection in collaborative recommender systems. *The 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Philadelphia, PA, USA, 542-547.

14. Cao, J., Wu, Z., Mao, B. ve Zhang, Y. (2013). Shilling attack detection utilizing semi-supervised learning method for collaborative recommender system. *World Wide Web*, **16** (**5-6**): 729-748.

15. O'Mahony, M. P., Hurley, N. J. ve Silvestre, G. C. M. (2003). Collaborative filtering-safe and sound?. *Lecture Notes in Computer Science*, **2464**: 506-510.

16. Mehta, B. ve Nejdl, W. (2009). Unsupervised strategies for shilling detection and robust collaborative filtering. *User Modeling and User-Adapted Interaction*, **19** (**1-2**): 65-97.

17. Zhang, Z. ve Kulkarni, S. R. (2014). Detection of shilling attacks in recommender systems via spectral clustering, *The 17th International Conference on Information Fusion*, Salamanca, Spain, 1-8.

18. Bilge, A., Ozdemir, Z. ve Polat, H. (2014). A novel shilling attack detection method. *The 2nd International Conference on Information Technology and Quantitative Management*, Moscow, Russia, 165-174.

19. Chakraborty, P. ve Karforma, S. (2013). Detection of profile-injection attacks in recommender systems using outlier analysis. *Procedia Technology*, **10**: 963-969.

20. Polat, H. ve Du, W. (2005). Privacy-preserving collaborative filtering. *International Journal of Electronic Commerce*, **9** (**4**): 9-35.

21. Bilge, A. ve Polat, H. (2013). A scalable privacy-preserving recommendation scheme via bisecting k-means clustering. *Information Processing & Management*, **49** (**4**): 912-927.