

**KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

YÜKSEK LİSANS TEZİ

**EĞİTSEL VERİ MADENCİLİĞİNDE KULLANILMAK ÜZERE
EXPERIENCE API (XAPI) TEMELLİ ÖĞRENME DENEYİMİ
KAYITLARININ İŞLENEBİLMESİ İÇİN BİR MODEL
GELİŞTİRİLMESİ**

ERDEM KISMET

KOCAELİ 2018

KOCAELİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

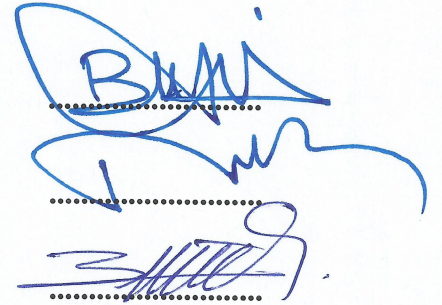
BİLGİSAYAR MÜHENDİSLİĞİ
ANABİLİM DALI

YÜKSEK LİSANS TEZİ

EĞİTSEL VERİ MADENCİLİĞİNDE KULLANILMAK ÜZERE
EXPERIENCE API (XAPI) TEMELLİ ÖĞRENME DENEYİMİ
KAYITLARININ İŞLENEBİLMESİ İÇİN BİR MODEL
GELİŞTİRİLMESİ

ERDEM KISMET

Yrd. Doç. Dr. Alpaslan Burak İNNER
Danışman, Kocaeli Üniversitesi
Prof. Dr. Nevcihan DURU
Jüri Üyesi, Kocaeli Üniversitesi
Yrd. Doç. Dr. Metin TOZ
Jüri Üyesi, Düzce Üniversitesi


.....
.....
.....

Tezin Savunulduğu Tarih: 08.01.2018

ÖNSÖZ VE TEŞEKKÜR

Bu tez çalışmasında, öğrenci başarısının analizinde öğrenme deneyimi kayıtları kullanılmış ve veri madenciliği yöntemlerinin öğrenme deneyimi kayıtları üzerine uygulanmasına ilişkin yeni bir model geliştirilmiştir.

Tez çalışmamın her aşamasında desteğini esirgemeyen, çalışmalarına yön veren, bana güvenen değerli danışmanım Yrd. Doç. Dr. Alpaslan Burak İNNER'e sonsuz teşekkürlerimi sunarım.

Ocak – 2018

Erdem KISMET

İÇİNDEKİLER

ÖNSÖZ VE TEŞEKKÜR	i
İÇİNDEKİLER	ii
ŞEKİLLER DİZİNİ.....	iv
TABLolar DİZİNİ	v
SİMGELEr VE KISALTMALAR DİZİNİ	vi
ÖZET.....	vii
ABSTRACT	viii
GİRİŞ	1
1. GENEL BİLGİLER.....	3
1.1. Çalışmanın Amacı.....	4
1.2. Önceki Çalışmalar.....	5
2. EĞİTSEL VERİ MADENCİLİĞİ.....	10
2.1. Eğitsel Ortamlar	11
2.2. Kullanılan Veri Kaynakları.....	11
2.2.1. Öğrenci bilgi sistemi	12
2.2.2. Öğrenme yönetim sistemleri	12
2.3. Veri Ön İşleme	13
2.4. Veri Madenciliği	14
2.5. Karar Ağacı Algoritmaları	15
2.5.1. CLS algoritması	16
2.5.2. ID3 algoritması.....	16
2.5.3. C4.5 ve C5.0 algoritmaları	16
2.5.4. CHAID algoritması	18
2.5.5. CART algoritması	18
2.5. Sonuçlarının Değerlendirilmesi	19
2.5.1. Doğruluk ve hata oranı.....	20
2.5.2. Kesinlik	20
2.5.3. Duyarlılık	20
2.5.4. F Ölçütü.....	21
3. ÖĞRENME DENEYİMİ KAYITLARININ YAKALANMASI.....	22
3.1. Experience Api.....	23
3.2. Learning Record Store	26
3.3. Deneyim İfadeleri	28
3.4. Experience Api ve LRS Kurulumu	32
3.5. Veri Toplama	33
3.5.1. Experience API kütüphaneleri.....	35
3.5.2. Mobil uygulama.....	36
3.5.3. Öğrenme yönetim sistemlerinin entegre edilmesi	37
4. ÖĞRENME DENEYİMİ KAYITLARI ÜZERİNDE EĞİTSEL VERİ MADENCİLİĞİ UYGULAMASI	38
4.1. Problemin Tanımlanması	39
4.2. Kullanılan Teknolojiler	39
4.3. Öğrenme Deneyimi Kayıtları Veri Seti ve Ön İşleme	39

4.4. Algoritma Seçimi ve Model Kurma.....	45
4.4.1. C4.5 algoritması	45
4.4.2. Gini algoritması	51
5. SONUÇLAR VE ÖNERİLER	53
KAYNAKLAR	55
EKLER	58
KİŞİSEL YAYIN VE ESERLER	71
ÖZGEÇMİŞ	72



ŞEKİLLER DİZİNİ

Şekil 2.1. Eğitsel veri madenciliği süreci	10
Şekil 3.1. ÖYS ile eski yaklaşım	25
Şekil 3.2. Experience API ile deneyim izlemede yeni yaklaşım	26
Şekil 3.3. Deneyim ifadelerinin XAPI'den LRS'ye gönderilmesi	27
Şekil 3.4. Deneyim ifadelerinin LRS üzerine kayıt edilmesi	27
Şekil 3.5. Simple statement	31
Şekil 3.6. Complex statement	32
Şekil 3.7. ADL LRS	33
Şekil 3.8. Verb: Initialized, simple statement	35
Şekil 3.9. Mobil uygulamada XAPI entegrasyonunda kullanılan kodlar	36
Şekil 3.10. Logstore XAPI modülü ayar ekranı	37
Şekil 4.1. Öğrenci performansına ilişkin EVM model diyagramı	38
Şekil 4.2. Toplanan öğrenme deneyimi kayıtlarının durumu	40
Şekil 4.3. Öğrenme deneyimi kayıtlarının XAPI üzerinden elde edilmesi	42
Şekil 4.4. Veri setinin eğitim ve test verisi olarak ikiye bölünmesi	44
Şekil 4.5. Modelin oluşturulması	45
Şekil 4.6. Modelin gösterimi	46
Şekil 4.7. Sınıflandırma sonucunda göre istatistiksel bilgi	46
Şekil 4.8. C4.5 ile elde edilen karar ağacı	47
Şekil 4.9. Karar ağacından elde edilen kurallar	48
Şekil 4.10. C4.5 karşılık matrisi	49
Şekil 4.11. Gini algoritması için modelin oluşturulması	51
Şekil 4.12. Gini algoritması için oluşturulan model	51
Şekil 4.13. Gini modeli karşılık matrisi	52

TABLULAR DİZİNİ

Tablo 2.1. Modelin sınıflandırma sonucunda oluşan karşılık matrisi	20
Tablo 3.1. XAPI Statement Veri Modeli	28
Tablo 3.2. Statement veri niteliklerinin açıklamaları	30
Tablo 3.3. XAPI fiilleri listesi örnekleri	31
Tablo 3.4. XAPI statement use case tablosu	34
Tablo 4.1. Veri setinde kullanılan XAPI fiilleri	41
Tablo 4.2. Veri seti özellikleri ve açıklamaları	43
Tablo 4.3. Veri seti örneği	44
Tablo 4.4. C4.5 model başarımı	49
Tablo 4.5. Önerilen davranışlar	50
Tablo 4.6. Gini algoritması model başarımı	52

SİMGELER VE KISALTMALAR DİZİNİ

Kisaltmalar

ADL	: Advanced Distributed Learning (Gelişmiş Dağıtık Öğrenme)
API	: Application Programming Interface (Uygulama Programlama Arayüzü)
CLS	: Concept Learning System (Kavram Öğrenme Sistemi)
EVM	: Eğitsel Veri Madenciliği
ID3	: Iterative Dichotomiser 3 (İteratif Bölücü 3)
LRS	: Learning Record Store (Öğrenme Kayıt Deposu)
MOOC	: Massive Open Online Course (Kitleli Açık Çevrimiçi Kurslar)
ÖBS	: Öğrenci Bilgi Sistemi
ÖSS	: Öğrenci Seçme Sınavı
ÖYS	: Öğrenme Yönetim Sistemi
SCORM	: Sharable Courseware Object Reference Model (Paylaşılabilir Eğitim Yazılımı Nesnesi Referans Modeli)
XAPI	: Experience API (Deneyim API'si)

EĞİTSEL VERİ MADENCİLİĞİNDE KULLANILMAK ÜZERE EXPERIENCE API (XAPI) TEMELLİ ÖĞRENME DENEYİMİ KAYITLARININ İŞLENEBİLMESİ İÇİN BİR MODEL GELİŞTİRİLMESİ

ÖZET

Eğitsel veri madenciliği, eğitimle ilgili olarak elde edilen verilere veri madenciliği teknikleri uygulayarak öğretim yöntemlerini ve öğrenme sürecini geliştirmeye yardımcı olmaya çalışan bir çalışma alanıdır. Bu alanda yapılan çalışmaların amacı öğrenci performansını ve öğrencinin eğitim çıktılarını geliştirmeye yardımcı olacak bilgileri keşfetmektir. Bu tez çalışmasında hangi öğrenci davranışının öğrenme üzerinde daha çok katkısı olduğunun araştırılabileceği bir model önerilmektedir. Bu nedenle bu çalışmada öğrencilerin davranışsal özellikleri olarak adlandırılan yeni bir özellik seti kullanılmaktadır. Bu özellikler öğrencilerin etkileşime girebileceği tüm fiziksel ve dijital eğitim ortamlarından elde edebileceği öğrenme deneyimleriyle ilgilidir. Öğrenme deneyimleri, bireylerin eğitim ortamlarında yaşadıkları öğrenme yaşantıdır. Bu çalışmada literatürdeki çalışmaların aksine öğretime etkisi olan parametre değil öğretimi iyileştirecek parametre üzerinde çalışılmaktadır. Bu nedenle öğrencilere ait demografik veriler bu çalışmada kullanılmamıştır. Bu çalışmada öğrenme deneyimlerinin Experience Api ile kaydedilebildiği bir sistem kurulmuştur. Öğrenme deneyimi kayıtlarına veri ön işleme adımları uygulanıp veri seti elde edilmiştir. Elde edilen veri seti sınıflandırma algoritmalarından Karar Ağacı ve Gini algoritmalarıyla sınıflandırılmıştır. Oluşturulan modelde hangi deneyim kaydının başarı üzerinde etkisinin daha fazla olduğu tespit edilebilmektedir. Test verisiyle oluşturulan model sırandığında %76 oranında doğru sonuç elde edildiği görülmektedir.

Anahtar Kelimeler: Eğitsel Veri Madenciliği, Experience Api, Karar Ağaçları, Öğrenme Deneyimi Kayıtları.

DEVELOPMENT OF A MODEL TO USE IN EDUCATIONAL DATA MINING TO PROCESS THE RECORDS OF EXPERIENCE API (XAPI) BASED ON LEARNING EXPERIENCE

ABSTRACT

Educational data mining is a field of study which is trying to help to improve teaching methods and learning process by applying data mining techniques to educational data which are acquired. The purpose of the studies is to discover the informations that provide to improve a learner's performance and educational outputs. In this thesis, a model is proposed that which student behavior may be investigated as a contribution to learning. A new feature set called behavioral characteristics of students is used in this study. These features relate to learning experiences that students can obtain from all the physical and digital learning environments in which they can interact. Learning experiences are individuals' learnings through experiencing in learning environments. In this thesis, contrary to the studies in the literature, the parameter which will improve the teaching is studied, not the parameter which has effect on teaching. On this purpose, demographic data belonging to the students has not been used. In this study, a system has been set up where learning experiences can be recorded via Experience API. Developed learning environments have been integrated into the Experience API. Data pre-processing steps have been applied to these learning experience records and the data set has been obtained. The obtained dataset is classified by algorithms classification; Decision Tree and Gini algorithms. In the created model, it can be determined which experience record has got more effect on achievement. When the model generated by the test data has been tried, 76% correct results have been obtained.

Keywords: Educational Data Mining, Experience Api, Decision Trees, Learning Record Store.

GİRİŞ

Teknolojinin gelişmesiyle birlikte birçok dijital eğitim ortamı ortaya çıkmıştır. Öğrenme yönetim sistemleri (ÖYS), mobil öğrenme yazılımları, kitlesel açık çevrimiçi kurs (MOOC) platformları, sanal seminerler, benzetimler, giyilebilir cihazlarla gerçekleştirilen öğrenme etkinlikleri, eğitsel oyun ve artırılmış gerçeklik uygulamaları teknolojinin gelişmesiyle ortaya çıkan, giderek gelişen öğrenme deneyimi yaşanabilecek dijital öğrenme ortamlarından sadece birkaç tanesidir.

Eğitim ortamlarındaki çeşitliliğin artması, eğitim ortamlarından elde edilen verilerin miktarını da artırmaktadır. Sürekli artan veri yığınları içerisinde anlamlı bilgiler elde edebilmek için veri madenciliği yöntemleri kullanılmaktadır. Eğitim alanında yapılan veri madenciliği çalışmaları ve uygulamaları veri yığınlarının çeşitlenmesine bağlı olarak artmaktadır.

Eğitim alanında yapılan veri madenciliği çalışmalarının amacı eğitim ortamlarının iyileştirilmesi ve öğrenci başarısının artırılmasını sağlamaktır. Bu doğrultuda öğrenci profili çıkartma, öğrenci başarısına etki eden faktörleri tespit etme, öğrenci destek sistemi, ders öneri sistemi gibi çalışmalar yapılmış ve eğitsel veri yığınları ve veri madenciliği yöntemleri kullanılmıştır.

Bu çalışmada bireyin akademik başarısını, akademik sürecin içindeyken arttıracak bir çözüm üretmek amaçlanmıştır. Bu nedenle öğrencilerin eğitsel ortamlardaki davranışlarının kaydedilmesi sağlanmış, sonra da öğrencilerin akademik başarısını akademik sürecin içerisindeyken artırabilecek davranışları önerebilecek bir eğitsel veri madenciliği modeli oluşturulmuştur. Öğrencilerin çevrimiçi ve çevrimdışı ortamlarda hangi deneyimleri tamamlaması hâlinde başarısını artırabileceğini ön görmeye çalışan bir model geliştirilmiştir.

Bu tez çalışması beş bölümden oluşmaktadır.

Birinci bölümde tez çalışması hakkında genel bilgi verilmiş, çalışmanın amacı ve önemi üzerinde durulmuştur. Ayrıca eğitsel veri madenciliği konusunda yapılan çalışmalardan örnekler verilmiş ve bu çalışmayla benzer hedeflere sahip literatürde bulunan diğer çalışmalardan bahsedilmiştir.

İkinci bölümde eğitsel veri madenciliği hakkında genel bilgiler verilmiş, bu alanda kullanılan yöntemler üzerinde durulmuştur.

Üçüncü bölümde öğrenme deneyimi kayıtlarının nasıl standartlara uygun ve merkezi bir şekilde tutulabileceği ve tez çalışmasında kullanılan teknolojiler hakkında bilgiler verilmiştir. Öğrenme deneyimi kayıtlarının tutulabileceği alt yapının kurulumu, fiziksel ve dijital öğrenme ortamlarının bu teknoloji ile nasıl bütünleştirilebileceği konusunda elde edilen deneyimler paylaşılmıştır.

Dördüncü bölümde, öğrenme deneyimi kayıtları üzerinde eğitsel veri madenciliği çalışmasının yapılabilmesi için geliştirilen model, kullanılan araçlar, veri setini oluşturma, ön işleme süreci ve veri madenciliği uygulanması üzerinde durulmuştur.

Beşinci bölümde elde edilen sonuçlar değerlendirilmiştir. Gelecekteki çalışmalarda nelere dikkat edilmesi gerektiği hususunda öneriler sunulmuştur

1. GENEL BİLGİLER

Öğrenme; farklı durumlarda, farklı ortamlarda gerçekleşir ve günümüzde fiziksel eğitim ortamlarıyla sınırlı değildir. Uzaktan eğitim, mobil öğrenme yazılımları, artırılmış gerçeklik uygulamaları, çevrimiçi kurslar ve daha birçok dijital ortam fiziksel ortamlarla birlikte kullanılarak öğrenmenin daha etkili bir şekilde gerçekleştirilebilmesini sağlamak amacıyla kullanılmaktadır.

Eğitim sürecindeki bir birey, çevrimiçi ve çevrimdışı ortamlarda öğrenmeler gerçekleştirir. Bu ortamlardaki davranışları ve deneyimleri doğrultusunda elde ettiği öğrenmeler sonucunda ulaşılan akademik başarıya etki eden birçok unsur bulunmaktadır.

Bireyin akademik başarısına etki eden faktörler, bugüne kadar pek çok araştırmanın konusu olmuştur. Literatürde bu konuda yapılan çalışmaların çoğunda bireyin, akademik başarısını etkileyen demografik özellikleri ele alınmıştır. Bu çalışmalarda; cinsiyet, ekonomik durum, anne-baba mesleği, yaş, bir önceki okulun türü gibi nitelikler araştırılmış ve bu faktörlerin bireyin akademik başarısı üzerindeki etkisi ortaya konulmuştur.

Bununla birlikte bu çalışmalar yalnızca durum tespitinde bulunmakta, bireyin başarısını arttırmaya yönelik herhangi bir çözüm önermemektedir. Yapılan çalışmalarda elde edilen bilgiler, yalnızca bireyin başarısına etki eden, geçmişten gelen faktörleri belirlemektedir. Elbette bu faktörlerin tespiti önemlidir ancak tespit edilen bu faktörler bireyin geçmiş yaşantısıyla ilgili olduğundan -bir önceki okul gibi- şimdiki öğrenme sürecine yönelik bir çözüm getirmemektedir. Elde edilen bu bilgi yalnızca başarısızlığı kestirilebilir öğrenciler için bir önlem alınmasını sağlayabilmektedir.

Öğrencilerin akademik başarısı, öğrenme miktarlarına bağlıdır. Öğrenmeler ise eğitsel ortamlardaki davranışlar sonucunda gerçekleşmektedir. Bu nedenle öğrencilerin başarısına etki eden ana faktör çevrimiçi ve çevrimdışı ortamlardaki davranışları ve deneyimleridir. Hangi eğitim ortamındaki öğrenme yaşantısının akademik başarıya daha fazla etki ettiği tespit edilebilirse öğrencilere hangi davranışı gerçekleştirmelerinin önemli olduğu söylenebilir.

Literatürde bugüne kadar yapılmış olan çalışmalarda veri seti olarak bireylere ait demografik bilgiler ve akademik geçmişine ait bilgiler kullanılırken bu çalışmada bireylerin öğrenme ortamlarından elde edilen öğrenme deneyimi kayıtları kullanılmaktadır.

1.1. Çalışmanın Amacı

Tezin amacı olan şu soruların cevapları eğitsel veri madenciliği yöntemleri ile incelenmiştir:

- Hangi öğrenme deneyimi ders başarısı üzerinde daha etkili olmuştur?
- Hangi öğrenme deneyimleri öğrencilere önerilirse dersi başarı ve öğrenme başarısı artar?
- Başarısız olma olasılığı olan öğrencileri öğrenme deneyimleri üzerinden tespit edilmesi mümkün müdür?

Bu çalışmayla birlikte öğrencilerin tüm ortamlardaki öğrenme deneyimlerinin izlenebileceği ve kayıt altına alınabileceği, standartlara uygun bir alt yapı kurulmuştur. Bu alt yapıya eğitsel ortamların nasıl entegre edilebileceği denenmiş ve öğrenme deneyimlerinin veri madenciliği yöntemlerine uygulanabilir formata dönüştürülmesi için yazılım geliştirilmiştir. Bu çalışmalar sonrasında akademik başarının hangi davranışlara bağlı olduğunun veri madenciliği teknikleriyle tespit edilmesini sağlayacak bir model geliştirilmiştir.

1.2. Önceki Çalışmalar

Literatürde eğitsel veri madenciliği üzerine yapılan birçok çalışma bulunmaktadır. Bu çalışmalar öğrencilerin öğrenme yöntemlerini sınıflandırmak, öğrenci profillerini çıkartmak, öğrencilerin e-öğrenme sistemleri üzerindeki davranışlarını sınıflandırmak, öğrencilere en iyi ders kombinasyonunu önermek, öğrencilerin başarısına etki eden faktörleri belirlemek gibi amaçlarla eğitim verileri üzerinde gerçekleştirilmektedir [1].

Öğrencinin genel başarısına etki eden faktörlerin veri madenciliği yöntemleriyle tespit edilmesine ilişkin birçok akademik çalışma bulunmaktadır.

Erdoğan (2004) K-means algoritması kullanarak yaptığı çalışmada, 2003 yılında Maltepe Üniversitesini kazanan öğrencilerin üniversite sınavı başarı puanları ile üniversitedeki akademik başarıları arasındaki ilişkiyi incelemiştir. Veri setlerinde; öğrencilere ait Öğrenci Seçme sınavı (ÖSS) puanının yüzdelik sırası, üniversitedeki derslerinden aldığı başarı notu, öğrencinin cinsiyeti, lise tipi, fakülte türü bilgileri bulunmaktadır. K-means algoritmasıyla oluşturduğu kümelerden elde ettikleri sonuca göre üniversite sınavı başarısıyla üniversitedeki ders başarısı arasında yüksek bir ilişki bulunmaktadır [2].

Gülçe (2010) Pamukkale Üniversitesine kayıtlı öğrencilere ait demografik bilgilerden yararlanarak veri madenciliği çalışması yapmıştır. Çalışmasının amacı ders başarısı düşük olabilecek öğrencilerin önceden belirlenmesini sağlamaktır. Çalışmada öğrenci bilgi sistemi üzerindeki cinsiyet, öğrenci kimlik bilgilerin, uygulanan anket sonucu, ailenin eğitim durumu, gelir durumu bilgilerini kullanarak akademik başarı tahminleri yapılmıştır. Bunlara ek olarak öğrencilerin ders başarılarının önceden tahmin edilmesi için çalışmalar yürütmüştür [3].

Ekim'in (2011) çalışmasında, Selçuk Üniversitesi mevcut öğrencilerine uygulanan anket verilerine apriori ve karar ağacı algoritmaları uygulanmıştır. Bu çalışmanın sonucunda öğrenci ebeveynlerinin eğitim seviyeleri ve gelir düzeylerinin, öğrenci başarısına en çok etki eden faktör olduğu sonucuna ulaşılmıştır [4].

Taşdemir'in (2012) çalışmasında, 2010 yılında Dicle Üniversitesini kazanan öğrencilere ait bilgiler üzerinde regresyon analizi yapılarak öğrenci başarısına etki eden faktörler araştırılmıştır. Yapılan çalışmada; öğrencilere ait not ortalaması, başarısız oldukları ders sayısı, fakülteleri, bölüme giriş yaşı, yerleştirme puanları, cinsiyetleri, ebeveynlerinin eğitim durumları gibi bilgilerden yararlanılmıştır. Birçok hipotez oluşturulmuş ve bu hipotezlere cevap üretilmeye çalışılmıştır. Örneğin "ÖSS puanının öğrencinin not ortalamasına etkisi vardır." hipotezi üzerine çalışıldığında ÖSS puanı ile öğrenci not ortalaması üzerinde yüksek bir ilişki olduğu tespit edilmiştir [5].

Saygılı'nın (2013) çalışmasında, Namık Kemal Üniversitesi Çorlu Mühendislik Fakültesi 2011 yılı sonu itibarıyla kayıtlı olan lisans öğrencilerine ait veriler üzerinde kümeleme yöntemleri uygulanmıştır. Bu çalışmada öğrencilere ait demografik veriler, ÖSYS puanları ve sıraları ile yerleştirildikleri bölümlerdeki genel başarı notları giriş verisi olarak kullanılarak bu bilgiler üzerinde anlamlı sonuçlar elde edilmeye çalışılmıştır. Kümeleme yöntemleri sonucunda elde ettikleri sonuçlara göre üniversite başarı notlarıyla, üniversitedeki genel notları arasındaki yüksek ilişki tespit edilmiştir. Ayrıca başarılı öğrencilerin hangi bölge ve hangi okul türünden geldiği tespit edilmiştir [6].

Kılınç (2015) çalışmasında, Eskişehir Osmangazi Üniversitesi Bilgisayar Mühendisliği Bölümü'nde okuyan öğrencilerin verilerini kullanarak üniversite öğrenci başarısı üzerinde etkisi olan faktörleri sınıflandırma (kNN ve j48) ve birliktelik kuralları (Apriori, Predictive Apriori) algoritmalarıyla incelemiştir. Yapılan çalışma sonucunda atılma politikasındaki değişiklik ile öğrencilerin genel başarıları arasında ilişki saptanmış, ayrıca öğrencilerin eğitim sürelerinin öğrencilerin maddi durumları ve annelerinin meslekleriyle olan ilişkisi tespit edilmiştir [7].

Güler'in (2017) çalışmasında, ortaokul eğitim başarısını etkileyen faktörler karar ağacı yöntemi ile sınıflandırılmıştır. Kahramanmaraş Dulkadiroğlu ilçesine bağlı bulunan 44 okuldan elde edilen verilerde öğrenci başarısına etki eden faktörler araştırılmıştır. Veri setinde kullanılan özellikler; öğrencilerin kitap okuma alışkanlığı, sayısal yetenek becerisi, bir odaya sahip olma durumu, kardeş sayısı, hastalık durumu, baba iş durumu, anne iş durumu, yaşadığı konum, bilgisayar bağımlılığı, televizyon bağımlılığı, baba

eđitim durumu, anne eđitim durumu, okul öncesi eđitim durumu, ailesi ile birliktelik durumu olarak belirlenmiřtir. Yapılan alıřma sonucunda bařarı durumuna etki eden en önemli deđiřken anne iř durumu olarak tespit edilmiřtir [8].

Yukarıda bahsedilen alıřmalar zet olarak řu zellikleri iermektedir:

- đrencilerin yalnızca genel bařarı durumuna etki eden faktrler incelenmiřtir.
- niversite sınavı yerleřtirme puanıyla, genel not durumu arasındaki yksek iliřki farklı alıřmalarda, farklı algoritmalarla tespit edilmiřtir.
- Hangi okul trnden, hangi cinsiyetten, hangi yařtan bařlayan, hangi blgeden gelen đrenci daha bařarılıdır gibi hipotezler incelenmiřtir.
- đrencinin ekonomik durumunun, bařarisında ve okulu bitirme sresinde etkisi incelenmiřtir.
- đrencilere ait demografik bilgiler, ekonomik durumunu gsterir bilgiler ve đrenci bilgi sistemi zerindeki bilgiler kullanılmıřtır.

Bu alıřmalarda bařarıya etki eden faktrleri belirlemek devam eden eđitim srecindeki bařarıyı artırıcı sonular retilebileceđi anlamına gelmemektedir. Zira ortaya ıkan faktrler gemiř yařantıya ait ve hatta bazıları -cinsiyet gibi- deđiřtirilemez niteliktedir.

Altınıřık'ın (2006) alıřmasında, Kocaeli niversitesi đrencilerinin ders seimi ařamasına yardımcı olabilmek amacıyla BS zerindeki ders bařarılarına ait bilgilerden yararlanılarak veri madenciliđi alıřması yapılmıřtır. Yapılan alıřmada Apriori algoritması kullanılarak đrencilerin bařarısız olduđu dersler arasındaki birliktelikler tespit edilmiřtir. Bu birliktelikler kullanılarak derslerin birbirleri ile bađıntıları ortaya konulmuř ve đrencilerin ders seiminde rehberlik edecek bir uygulama geliřtirilmiřtir [9].

Bresfelean ve arkadaşları (2008), alıřmalarında đrencilerin sınav bařarısızlıđını nceden kestirebilmek amacıyla đrenci profillerini oluřturmak iin veri madenciliđi sınıflandırma ve kmeleme tekniklerini uygulamıřlardır [10].

Yavuzalp'in (2012) çalışmasında, öğrencilerin süreç içerisinde öğrenme yöntemlerinde meydana gelen değişimleri algılayan ve bireysel farklılıklarına uygun öğrenme ortamları sunabilen sistemlerin altyapısında kullanılacak bir model geliştirilmiştir. Yapılan çalışmada, e-öğrenme ortamında kullanılan öğrenme yöntem ve stratejilerinin web kullanım madenciliği ile analiz edilmesi amaçlanmıştır. Çalışmada Yapay Sinir Ağları ve Karar Ağaçları algoritmalarından yararlanılmıştır. Çalışmanın sonucunda web kullanım davranışlarından öğrencilerin öğrenme yöntemleri %91,7 başarı ile öngörülelebilmektedir [11].

Özbay'ın (2015) çalışmasında öğrenme yönetim sistemleri üzerindeki öğrenci hareketliliğine ilişkin kayıtlar ile akademik başarı arasındaki ilişki veri madenciliği yöntemleriyle incelenmiştir. Kullanılan kümeleme algoritmalarının sonuçları incelendiğinde öğrenme yönetim sistemini aktif kullanmanın, öğrencilerin başarılarını artırdığı sonucuna ulaşılmıştır [12].

Üniversitelerde eğitsel veri madenciliği üzerinde yapılan çalışmaların birbirine benzer özellikler taşıdığı ve özellikle öğrenme yönetim sistemleri veya öğrenci bilgi sistemleri üzerindeki verilere yoğunlaştığı gözükmektedir. Bununla beraber öğrenme deneyimi kayıtlarının önemine ilişkinde çalışmalar da bulunmaktadır.

Amrieh ve arkadaşlarının (2015) yaptığı çalışmada öğrenci davranışları ile akademik başarı arasında kuvvetli bir ilişki olduğu ortaya konulmuştur. Ayrıca yaptıkları çalışmada veri setindeki davranışsal özelliklerin akademik başarının tespitine olan etkisini Artificial Neural Network (ANN), Decision Tree ve Naive Bayes algoritmalarıyla incelemiş ve bu davranışsal özelliklerin kullanılmaması durumuna göre %29 daha iyi sonuçlar ürettiğini tespit etmişlerdir [13].

Chakravarthy ve Raman'ın (2014) yaptıkları çalışmada, ÖYS'lerin Experience API ile birlikte ürettikleri deneyim ifadelerini üç farklı senaryo üzerinde değerlendirmiş ve deneyim verilerinin sınav başarısını kestirmeye yönelik ilişkisini ifade etmeye çalışmışlardır [14].

Özetle, veri madenciliđi tekniklerini kullanarak eđitim problemlerini çözmek için çeşitli arařtırmalar yapılmıřtır. Ancak çok az sayıda arařtırma, öđrencinin akademik başarısını artırıcı dođrudan çözümler önermektedir.

Bu çalıřmada, öđrencilerin tüm ortamlardaki öđrenme deneyimlerinin takip edilerek hangi deneyimleri gerçekleřtirmesi durumunda başarılı olabileceđini kestirmeye çalıřan bir model geliřtirilmiřtir. Ayrıca, elde edilen bilgiler, okulların, öđrencinin akademik başarısını artırmasına yardımcı olacaktır. Buna ek olarak, yöneticilerin öđrenme sistemlerini e-öđrenmedeki yeni geliřmelere ve standartlara uygun şekilde geliřtirmelerine yardımcı olacaktır.

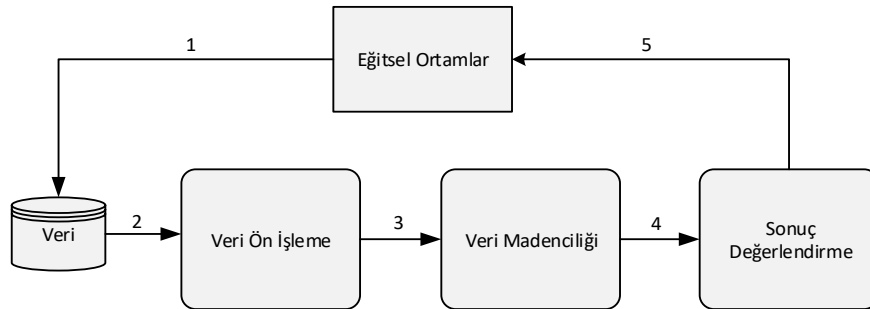


2. EĞİTSEL VERİ MADENCİLİĞİ

Eğitsel veri madenciliği (EVM), öğrencilere ilişkin büyük veri yığınlarında anlamlı örüntüleri keşfederek, faydalı olan verileri özetleyerek ve veriler arasındaki ilişkileri bularak öğrenci başarı ve başarısızlıklarının nedenlerinin tespit edilmesi ile ilgilenen bir disiplin olarak tanımlanmaktadır. Öğrenci başarısının artırılması, eğitim ortamlarındaki problemlerin tespit edilmesi ve daha etkili eğitim öğretim ortamlarının oluşturulması için eğitsel veri madenciliği kullanılmaktadır [6].

EVM alanında yapılan çalışmaların sayısı yeni veri yığınlarının oluşturulmasına bağlı olarak artmaktadır [1].

Eğitsel veri madenciliğine ilişkin süreç Şekil 2.1’de verilmiştir. Klasik bir EVM sürecinde eğitsel ortamlardan veri tabanlarına öğrencilere ait veriler kaydedilir (1), veriler ön işleme sürecine alınır (2), veri madenciliği algoritmasına uygun hâle getirilen veriler üzerinde veri madenciliği algoritmaları çalıştırılır (3), elde edilen sonuçlar değerlendirilir (4), üretilen sonuçlardan yararlanılarak eğitsel ortamların veya eğitsel süreçlerin iyileştirilmesi amaçlanır (5).



Şekil 2.1 Eğitsel veri madenciliği süreci

Veri madenciliği tekniklerinin kalitesi, toplanan verilere ve özelliklere bağlıdır. Öğrenci başarısını artırmak veya eğitsel ortamların iyileştirilmesini sağlamak için veriler doğru kaynaklardan amaca yönelik olarak elde edilmelidir. Eğitsel veri madenciliğinde kullanılan veriler eğitsel ortamlardan elde edilen öğrencilere ait verilerdir.

2.1. Eğitsel Ortamlar

Eğitsel ortamlar öğrenme-öğretme sürecinin olduğu, bireylerin kendi hazır bulunuşluk durumları, ilgi ve yetenekleri doğrultusunda katılımda buldukları, kişiler arası etkileşimin bulunduğu ve öğrenme yaşantılarının olduğu ortamlardır.

Bu veriler elde edildikleri eğitsel ortamların türüne göre çevrimdışı ortamlar ve çevrimiçi dijital ortamlar olarak iki sınıf altında değerlendirilebilmektedir.

Çevrimdışı fiziksel ortamların aksine çevrimiçi dijital öğrenme ortamları öğrencilerin öğrenme aktivitelerine ilişkin verileri çok detaylı bir şekilde veri tabanlarına kaydedebilmektedir. Bu nedenle EVM’de daha çok çevrimiçi öğrenme ortamlarının verilerinin kullanılmaktadır. Bu veriler; kullanıcılara ait log kayıtları, etkileşim verileri, ölçek ve kontrollü deney verilerini içermektedir.

EVM’de sıklıkla veri elde edilen çevrimiçi ortamlara; uzaktan eğitim sistemleri, öğrenme yönetim sistemleri ve öğrenci bilgi sistemleri örnek olarak verilebilir.

Günümüzde çevrimiçi eğitsel ortamların çeşitliliği artmaktadır. Mobil öğrenme yazılımları, MOOC (Massive Open Online Courses) platformlar, sanal seminerler, benzetimler, giyilebilir cihazlarla gerçekleştirilen öğrenme etkinlikleri, eğitsel oyun, artırılmış ve sanal gerçeklik uygulamaları gibi ortamlar çevrimiçi ortam olarak kullanılmaktadır. Her yeni ortam, öğrencilere ait yeni veri yığınlarının ortaya çıkmasına neden olmaktadır.

Çevrimdışı fiziksel eğitim ortamlarına ait verilerin veri tabanlarında saklanması ile ilgili sınırlılıklar e-öğrenme alanındaki yeniliklerle birlikte ortadan kalkmaktadır [15].

2.2. Kullanılan Veri Kaynakları

Literatür araştırması bölümünde verilen eğitsel veri madenciliği çalışmalarında öğrencilere uygulanan anket verileri, öğrenme yönetim sistemi üzerindeki etkileşim verileri ve öğrenci bilgi sistemi üzerindeki akademik başarıya ilişkin verilerle birlikte demografik veriler sıklıkla veri yığını olarak tercih edilmiştir.

Literatürde bulunan çalışmalarda veri kaynakları tek başlarına kullanılabilirdiği gibi birbirlerini destekleyici olarak kullanılabilirdiği görülmektedir. Saygılı'nın (2013) çalışmasında, öğrenci bilgi sisteminden elde edilen mevcut veri setine katkı sağlaması için Bilgisayar Mühendisliği bölümü öğrencilerine ayrıca bir anket uygulanmış ve bu anket verileri üzerinde veri madenciliği çalışmaları gerçekleştirilmiştir. [6].

2.2.1. Öğrenci bilgi sistemleri

Öğrenci bilgi sistemleri (ÖBS), öğrenciler hakkında detaylı bilgiler içerebilen ve seçtikleri derslere ilişkin notlarının depolandığı veri tabanlarına sahiptir. Öğrenci bilgi sistemi üzerinde tutulan veriler EVM çalışmalarında sıklıkla kullanılmaktadır.

ÖBS'ler üzerindeki verilere ilişkin bazı sınırlılıklar bulunmaktadır. Bu sınırlılıklar şu şekilde sıralanabilir:

- ÖBS'ler öğrenme aktiviteleri veya deneyimlerinin tamamını takip edemezler.
- Öğrenci kişisel bilgileri dışında öğretime ilişkin olarak “sınav notları”, “ödev notları”, “proje notları”, “uygulama notları” gibi bireylerin yalnızca not bilgilerini depolanmaktadır.
- Genellikle eğitim kurumunun kullandığı öğrenci bilgi sistemleri benzer bilgileri veri tabanlarında saklamaktadır ancak standart bir veri yapısına sahip değildir.
- Veriler farklı formatlarda, farklı veri tabanı teknolojisinin üzerinde saklanmaktadır.

ÖBS veri tabanları üzerinde birçok bilgi keşfi çalışması yapılmıştır ve bu çalışmalar literatür taraması bölümünde paylaşılmıştır.

2.2.2. Öğrenme yönetim sistemleri

Öğrenme yönetim sistemleri (ÖYS) gelişmiş loglama sistemleri sayesinde bireylere ait detaylı davranış kayıtlarını yine kendi veri tabanlarında saklamaktadırlar. Bireyin sisteme ne zaman giriş yaptığı, hangi materyalle ne kadar süreyle etkileşime girdiği, hangi soruya kaç saniyede ne cevap verdiği, hangi videoyu ne kadar süreyle izlediği ve daha çok bilgiyi ÖYS'ler takip edebilmektedirler.

ÖYS'ler gelişmiş öğrenci raporlama özelliğine sahiptir ve bu raporlamalar öğrencilerin sadece ÖYS üzerindeki öğrenme etkinliklerinden oluşmaktadır.

Öğrenme aktivitelerinin gerçekleştiği ortamların çeşitliliği giderek artmaktadır. Örneğin mobil öğrenme yazılımları, ÖYS haricindeki eğitsel içerik sunan web siteleri, artırılmış gerçeklik uygulamaları veya benzetimler bireylerin uygulama içerisindeki davranışlarını bir veri yığını oluşturabilecek şekilde saklayabilmektedirler. Teknolojideki ilerlemeler eğitsel veri madenciliğindeki veri kaynaklarının çeşitlenmesini sağlamaktadır.

ÖYS üzerindeki verilere ilişkin sınırlılıklar şunlardır:

- Bireylerin etkileşime gireceği tüm eğitim materyalleri ÖYS üzerine yüklenemeyecek kadar çeşitliliğe sahiptir.
- Bireyin dijital dünyadaki tüm öğrenme deneyimlerini izlemek için alt yapı sunmaz. Sadece kendi üzerindeki davranışları takip edebilir.
- ÖYS logları fiziksel ortamlardaki öğrenme deneyimlerini takip edemezler.

ÖYS logları üzerinde birçok bilgi keşfi çalışması yapılmıştır ve bu çalışmalar literatür taraması bölümünde paylaşılmıştır.

2.3. Veri Ön İşleme

Veri madenciliği çalışmalarında olduğu gibi veri kaynaklarından toplanan veriler algoritmalara uygulanmadan önce bir ön işleme yapılmalı ve algoritmalar için hazır hâle getirilmelidir. Yapılacak ön işlemeyle veri setinin kalitesi artacak böylelikle seçilen yöntem için en uygun özelliğe ulaşılabilecektir.

Ön işleme aşaması veri madenciliği sürecinde yer alan verilerin hazırlanması aşamasıyla birebir benzerdir. Veri ön işleme aşamasında veri temizleme, veri dönüştürme, veri birleştirme, özellik seçme işlemleri gerçekleştirilir.

Veri temizleme sürecinde sonuca etkisi merak edilmeyen, problem ile ilgisiz verilerin silinmesi, kayıp ya da aykırı gözlemlerin tespit edilmesi işlemleri yapılmaktadır. Veri tabanlarında yer alan tutarsız ve hatalı veriler gürültü olarak değerlendirilmektedir.

Veri dönüştürme sürecinde veri madenciliği algoritmasına doğrudan katmanın uygun olmayacağı veriler bir dönüşüm yöntemi uygulanarak verinin normalleştirilmesi veya standartlaştırılması sağlanır.

Veri birleřtirme/bütünleřtirme süreci, farklı kaynaklardan elde edilen verilerin tek bir veri seti hâline dönüřtürülmesidir. Örnek olarak, anket ile elde edilen verilerin öđrenci bilgi sistemi verileriyle birlikte kullanılabilmesi için tek bir dosyada toplanması gösterilebilir.

Özellik seçme sürecinde, ilgisiz veya fazla deđişkenler veri setinden kaldırılarak, analizde kullanılacak deđişkenlerin seçilmesi için çalışılır. Bu aşamada karar ağacı gibi sınıflandırma algoritmalarından sıklıkla yararlanılmaktadır.

2.4. Veri Madenciliđi

Veri toplama ve veri ön işleme aşamasından sonra elde edilen veri seti, veri madenciliđi yöntemlerine uygulanır. Seçilen probleme uygun veri madenciliđi yöntemleri kullanılmalıdır. Eğitsel veri madenciliđi çalışmalarında kümeleme, sınıflandırma, regresyon gibi yöntemler oldukça fazla tercih edilmektedir [1].

Veri madenciliđinde tahminde bulunabilmek için bir model geliřtirmek için sınıflandırma veya regresyon kullanılmaktadır. Bir sınıflandırma veya regresyon işleminde, modeli oluşturmak için eğitim kayıtları (training records) kullanılmaktadır. Her bir kayıt için birçok öznitelik (attribute) bulunmaktadır. Tanım kümesi sayısal olmayan öznitelikler kategorik, sayısal olan özniteliklere ise sayısal özniteliklerdir. Özniteliklerden bir tanesi Bađımlı (dependent) niteliklerdir. Diğer öznitelikler belirleyici (predictor) nitelik olarak adlandırılır. Eğer bađımlı öznitelik kategorik bir deđer ise problem bir sınıflandırma problemidir. Bu durumda bađımlı öznitelik sınıf etiketi (class label) olarak adlandırılır. Eğer bađımlı nitelik sayısal bir deđer ise, problem regresyon problemidir.

Bu çalışmanın problemine uygun yöntem, sınıflandırma yöntemi olduđu için tezin bu bölümünde sınıflandırma algoritmalarından karar ağacı öğrenimi açıklanmaktadır.

2.5. Karar Ağacı Algoritmaları

Karar ağaçları sınıflandırma problemlerinde kullanılan en popüler yaklaşımlardan biri olarak kabul edilir.

Karar ağaçları düğümlerden ve düğümleri oluşturan bağlantılardan oluşan hiyerarşik bir yapıdır. Karar ağaçları kolay yorumlanabilir ve anlaşılabilirliği yüksektir. Karar ağaçlarının sıklıkla tercih edilmesine sebep olan avantajlara, oluşturmanın düşük maliyetli olması, verilere uygulanmasının kolay olması ve güvenilirlikleri yüksek olması gibi avantajlar örnek olarak verilebilir.

Karar ağacı algoritmaları verilerin ortak özelliklerine göre ayrıştırılmasını sağlar. Sınıflandırma işlemi bir öğrenme algoritmasına dayanır. Bu nedenle verilerin bir kısmı sınıflandırma modelinin oluşturulması amacıyla modeli eğitmek için kullanılır. Verilerin diğer kısmı da oluşturulan modelin doğruluğunu test etmek için kullanılır. Oluşturulan modelin test verilerini sınıflandırma doğruluğu kabul edilebilir bir oranda olan model yeni verilerin sınıflandırılması amacıyla kullanılabilir.

Eğitim verisindeki hangi özelliklerin karar ağacı modelini oluştururken kullanılacağını belirlemek ve kök düğümden itibaren bölünmenin hangi kısıta göre yapılacağı belirlemek için farklı karar ağacı algoritmaları kullanılmaktadır. En iyi bölünmeyi gerçekleştirecek özneliğin seçilmesi için kategorik değişkenlerde Entropi Endeksi, Gini Endeksi, Twoning, Sınıflandırma hatası endeksi gibi yöntemler kullanılmaktadır. Sürekli değişkenlerde ise En Küçük Karalar Sapması yöntemi kullanılan yöntemlerden biridir.

Karar ağaçları yardımıyla sınıflandırma işlemlerini yerine getirmek için birçok algoritma geliştirilmiştir. Geliştirilen algoritmalar kendi içlerinde, kronolojik olarak bir öncekinin eskikliklerini tamamladığından algoritmaları gruplandırabilmek mümkündür. En büyük gruplara örnek olarak CLS ve AID grupları verilebilir. CLS algoritma grubunda CLS (Concept Learning System), ID3 (Iterative Dichotomiser 3), C4.5 ve C5.0 algoritmaları bulunmaktadır. AID algoritma grubunda ise AID, CHAID, Exhaustive CHAID ve CART algoritmaları bulunmaktadır.

2.5.1. CLS Algoritması

CLS birçok karar ağacı yönteminin temel taşı olarak kabul edilir ve Hoveland ve Hunt tarafından 1959 yılında yayınlanmıştır. Sadece sınıflandırma yapan bir karar ağacı üretilmekte ve eksik sınıflandırma maliyetleri göz önüne alınmaktadır.

2.5.2. ID3 algoritması

ID3 algoritması Ross Quinlan tarafından 1986 yılında yayınlanmıştır. Entropiye dayalı bölünmeyi kullanır ve daha önce bölen olarak kullanılmamış öznitelikleri kullanarak ağacın yapraklara doğru homojenleşmesi için çalışır. Tüm özniteliklerin tüketilmesi ve seçilen özneliğin kategorisinde tanımlanan krtere uygun gözlemlerin tüketilmesi durumunda ağaç oluşumu sona erer. Bu algorithmada sayısal öznitelikler kullanılamamaktadır.

2.5.3. C4.5 ve C5.0 algoritmaları

C4.5 algoritması, en yaygın kullanılan karar ağacı algoritmalarından biridir ve Ross Quinlan'ın ID3 algoritmasının geliştirilmiş halidir. ID3 algoritması sayısal değerlere sahip nitelikler söz konusu olduğunda kullanılamamaktadır. Bu nedenle ID3 algoritmasının geliştirilmiş hali C4.5 algoritması sayısal değerlere sahip nitelikler söz konusu olduğunda tercih edilmektedir. C4.5 algoritması sürekli öznelik değerlerini işleyebilmek amacı ile bir eşik değer hesaplar, eşik değer üstünde olanlar ve eşik değer altında olanlar şeklinde veri dizisini böler. C4.5 algoritmasında en iyi bölen öznelik seçiminde entropi ve enformasyon kazancı teknikleri kullanılmaktadır. ID3 algoritmasından farklı olarak veri dizisinde eksik değerler bulunması durumunda bu değerler için entropi ve enformasyon kazancı hesaplamaları yapılmaz [16].

C4.5 algoritması öncelikle hedef nitelik için entropi değerini hesaplar. Entropi bir sistemdeki belirsizliğin ölçüsüdür. Bir özneliğin kategorilerinin entropi değerleri $H(t)$, bit cinsinden iki temelli logaritma kullanılarak Denklem 2.1'de belirtildiği şekilde hesaplanır.

$$H(T) = -\sum_{i=1}^m p(i | t) * \log_2 p(i | t) \quad (2.1)$$

Denklem 2.1’de verilen eşitlikte veri dizisindeki toplam gözlem sayısı t ile, ilgili gözlem sayısının i ile, toplam kategori sayısı m ile, ilgili kategorideki gözlem sayısının toplam gözlem sayısına olan $p(i|t)$ ile temsil edilmektedir. Bütün örnekler aynı sınıfa ait ise entropi $H(T) = 0$ olarak hesaplanır. Bütün örnekler sınıfa eşit olarak dağıtılmışsa entropi $H(T) = 1$ olarak hesaplanır.

Daha sonra C4.5 algoritması her bir tahmin edici değişken için entropi değerini Denklem 2.2’de belirtildiği şekilde hesaplar.

$$H(X, T) = -\sum_{i=1}^n p(i | t) * H(T_i) \quad (2.2)$$

Bunun ardından her bir tahmin edici değişkenin bilgi kazanımını algoritma tarafından Denklem 2.3’te belirtildiği şekilde hesaplanır. Seçilen özniteliğin entropi değeriyle her bir özniteliğin entropi değerleri arasındaki fark bilgi kazancını (information gain) olarak isimlendirilir.

$$\text{Kazanç}(X, T) = H(T) - H(X, T) \quad (2.3)$$

Bu hesaplamaların amacı en yüksek bilgi kazanımı sağlayan tahmin edici sınıfı tespit etmektir. En yüksek bilgi kazanımı sağlayan tahmin edici nitelik tespit edilir ve ağaç bu nitelikten itibaren dallandırılmaya başlanır. Algoritma sonraki dallanmalar için düğümün sınıflarından oluşan alt kümelerde entropi hesabı, bilgi ve bilgi kazanımı hesabı yaparak en yüksek bilgi kazancı sağlayan düğümü bulurak kırılmaya o düğümde devam eder.

Oluşturulan karar ağaçlarında sınıflamaya hiçbir etkisi olmayan dalların ağaçtan çıkarılması işlemi budama olarak adlandırılmaktadır. Budama işlemi ağaç kurulurken veya kurulduktan sonra da yapılabilir. Budama işlemi yapılırken hangi ölçütün baz alınacağı önemlidir.

C5.0 algoritması ise C4.5’in geliştirilmiş hali olup, özellikle büyük veri setleri için kullanılmaktadır. C5.0 algoritması Unix / Linux için See5 algoritması ise Windows için geliştirmiş ve 1994 yılında ticari olarak piyasaya sürülmüştür. C5.0 algoritması daha etkili bir bellek kullanımına sahip ve C4.5’e göre daha hızlı bir algoritmadır.

2.5.4. CHAID algoritması

AID algoritması, 1971 yılında Sonquist, Baker ve Morgan tarafından sunulan AID grubu algoritmalarının ilkidir. 1980 yılında Gordon V. Kass tarafından geliştirilen CHAID algoritması bu grubun en popüler karar ağacı algoritmalarından biri olmuştur. CHAID algoritması ile oluşturulan karar ağacı, her hiyerarşik seviyede ikiden fazla dala ayrılabilir. Algoritmanın avantajları arasında bütün ölçü skalaları ile çalışabilir olması, nesnelere ağırlık ve frekans değerlerinin atanabilmesi, eksik değerleri ayrı bir kategoride analiz edebilmesi örnek olarak verilebilir.

2.5.5. CART algoritması

CART (Classification and Regression Trees) algoritması Breiman ve Friedman tarafından 1984 yılında önerilmiştir. Morgan ve Sonquist'in AID (Automatic Interaction Detection) adlı karar ağacı algoritmasının devamı niteliğindedir. CART algoritması, sınıflandırma ve regresyon analizi yapmak için kullanılan algoritmalarından biridir. ID3 ve C4.5 gibi karar ağacı oluşturulması esasına dayanır. C4.5 ağacında kırılma niteliğinin sahip olduğu sınıf kadar olabilirken CART algoritmasında ikili biçimde kırılım esastır. CART algoritmasında kök düğümden başlayarak her seviyede iki alt düğüm olur. Amaç bir sonraki seviyede daha homojen hale getirilmiş düğümleri elde edebilmektir. Oluşan ağaç sınıflandırma ağacı ve aynı zamanda da regresyon ağacı olmaktadır. CART algoritması kategorik ve sürekli değişkenlerle çalışabilmektedir. Kategorik değerler için düğümlerden alt düğümlere dallanırken bölünmeyi gerçekleştirecek özneliği tespit etmek için Gini, Twoning ve Ordered Twoning kriterlerini; sürekli değişkenlerin bölen noktalarının bulunmasında ise en küçük kareler sapması kriterini kullanmaktadır. CART ağacı, yeni bir bölünmenin gerçekleşmeyeceği duruma kadar sürekli bölünerek oluşmaktadır. Son veya uç olmayan her bir düğümden iki adet dal bulunmaktadır. Ağaç oluştuktan sonra uçtan köke doğru budama işlemi yapılmaktadır. Her budama işlemi sonrası bir test verisi ile değerlendirme yapılarak olası en başarılı karar ağacı tespit edilmeye çalışılmaktadır.

Sınıflandırma ve regresyon ağacı oluşturma için kullanılan Gini algoritmasında düğümlerin iki alt dalı olabilir. Bu algoritmada ağacın kırılımlarını oluşturmak için Gini indeksi hesaplanır ve sol bölünmeler için $Gini_{sol}$, sağ bölünmeler için $Gini_{sağ}$ değerleri hesaplanır.

K sınıfının bulunduğu T düğümdeki örnekleri L_i sol tarafta i kategorisindeki örneklerin sayısını, R_i sağ taraftaki i kategorisindeki örneklerin sayısını göstermek üzere Denklem 2.4 ve 2.5'te verilen bağıntılar kullanılır.

$$\text{Gini}_{\text{sol}} = 1 - \sum_{i=1}^k \frac{L_i^2}{T_{\text{sol}}} \quad (2.4)$$

$$\text{Gini}_{\text{sağ}} = 1 - \sum_{i=1}^k \frac{R_i^2}{T_{\text{sağ}}} \quad (2.5)$$

Bu işlemlerin ardından her j niteliği için, n satır sayısı olmak üzere Denklem 2.6'da verilen bağıntı yardımıyla hesaplanır.

$$\text{Gini}_j = \frac{1}{n} - (T_{\text{sol}} \text{Gini}_{\text{sol}} + T_{\text{sağ}} \text{Gini}_{\text{sağ}}) \quad (2.6)$$

Elde edilen Gini_j değerleri arasından en küçük olanı seçilir ve bölünme bu niteliğe göre olur.

2.5. Sonuçların Değerlendirilmesi

Veri ön işleme, parametre seçimi ve test kümesi seçimi veri madenciliği uygulamasında ortaya çıkacak olan modelin başarımını etkiler. Eğitim ortamlarını ve eğitsel süreçleri iyileştirmek için veri madenciliği sonucunda ortaya çıkan bilgi, örüntü veya modellerin başarımları değerlendirilmelidir.

Model başarımını değerlendirirken kullanılan temel kavramlar; hata oranı, kesinlik, duyarlılık ve F-ölçütüdür. Modelin başarısı, modelin test verileriyle sınanması sırasında test verisinin doğru sınıfa atanan örnek sayısı ve yanlış sınıfa atılan örnek sayısı nicelikleriyle alakalıdır.

Tablo 2.1 'de verilen karşılık matrisinde satırlar test kümesindeki örneklere ait gerçek sayıları, kolonlar ise modelin tahminlemesini ifade eder. Modelin doğru sınıflandırdığı sınıf tahmininde doğru tespit edilen sonuçları için True Positive (TP) ve True Negative (TN) olarak, yanlış tespit edilen sınıf tahminleri için ise False Positive (FP) ve False Negative (FN) olarak belirtilmektedir.

Tablo 2.1 Modelin sınıflandırma sonucunda oluşan karşılık matrisi

		Sınıfın Tahmini	
		Sınıf = 1	Sınıf = 0
Doğru Sınıflandırma	Sınıf = 1	True Positive (TP)	False Negative (FN)
	Sınıf = 0	False Positive (FP)	True Negative (TN)

2.5.1. Doğruluk ve hata oranı

Model başarımının ölçülmesi için kullanılan en popüler ve basit yöntem model için hesaplanan doğruluk oranıdır. Eğitim verileriyle oluşturulan model test verileriyle sınanır ve oluşturulan karşılık matrisinden doğru sınıflandırılmış örnek sayısının (TP + FP) tüm örnek sayısına oranı (TP+FP+FN+TN) doğruluk oranını verir. Doğruluk oranı Denklem 2.7’de verilen ifade ile hesaplanabilir.

$$\text{Doğruluk} = \frac{TP + TN}{TP + FP + FN + TN} \quad (2.7)$$

Hata oranı doğruluk oranını 1’e tamamlayan değerdir.

2.5.2. Kesinlik

Denklem 2.8’de gösterildiği gibi kesinlik değeri sınıfı 1 olarak tahminlenmiş TP sayısının, sınıfı doğru olarak tahminlenmiş tüm örnek sayısına oranıdır (TP + FP).

$$\text{Kesinlik} = \frac{TP}{TP + FP} \quad (2.8)$$

2.5.3. Duyarlılık

Denklem 2.9’da gösterildiği gibi doğru sınıflandırılmış pozitif örnek sayısının (TP), toplam pozitif örnek sayısına oranıdır (TP + FN).

$$\text{Duyarlılık} = \frac{TP}{TP + FN} \quad (2.9)$$

2.5.4. F ölçütü

Kesinlik ve duyarlılık ölçütleri tek başına bir karşılaştırma sonucu çıkartmak için yeterli değildir. Her iki ölçütün bir arada değerlendirildiği F ölçütü modeli değerlendirmek için daha iyi bir sonuç vermektedir. Kesinlik ve duyarlılık değerlerinin harmonik ortalaması Denklem 2.10'da gösterildiği gibi F-ölçütünü verir.

$$F\text{-Ölçütü} = \frac{2 \times \text{Duyarlılık} \times \text{Kesinlik}}{\text{Duyarlılık} + \text{Kesinlik}} \quad (2.10)$$

3. ÖĞRENME DENEYİMİ KAYITLARININ YAKALANMASI

Bireylere yeni bilgi, davranış, beceri veya değer kazandıran ya da var olanları değiştiren veya güçlendiren deneyimler öğrenme deneyimleridir [17].

Bireylerin eğitim ortamlarındaki öğrenme yaşantıları bireylere ait öğrenme deneyimleridir. Eğitim sürecindeki bir birey, çevrimiçi ve çevrimdışı ortamlarda öğrenmeler gerçekleştirir.

Çevrimiçi ve çevrimdışı öğrenme ortamları bir eğitim planı ile birleştirilerek bir arada sunulduğunda öğretimin kalitesini artırmakta ve bireyler üzerinde öğrenmeyi daha etkili bir şekilde gerçekleştirebilmektedir. Ancak tüm bu farklı ortamların birlikte işe koşulması bireylerin öğrenme sürecini takip etmeyle ilgili birtakım problemler ortaya çıkartmaktadır. Öğrenenin ortam ile girdiği etkileşimleriyle oluşan öğrenme deneyimlerinin kayıt altına alınması ve takip edilmesi bireyin performansı hakkında çıkarım yapabilmek için gereklidir.

Dijital öğrenme ortamları bireye ait öğrenme deneyimlerini kendi içerisine kayıt altına alabilecek şekilde tasarlanabilmektedir. Örnek olarak öğrenme yönetim sistemleri (ÖYS) verilebilir. Bu sistemler öğrencilerle ilgili çok miktarda bilgiyi takip etme ve biriktirme kabiliyetine sahiptir. ÖYS'ler üzerinde bulunan günlükler öğrenciler tarafından gerçekleştirilen birçok etkinliği ve öğrencilerin elde ettiği deneyimleri izler. Birçok ÖYS yazılımı, bireylerin sistem üzerinde gerçekleştirdiği belirli bir içeriği okumak, değerlendirme sorularına cevap vermek, kursta sağlanan çeşitli görevleri yerine getirmek gibi davranışları ayrıntılı olarak kayıt altına alabilmektedir. Ancak her dijital ortam birey davranışlarını kendi içerisinde kendi standartlarıyla ayrı ayrı izlemekte ve kaydetmektedir.

Bireyin gerçek başarısı tüm ortamlardaki deneyimlerinin bileşkesidir. Dijital ortamların kaydettiği deneyim verileri birlikte analiz etmek için standart bir formata sahip değildir. Bu nedenle bireyin hangi ortamla ne kadar etkileşimde bulunduğu standart ve merkezi bir şekilde kaydedilerek takip edilebilmesi gerekmektedir. Ancak standart bir formatta tutulan öğrenme deneyimi kayıtlarında yapılacak çalışmada hangi öğrenme deneyiminin akademik başarı üzerinde ne kadar bir etkiye sahip olduğunun tespiti yapılabilecektir.

Ayrıca bu çalışmanın da konusu olan eğitsel veri madenciliği çalışmalarında karşılaşılan problemlerden biri büyük boyutlu ama aynı düzene sahip olmayan verilerin içinden işe yarar bilgilerinin çekilmesinin zor olmasıdır. Eğitsel veri madenciliği için öğrenme deneyimlerinin aynı düzene sahip, standartlar çerçevesinde kayıt altına alınabilmesi büyük bir çalışma ortamı oluşturmaktadır.

3.1. Experience API

Experience API her türlü öğrenme deneyiminin izlenebilmesini ve kaydedilebilmesini sağlayan açık kaynak kodlu bir e-öğrenme yazılımı standardı ve bir API'dir [18] .

API (Application Programming Interface) bir uygulamaya ait yeteneklerin başka uygulamalar ile paylaşılabilmesini sağlar.

Experience API projesinin geliştirilmesi SCORM'un (Sharable Courseware Object Reference Model) geliştiricisi Advanced Distributed Learning (ADL) tarafından yapılmaktadır [18].

Scorm, Experience API ortaya çıkana kadar e-öğrenme aktivitelerine çok fazla katkıda bulunmuş bir e-öğrenme standardıdır. Kendi zamanı içerisinde çığır açmış bir teknolojidir. Experience API, Scorm'un güncellenmiş ve geliştirilmiş bir sürümüdür. Experience API ÖYS dışındaki teknolojilerle de birlikte çalışabilmektedir [19].

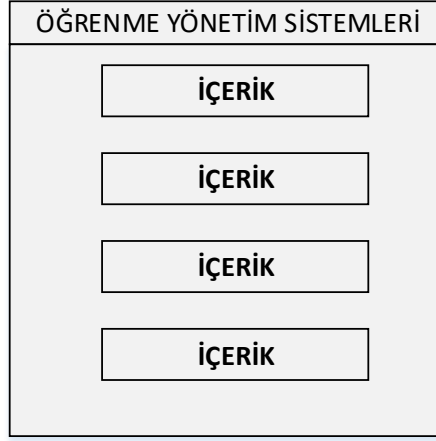
SCORM e-öğrenme içeriklerinin paketlenmesi için kabul görmüş bir standarttır. SCORM ile birlikte ÖYS'lerdeki e-öğrenme içerikleriyle etkileşime giren öğrencilerin davranışlarının izlenebilmesi için birtakım standartlar oluşturmuştur. SCORM sayesinde öğrenci deneyimi logları standart bir şekilde kaydedilebilmektedir. Ayrıca SCORM ile e-öğrenme içeriklerinin farklı öğretim yönetim sistemlerine taşınamaması, ortak olarak kullanılamaması gibi sorunların çözülmesini hedeflemiştir. SCORM paylaşılabilir, yönetilebilir, yeniden kullanılabilir, farklı platformlarda çalışabilir, taşınabilir ve paketlenabilir e-öğrenme içeriklerinin yapılandırılması ve yayımı ile ilgili standartlardan oluşmaktadır [20].

Öğrenme etkinliklerinin bir web tarayıcısı ve ÖYS üzerinde gerçekleştirilmesinin zorunluluk olması SCORM'un en büyük sınırlılığıdır. Ayrıca SCORM paketi ÖYS yöneticisi tarafından önceden belirlenen bir e-öğrenme dersinin içerisine aktarılmalıdır. SCORM'da öğrenme deneyimlerini izlemenin tek yolu, öğrenenin bir ÖYS'ye giriş yapması ve önceden seçilmiş bir e-öğrenme kursuna kaydolması ve daha sonra ÖYS'den SCORM içeriğini başlatması ile mümkün olmaktadır [19].

XAPI ile takip edilecek deneyimler için içeriğin herhangi bir sisteme aktarılmasına ve web tarayıcısına gerek yoktur. Ayrıca XAPI'de, öğrenme deneyiminin nerede ve ne zaman keşfedilip başlatıldığı, ne zaman gerçekleştirildiği önemli değildir, hepsi kaydedilebilmekte ve izlenebilmektedir [19].

SCORM'da sadece ÖYS üzerindeki çevrimiçi öğrenme deneyimleri kaydedilebilmektedir ve SCORM gerçek dünya deneyimleriyle ilgilenmez. Experience API'de ise informal öğrenmeler de bu apinin bir parçasıdır. Öğrenme her yerde, her şekilde ve her ortamda gerçekleşebilir. Çevrimiçi ve çevrimdışı tüm öğrenme deneyimlerinin takip edilebilmesi, akademik başarının kestirilebilmesi için önemlidir. Öğrenme deneyimlerinin bileşimi, parçalardan daha büyük bir toplamı ifade eder.

Ayrıca öğretim faaliyetlerinin tamamı SCORM ile paketlenerek bir öğrenme yönetim sistemine yüklenemeyecek kadar çeşitlilik içerir. Şekil 3.1 'de öğrenim yönetim sistemlerine SCORM paketlerinin yüklenerek eğitim öğretimi tek ortama toplama yaklaşımına ilişkin eski yaklaşım verilmiştir.



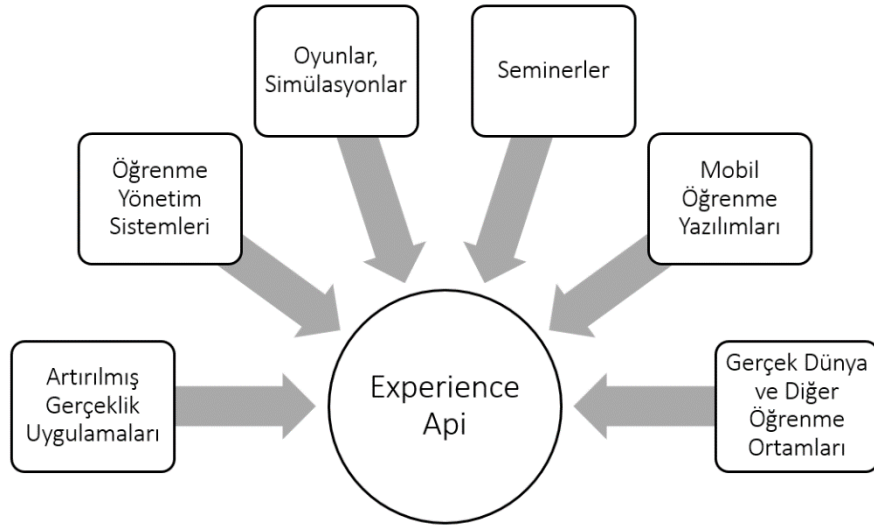
Şekil 3.1. ÖYS ile eski yaklaşım

Bir konferansa katılmak, bir danışmana fikir sormak, bir sınıfa gitmek ve orada kitap okumak bir öğrenme deneyimidir. SCORM ile bu deneyimler ÖYS'ye aktarılamaz ve takip edilemezler. Experience API ile bu deneyimler kolaylıkla takip edilebilmekte ve bu etkinliklerle ilgili deneyim kayıtları kaydedilebilmektedir.

XAPI, bir kişinin öğrenme deneyimlerinin eksiksiz bir resmini oluşturmamızı sağlar [21].

Experience API oyunları, simülasyonları, gerçek dünya performansını, takım temelli e-öğrenmeyi, mobil uygulamalarda e-öğrenmeyi, web tarayıcısı ve web tarayıcısı dışındaki öğrenme deneyimlerini takip edebilecek şekilde tasarlanmıştır [22].

Öğrenme ortamları Şekil 3.2'de verildiği gibi Experience API ile iletişime geçebilecek şekilde tasarlanmalı ve bireylere ait öğrenme deneyimleri API'ye iletilebilmelidir.



Şekil 3.2. Experience API ile deneyim izlemede yeni yaklaşım

Öğrencinin çevrimiçi ve çevrimdışı ortamlarla olan etkileşimleri, API aracılığıyla diğer sistemlere mantıklı bir şekilde aktarılır ve geliştirebilecek bir şekilde iletilir.

XAPI ile davranışın gerçekleştiği ortam ile ilgili sıcaklık, ışık, nem bilgisi, kişiye ait nabız, vücut ısısı gibi bilgiler takip edilebilir.

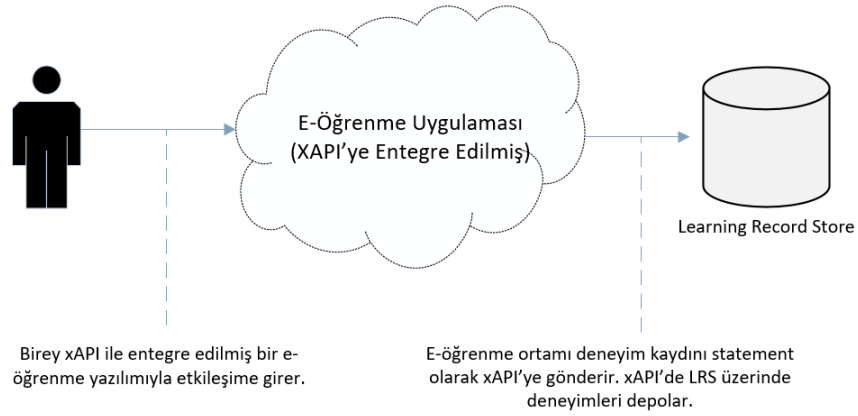
Experience API restful bir servistir ve response (yanıt) tipi olarak JSON kullanmaktadır.

Experience API standardı “<Aktör><Fiil><Nesne>” yapısı şeklinde statement adı verilen deneyim ifadeleriyle çalışmaktadır. Burada statement’dan kastedilen Javascript Object Notation (JSON) formatında öğrenci deneyimlerinin ifade edilmesidir. Her Statement bir öğrenme deneyimini temsil etmektedir [23].

3.2. Learning Record Store (LRS)

Bireyin öğrenme deneyimi elde edebileceği ortamlar Experience API ile entegre edilmelidir. Experience API kendisine iletilen öğrenme deneyimi kayıtlarını Learning Record Store üzerinde depolamaktadır.

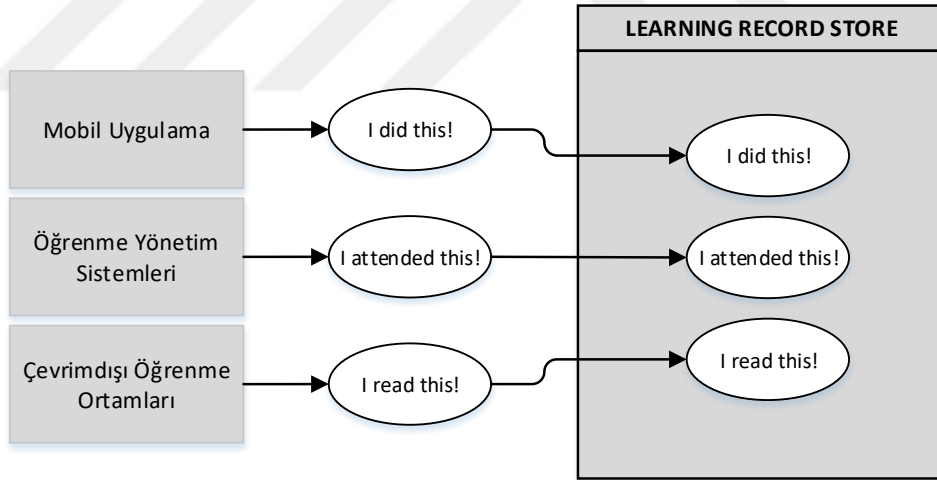
Şekil 3.3’te bireyin Experience API ile entegre edilmiş ortamla girdiği etkileşime ait süreç verilmiştir.



Şekil 3.3. Deneyim ifadelerinin XAPI'den LRS'ye gönderilmesi

LRS, PostgreSQL ile geliştirilmiş dosya tabanlı JSON kayıt tutabilen bir veritabanıdır. Tüm statement'lar LRS üzerinde depolanmaktadır [24].

Şekil 3.4'te farklı ortamların Experience API üzerinden LRS'ye deneyim kayıtlarını kaydettiği süreç verilmiştir.



Şekil 3.4. Deneyim ifadelerinin LRS üzerine kayıt edilmesi

LRS'ler kendi içerisinde bilgi alışverişi yapabilecek şekilde tasarlanmıştır. Statementlar LRS üzerinde depolanmakta ve gerektiğinde XAPI üzerinden öğrenme analitikleri oluşturabilmek için sorgulanabilmektedir.

3.3. Deneyim İfadeleri

Bütün öğrenme deneyimleri bir deyim (statement) olarak ifade edilir ve deyimler XAPI'nin çekirdeğini oluşturur. Tüm öğrenme deneyimleri "I did this" şeklinde bir deyime dönüştürülür. Tüm deyimler XAPI'ye Java Script Object Notation (JSON) formatında iletilir.

Bir statement içerisinde Actor, Verb, Object alanı mutlaka bulunmalıdır. Tablo 3.1 'de bir statement'ın veri modeli gösterilmektedir.

Tablo 3.1. XAPI Statement Veri Modeli [25]

Property	Value	Usage
Actor		
Name	Name of the student	Required
Account		
homePage	URL of the home page of the account management system	Required
name	User identifier in the account management system	Required
Verb		
İd	IRI of the verb (see table below for list of verbs)	Required
Display	Language map of lang code keys to display values	"en": "<verb display>" Required
Object		
İd	IRI of the activity	Required
Definition		
name	Lang map of the name of the activity	Preferred
description	Lang map of the description of the activity	Optional
type	IRI representing the type of activity (see table below for list of activity types)	Required

Tablo 3.1. (Devam) XAPI Statement Veri Modeli [25]

moreInfo	URL to more information	Optional
extensions		
Result		
Score		
Success	Boolean representing the learner's success	Optional
completion	Boolean representing if the learner completed the activity	Optional
Response	Learner's response regarding the activity	Optional
Duration	Duration of time spent in the activity (ISO 8601 duration)	Optional
extensions		
Context		
registration	UUID	Optional
Instructor	Agent	Optional
Team	Group	Optional
contextActivities	Object	See Use Cases Below
Revision	String	Optional
Platform	String	Optional
Language	String / lang code	Optional
Statement	Statement Reference	Optional
extensions		
		Optional
timestamp	ISO timestamp String	Required
AUT		
Attachments		
		Optional

Tablo 3.1 ‘de verilen veri modelinde Actor, Verb, Object, Result, Context, timestamp, Attachments nitelikleri olduğu gözükmektedir. Bu nesnelere ait açıklamalar Tablo 3.2’de verilmiştir.

Tablo 3.2. Statement veri niteliklerinin açıklamaları

Nitelik	Tür	Varsayılan	Açıklama
Id	UUID		UUID LRS tarafından veya güvenilir bir servis tarafından atanır.
Actor	Object		Statement’ın kimin tarafından gerçekleştiğini gösterir. Actor veya Group Object olarak kullanılabilir. “I”
Verb	Object		Statement’ı gerçekleştiren kişi veya grubun eylemidir. “Did”
Object	Object		Gerçekleştirilen eylemdir. “this”
Result	Object		Sonuç nesnesi belirtilen fiille ilişkili daha ayrıntılı bilgi içerir.
Context	Object		Statement’a daha fazla anlam veren bağlamdır. Örnek olarak “Takım oyuncusu, bir uçuş simülasyonu üzerinde çalışıyor”
Timestamp	Date / Time		Statement’ın oluştuğu zaman damgasını ifade etmektedir. ISO 8601’e göre biçimlendirilmiştir. Eğer bu bilgi set edilmemişse LRS bu alana “stored” zamanını yazacaktır.
Stored	Date / Time		Statement’ın LRS’ye kaydedildiği zaman damgasıdır. LRS tarafından otomatik set edilir.
Authority	Object		Bu ifadeyi gönderen uygulamanın kimliğine ilişkin bilgiler bulunur.
Voided	Boolean	False	Bildirim geçersiz olduğunu bildirir.

Tablo 3.3’te Experience API standardında kullanılacak fiillerden bir kısmı verilmiştir. Listenin tamamına XAPI fiiller listesinden erişilebilir [26].

Tablo 3.3. XAPI fiilleri listesi örnekleri

abandoned	experienced	logged-out	satisfied
answered	failed	mastered	scored
asked	imported	passed	shared
attempted	initialized	preferred	suspended
attended	interacted	progressed	terminated
commented	interacted	registered	voided
completed	launched	responded	waived
exited	logged-in	resumed	

İçerisinde sadece Aktör, Fiil ve Nesne barındıran basit JSON ifadeler Simple Statement olarak ifade edilir.

Şekil 3.5 'te bir simple statement örneği verilmektedir.

```
{
  "id": "fd41c918-b88b-4b20-a0a5-a4c32391aaa0",
  "actor": {
    "objectType": "Agent",
    "name": "Erdem Kismet",
    "mbox": "mailto:ekismet@msn.com"
  },
  "verb": {
    "id": "http://adlnet.gov/expapi/verbs/created",
    "display": {
      "en-US": "created"
    }
  },
  "object": {
    "id":
"http://example.adlnet.gov/XAPI/example/simplestatement",
    "definition": {
      "name": { "en-US": "Simple Statement" },
      "description": { "en-US": "A simple Experience API
Statement." }
    }
  }
}
```

Şekil 3.5. Simple Statement

Şekil 3.5. 'te verilen deneyim kaydı şu cümleyi ifade etmektedir:

“Erdem KISMET created A Simple Statement”

Şekil 3.6 ‘da bir complex statement örneği verilmektedir. Verilen statement “Erdem KISMET experienced A Complex Statement with score 100” cümlesini ifade etmektedir. Burada deneyime ilişkin “score” bilgisi de ifadenin içerisine konulabildiği görülmektedir.

```
{
  "id": "fd41c918-b88b-4b20-a0a5-a4c32391aaa0",
  "actor": {
    "objectType": "Agent",
    "name": "Erdem KISMET",
    "mbox": "mailto:ekismet@msn.com"
  },
  "verb": {
    "id": "http://adlnet.gov/expapi/verbs/experienced",
    "display": {
      "en-US": "experienced"
    }
  },
  "object": {
    "id": "http://example.adlnet.gov/XAPI/example/simpleCBT",
    "definition": {
      "name": {
        "en-US": "Experienced a Complex Statement"
      },
      "description": {
        "en-US": "Experienced a Complex Statement"
      }
    }
  },
  "result": {
    "score": {
      "scaled": 1.0
    },
    "success": true,
    "completion": true
  }
}
```

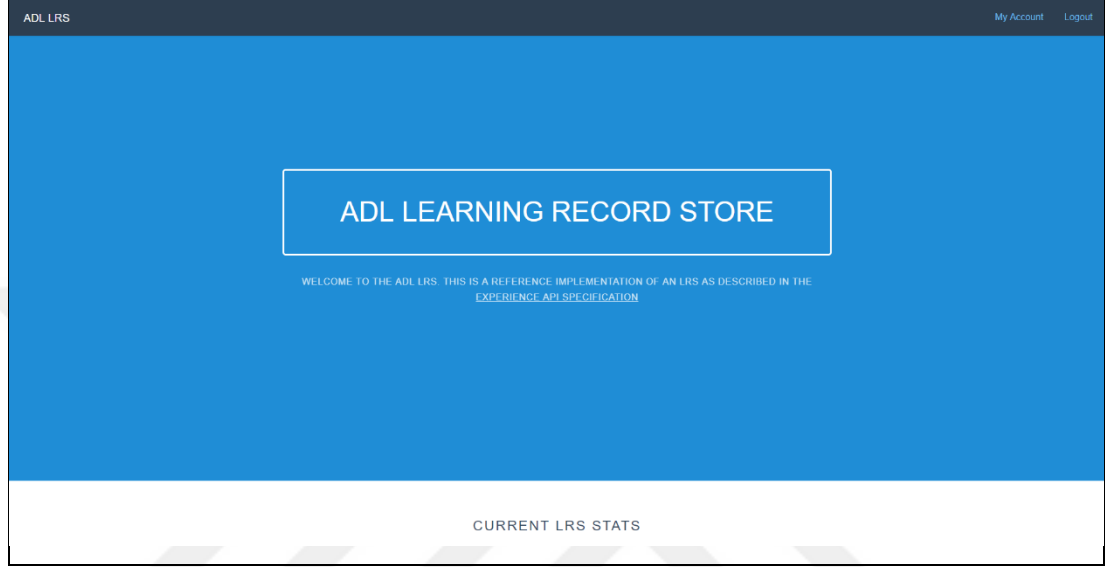
Şekil 3.6. Complex Statement

3.4. Experience API ve Learning Record Store (LRS) Kurulumu

ADLnet tarafından geliştirilen Learning Record Store kurulumunu gerçekleştirebilmek için işletim sistemi olarak Ubuntu 14.04 veya daha üst bir sürüm kullanmak gerekmektedir. Ubuntu 16.04 üzerine kurulum işlemleri için gereken shell komutları açıklamalarıyla birlikte Ek-A’da verilmektedir [24].

İşletim sistemi tercihi ek olarak Python, Django, uWSGI, PostgreSQL, Nginx gibi bir dizi yazılımın kurulması ve yapılandırılması gerekmektedir [24].

Kurulum işlemlerini tamamlayıp kurulumun yapıldığı sunucunun adresine web tarayıcısı üzerinden erişildiğinde Şekil 3.7’deki sayfa açılmaktadır.



Şekil 3.7. ADL LRS

Ek-A’da verilen komutlar yardımıyla tüm yazılımlar kurulum gerekli ayarlamalar yapılandırıldıktan sonra Experience API ve LRS kullanıma hazır hale gelmektedir.

3.5. Veri Toplama

Veri toplamaya ilgili kısıtlardan en önemlisi eğitim ortamlarının Experience API’ye entegre edilmesi gerekliliğidir. Birçok platform bu entegrasyonu sağladı ve her geçen gün entegrasyon sağlayan eğitim ortamlarının sayısı artmaktadır [27].

Ayrıca bir eğitmenin kendi geliştirdiği dijital eğitim ortamını Experience API’ye entegre etmesini sağlayacak birçok hazır yazılım kütüphanesi veya araç ADLnet tarafından açık kaynak kodlu olarak paylaşılmaktadır. Fiziksel ortamların entegre edilmesi konusu programlama bilgisi gerektirmektedir.

Bir birey eğitim ortamına girmeyi kabul ettiğinde, başarısının eğitmen tarafından değerlendirilebilmesi için eğitim sürecinin izleme altında olacağını kabul etmektedir. Aksi takdirde öğrenmenin ne derecede gerçekleşmiş olabileceğine ilişkin kestirim

yapılamaz. Bu nedenle bireyi izleme ve izleme kayıtlarıyla ilgili etik sorunların söz konusu olmadığı düşünülmektedir. Aksi takdirde bu durum bireyin hastalığının teşhisi için doktorunun istediği tahlil sonuçlarını doktordan gizleyerek teşhis koymasını beklemesine benzer.

XAPI ile entegre edilmiş birçok yazılım bulunmaktadır [27]. Bu yazılımlar ile yapılan öğrenme aktivitelerinde bireylere ait öğrenme deneyimlerini kaydetmek oldukça kolaydır. Ancak kendi geliştirdiğiniz projeleri XAPI'ye entegre etmek için bir takım yazılım kütüphanelerinden yararlanılması gerekmektedir.

XAPI ile entegre edilecek çevrimiçi öğrenme ortamları için her deneyim durumunun tespit edilerek açıklanması tavsiye edilmektedir. Bunun için kullanım durumu (use-case) raporu çıkartılarak bu raporda “Deneyim kaydının raporlama ve öğrenme analitiği faaliyetlerine etkisi nedir?” sorusu detaylı bir şekilde açıklanmalıdır. Tablo 3.4’te bir use-case örneği verilmiştir.

Tablo 3.4. XAPI statement use case tablosu [25]

UC#01: Actor Initialized A Video (Örnek)	
Açıklama:	Aktör bir videoyu başlatır.
Aktör (ler):	<ul style="list-style-type: none">• Öğrenci
Ön Koşullar:	<ul style="list-style-type: none">• Video kaydedilir, düzenlenir ve web üzerindeki bir sisteme yüklenir.• Video linki öğrenciye gönderilir.• Bilgisayar veya mobil cihazlar üzerinden video sayfasına bağlanılıp video başlatılır.
Kullanım Durumu Senaryosu / İş Akışı / Sistem Arayüzü: <ol style="list-style-type: none">1. Öğrenci, öğreticisi tarafından paylaşılan linke bağlanır.2. “initialized” deyimini XAPI’ye bir statement olarak gönderilir.	

Tablo 3.4. XAPI statement use case tablosu (devamı) [25]

Son Koşullar:	<ul style="list-style-type: none">● Video izlenilir.● XAPI statement'ı LRS'ye gönderilir.
Faaliyetin Etkisi:	<ul style="list-style-type: none">● Kullanıcıların kaç kez video'ya eriştiği raporlanır.● Kullanıcı erişimlerini ileten görselleştirmeler ve gösterge tabloları oluşturulur.● Video'yu başlatan ve tamamlamayan kullanıcıların korelasyonu hesaplanır.

Yukarıda verilen kullanım durumu (use-case) raporuna ait statement Şekil 3.8'de verilmiştir.

```
{
  "id": "fd41c918-b88b-4b20-a0a5-a4c32391aaa0",
  "actor": {
    "objectType": "Agent",
    "name": "Erdem Kısmet",
    "mbox": "mailto:ekismet@msn.com"
  },
  "verb": {
    "id": "http://adlnet.gov/expapi/verbs/initialized",
    "display": {
      "en-US": "initialized"
    }
  },
  "object": {
    "id": "http://example.adlnet.gov/XAPI/example/video",
    "definition": {
      "name": { "en-US": "Video Example" },
      "description": { "en-US": "Video Example." }
    }
  }
}
```

Şekil 3.8. Verb: Initialized, Simple Statement

3.5.1. Experience Api kütüphaneleri

ADLnet; PHP, JAVA, C#, Python, Javascript gibi birçok popüler programlama dili için Experience API aracılığıyla öğrenme kayıt deposuna kayıt göndermeyi sağlayacak yazılım kütüphanelerini sağlamaktadır. Bu yazılım kütüphaneleri ADLnet'in Github hesabında paylaşılmaktadır.

Ayrıca Restful API yapısı sayesinde projeleri entegre etmek oldukça kolaydır.

3.5.2. Mobil uygulama

Experience API farklı öğrenme ortamlarının birlikte çalışabilirliğini sağlamayı amaçlayan bir standarttır. Bu standart bağlamında cihazların sensör verileri öğrenme deneyimlerinin içerisine eklenebilir ve analiz etmek için kullanılabilir [28].

Bu tez çalışmasında Unity ile geliştirilen mobil uygulama üzerinden öğrencilerle yalnızca Nesne Tabanlı Programlama dersine ilişkin ders videoları paylaşılmıştır. Bu videoların izlenme durumu Experience API'ye iletilerek öğrenci davranışları takip edilmiştir. Unity üzerinde gerçekleştirilen yazılımı XAPI'ye entegre etmek için C# kütüphanesi bu çalışma kapsamında kullanılarak test edilmiştir [29].

Şekil 3.9 'da verilen kodlar yardımıyla mobil uygulama üzerindeki öğrenme aktiviteleri Experience API'ye kolaylıkla iletilebilmektedir.

```
using System;
using TinCan;
using TinCan.LRSResponses;

var lrs = new RemoteLRS (
    "<Experience API endpoint address>",
    "<username>",
    "<password>"
);

var actor = new Agent ();
actor.mbox = "mailto:info@XAPI.com";

var verb = new Verb ();
verb.id = new Uri ("http://adlnet.gov/expapi/verbs/experienced");
verb.display = new LanguageMap ();
verb.display.Add("en-US", "experienced");

var activity = new Activity ();
activity.id = new Uri
("http://rusticissoftware.github.io/TinCan.NET");

var statement = new Statement ();
statement.actor = actor;
statement.verb = verb;
statement.target = activity;

StatementLRSResponse lrsResponse = lrs.SaveStatement (statement);
if (lrsResponse.success)
{
    Console.WriteLine("Save statement: " +
lrsResponse.content.id);
}
```

Şekil 3.9. Mobil uygulama XAPI entegrasyonunda kullanılan kodlar [29]

3.5.3. Öğrenme yönetim sistemlerinin entegre edilmesi

ÖYS'ler üzerindeki kullanıcı deneyimlerinin LRS üzerine kaydedilmesi için geliştirilmiş açık kaynak kodlu birçok modül bulunmaktadır.

Bu çalışmada öğrenme yönetim sistemi olarak açık kaynak kodlu Moodle yazılımı kullanılmaktadır. Bu nedenle Moodle için geliştirilen XAPI entegrasyon modülü kullanılmıştır. Logstore XAPI isminde açık kaynak kodlu modül Moodle üzerine yüklenerek XAPI ile öğrenme yönetim sistemimiz entegre edilmiştir [30].

Şekil 3.10'da Logstore XAPI modülünün ayarlar ekranı gösterilmektedir.

Logstore xAPI

Endpoint
logstore_xapi | endpoint
Varsayılan: http://your.domain.com/endpoint/location/

Username
logstore_xapi | username
Varsayılan: username

Password
logstore_xapi | password
Varsayılan: password

Send statements by scheduled task? Varsayılan: Hayır
logstore_xapi | backgroundmode
This will force Moodle to send the statements to the LRS in the background, via a cron task. This will make the process less close to real time, but will help to prevent unpredictable Moodle performance linked to the performance of the LRS.

Maximum batch size
logstore_xapi | maxbatchsize
Varsayılan: 30
Statements are sent to the LRS in batches. This setting controls the maximum number of statements that will be sent in a single operation. Setting this to zero will cause all available statements to be sent at once, although this is not recommended.

Şekil 3.10. Logstore XAPI modülü ayar ekranı

Endpoint bölümüne XAPI servisinin çalıştığı web sunucusunun adresi yazılmaktadır.

Username ve password bölümlerine XAPI sunucuna erişmeye yetkili kullanıcının bilgileri girilmektedir.

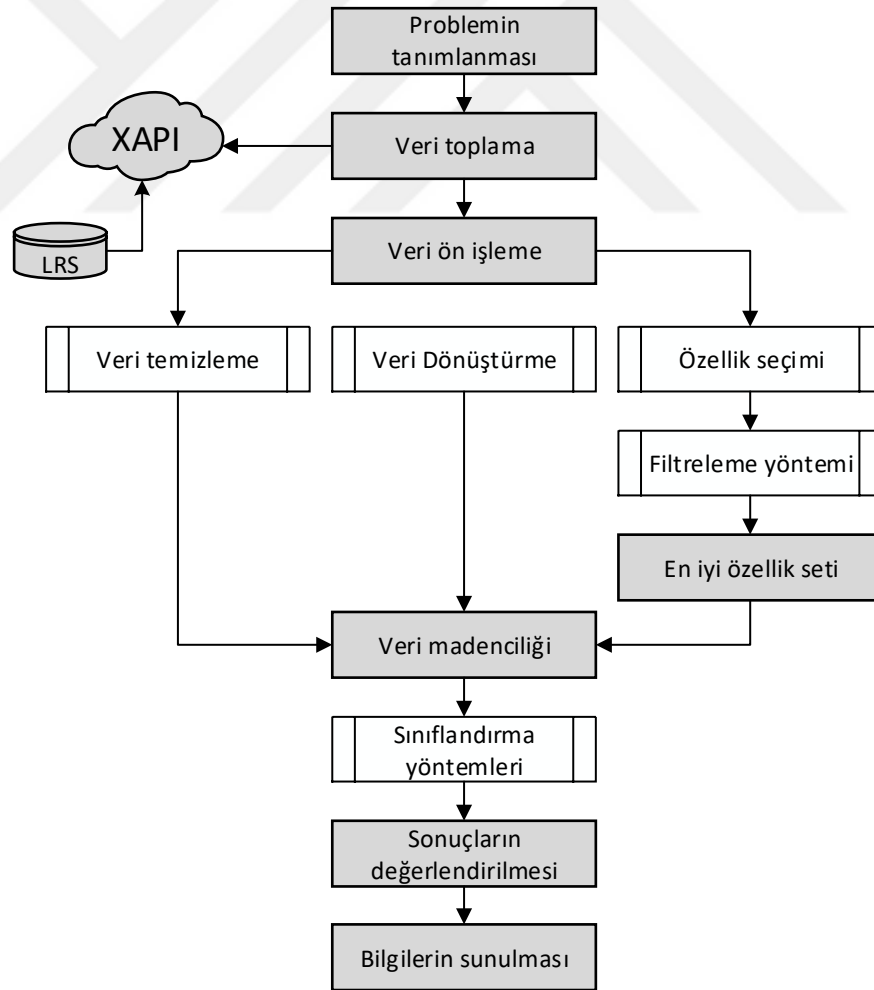
Send statements by scheduled task kısmı gönderme işleminin bir cron job yardımıyla belirli aralıklarla gönderimin sağlanması özelliğini aktif etmek için kullanılmaktadır.

Maximum batch size alanında bir XAPI gönderiminde kaç statement'ın gönderileceği belirtilmektedir.

4. ÖĞRENME DENEYİMİ KAYITLARI ÜZERİNDE EĞİTSEL VERİ MADENCİLİĞİ UYGULAMASI

Günümüzde eğitim verileri hızla artmaktadır. Eğitsel veri madenciliği, eğitim ortamlarından gelen verilerin gizli bilgilerini keşfederek eğitim problemlerini çözmek için geliştirilen yöntemlerle ilgilenen bir disiplindir [1]. Eğitim verilerinden keşfedilen bilgiler, herhangi bir eğitim kurumundaki karar vericilere eğitim sistemlerini geliştirmelerini veya iyileştirmelerini sağlamak için kullanılmaktadır.

Bu bölümde, öğrenci performansını etkileyen öğrenci davranışlarının tespit edilebileceği Şekil 4.1’de gösterilen model önerilmektedir.



Şekil 4.1 Öğrenci performansına ilişkin EVM model diyagramı

4.1. Problemin Tanımlanması

Bireylerin akademik başarısını artırmak, davranışlarını ve öğrenme deneyimlerini iyileştirmekle mümkündür. Öğrenme sürecindeki öğrencinin öğrenme deneyimlerinin takip edilebilmesinin sağlanması ile birlikte oluşan yeni veri yığınları üzerinde keşfedilmesi gereken bilgiler için eğitsel veri madenciliği çalışmaları bu alanda yoğunlaşabilecektir.

Bu çalışmada öğrencilerin öğrenme deneyimlerinin kaydedilebileceği standart bir sistem kurularak öğrencilerin eğitim ortamlarındaki davranışları XAPI aracılığıyla kaydedilmiş ve takip edilmiştir. Öğrenilerin akademik başarısının hangi davranışlarına bağlı olduğunun veri madenciliği teknikleriyle bulunmasını sağlayacak bir model geliştirmektir.

4.2. Kullanılan Teknolojiler

Çalışma kapsamında mobil uygulama geliştirmek için Unity kullanılmıştır. Öğrenme yönetim sistemi olarak açık kaynak kodlu Moodle yazılımı kullanılmıştır. Çevrimdışı ortamlardan kayıt toplayabilmek için Statement Generator yazılımı kullanılmıştır. Ortamlardan veri elde edebilmek için Experience API ve LRS teknolojileri ve bu teknolojilerin kurulumu için gerekli Ek-A'da belirtilen teknolojiler kullanılmıştır. [30]

Verilerin Experience API üzerinden çekilerek veri madenciliği algoritmalarına uygun hale getirilmesi için bir C# uygulaması geliştirilmiştir.

Eğitsel veri madenciliği çalışması R dili kullanılarak geliştirilmiştir.

4.3. Öğrenme Deneyimi Kayıtları Veri Seti ve Ön İşleme

Bu çalışmada veri kaynağı, Öğrenme Deneyimlerinin Yakalanması bölümünde açıklanan, tüm kaynaklardan diğer bir deyişle tüm eğitim ortamlarından veri kaydetme yeteneğine sahip olan Experience API (XAPI) aracılığıyla elde edilmektedir. Experience API'den elde edilen veriler bireylerin davranışsal özelliklerine aittir. XAPI, öğrenme deneyimlerini, bir makale okumak veya bir eğitim videosu izlemek gibi, öğrencinin eylemlerini izlemeyi sağlayan Eğitim ve Öğrenme Mimarisinin (TLA) bir bileşenidir.

Bu tez çalışmasında Nesne Tabanlı Programlama dersini alan 60 öğrencinin farklı eğitim ortamlarındaki öğrenme deneyimleri XAPI aracılığıyla kaydedilmiştir. Öğrenciler mobil uygulama ve öğrenme yönetim sistemi üzerinde çevrimiçi öğrenme deneyimleri yaşamış, çevrimdışı öğrenme deneyimleri ise öğrenmen tarafından kaydedilmiştir. Çalışmanın sonucunda 3902 deneyim kaydı toplanabilmektedir. Şekil 4.2’de LRS’de toplanan öğrenme deneyimi kayıtlarının durumunu gösteren istatistik bölümü gösterilmektedir.

CURRENT LRS STATS				
Users	Statements	Agents	Verbs	Activities
2	3902	60	42	53

Şekil 4.2. Toplanan öğrenme deneyimi kayıtlarının durumu

XAPI üzerindeki kayıtlara ek olarak veri setinde öğrenci bilgi sistemi üzerinden elde edilen öğrencilere ait ders başarı bilgisi kullanılmaktadır.

Öğrenme deneyimi kayıtları LRS içerisinde JSON formatında tutulmaktadır. Bu verileri doğrudan veri madenciliği algoritmalarınca kullanabilmek mümkün değildir. Bu nedenle verileri amaca ve veri madenciliği algoritmalarına uygun bir biçime dönüştürmek gerekmektedir.

Toplanan verilerden elde edilecek veri kümesinin kalitesini artırmak için bazı ön işleme mekanizmaları uygulanmalıdır. Veri ön işleme, veri temizleme, özellik seçimi, veri azaltma ve veri dönüştürme gibi bilgi keşfi sürecinde önemli bir adım olarak düşünülmektedir.

Veri temizleme, gürültüyü azaltmak ve eksik değerler için bu veri setine uygulanır. XAPI içerisine öğrenme yönetim sistemi olarak kullandığımız Moodle üzerinden birçok deneyim kaydı gönderilmiştir. Bu kayıtlar bireyin Moodle ile yaptığı etkileşimleri içermektedir. Örneğin öğrencinin sisteme log-in olması, kurs ekranını görüntülemesi, log-out olması gibi. Bu çalışmada öğrenme yönetim sistemi loglarının analizini yapılmadığı için öğrenci başarısına etkisi merak edilmeyen web sitesi kullanım logları XAPI üzerinden çekilmemiş ve veri setine dâhil edilmemiştir.

Tablo 4.1’de veri setinde kullanılmasına karar verilen deneyim kayıtlarına ilişkin XAPI standardında belirtilen fiillerin listesi verilmiştir.

Tablo 4.1. Veri setinde kullanılan XAPI filleri

Attended	Watch	Answered	Completed
Read	Passed	Experienced	Viewed

Veri setine dahil edilen aktivitelere ilişkin açıklamalar Ek-B’de verilmiştir.

Şekil 4.3. ‘te verilen kodlar geliştirilerek, XAPI üzerinden deneyim kayıtlarının “.csv” formatına dönüştürülmesi sağlanmıştır. Şekil 4.2. ‘de verilen kodlarda “actor.csv” ve “activity.csv” içerisindeki bilgilerden yararlanılarak “data.csv” dosyası oluşturan kodlar bulunmaktadır. Actor.csv dosyası içerisine 60 öğrencinin bilgisi yazılmıştır. Activity.csv içerisine başarıya katkısı merak edilen 53 aktivitenin ID bilgisi girilmiştir. Geliştirilen kodlar tarafından üretilen “data.csv” isimli dosyada 60 satır 53 sütundan oluşmaktadır.

```

var lrs = new RemoteLRS(
    "http://LRSEndPointAddress/XAPI/",
    "admin",
    "*****"
);
List<Agent> listActors = new List<Agent>();
List<string> listActivity = new List<string>();
using (var actorsCsv = new StreamReader(@"C:\temp\actors.csv"))
    while (!actorsCsv.EndOfStream)
    {
        var line = actorsCsv.ReadLine();
        var values = line.Split(';');
        listActors.Add(new Agent() { mbox = "mailto:" +
values[0] });
    }
using (var activityIdCsv = new
StreamReader(@"C:\temp\activityId.csv"))
    while (!activityIdCsv.EndOfStream)
    {
        var line = activityIdCsv.ReadLine();
        var values = line.Split(';');
        listActivity.Add(values[0]);
    }
var query = new StatementsQuery();
var csv = new StringBuilder();
List<string> newLines = new List<string>();
foreach (var agent in listActors)
{
    query.agent = agent;
    foreach (var activity in listActivity)
    {
        query.activityId = activity;
        StatementsResultLRSResponse lrsResponse =
lrs.QueryStatements(query);
        if (lrsResponse.success)
        {
            Console.WriteLine("Count of statements: " +
lrsResponse.content.statements.Count);

            newLines.Add(lrsResponse.content.statements.Count > 0 ?
"1" : "0");

            Console.WriteLine(lrsResponse.content.statements);
        }
    }
    csv.AppendLine(String.Join(",", newLines));
    newLines.Clear();
}
try
{
    File.WriteAllText(@"c:\temp\data.csv", csv.ToString());
    Console.WriteLine("Ok!");
}
catch (Exception e)
{
    Console.WriteLine(e.Message);
}

```

Şekil 4.3. Öğrenme deneyimi kayıtlarının XAPI üzerinden elde edilmesi

Data.csv dosyasına ayrıca öğrenci bilgi sisteminden elde edilen dersi geçme durumu bilgisi eklenmiştir. 54. Sütuna başarılı olan öğrenciler için “GEÇTİ”, dersi geçemeyen öğrenciler için “KALDI” bilgisi eklenmiştir. Tablo 4.2 ve Ek-B’de tablo özellikleri ve açıklamaları bulunmaktadır.

Tablo 4.2. Veri seti özellikleri ve açıklamaları

Özellik Kategorisi	Özellik	Açıklama
Davranışsal Özellikler	Deneyim Kodu 1	Deneyim kayıtlarına verilen kodlar Ek-B ‘de listelenmektedir. Bu çalışmada 53 farklı deneyim kodu bulunmaktadır.
	Deneyim Kodu 2	
	...	
	Deneyim Kodu 53	
Akademik Başarı	Ders Genel Başarısı	Geçti veya Kaldı

Elde edilen data.csv R dilinde veri madenciliği algoritmalarında model oluşturmak için kullanılmaktadır.

Her öğrencinin deneyim kayıtları aynı değildir. Bazı öğrenciler bazı deneyimleri yerine getirmemiş halde bulunmaktadır. Bu nedenle deneyim kaydı bulunmayan öğrenciler için veri setinde karşılık gelen yerlere 0 değeri yazılmıştır.

Öğrencinin başarı durumu sınıf geçme notunun üstündeki öğrenciler “geçti”, sınıf geçme notunun altında kalan öğrenciler “kaldı” olacak şekilde dönüştürülmüştür. Ayrıca bazı deneyimler 0-100 arasında ölçeklendirilmiş bir puan ile gönderilmektedir. Örneğin çevrimdışı bir proje görevi eğitmen tarafından puanlanmaktadır. Bu puanlamalar 0-1 arasına ölçeklendirilecek şekilde dönüştürülmüş.

Ayrıca bazı öğrenciler bir deneyimi birden fazla kez tekrar etmişlerdir. Geliştirdiğimiz modelde tekrar edilen davranışlar hesaba katılmamıştır. Örneğin bir birey bir mobil uygulama üzerinden bir videoyu 5 kere izleyebilir ve her seferinde sistem bunun için bir deneyim kaydı gönderebilir. Bu çalışmada davranış tekrarlarının etkisi araştırılmadığı için bu veriler yaptı, yapmadı şeklinde değerlendirilmiştir. Eğer bu etki araştırılmak istenseydi min-max normalizasyonu kullanılarak modele uygun hale dönüştürülebilirdi. Tablo 4.3’te veri setinin içeriğinden bir kısım paylaşılmıştır.

Tablo 4.3. Veri seti örneği

DENEYIM_50	DENEYIM_51	DENEYIM_52	DENEYIM_53	DURUM
1	1	0,9	0,9	GECTI
	1	0,8	0,9	GECTI
1	1	0,7	0,7	GECTI
	1	0,9	0,9	GECTI
1	1	1	0,7	GECTI
1	1	1	0,4	GECTI
1	1	0,8	0,8	GECTI
	1	0,2	0	KALDI
1	1	0,1	0	KALDI
	1	0,6	0,4	KALDI
1	1	0,4	0,6	KALDI
1	1	0	0,6	KALDI
	1	0,6	0,4	KALDI

Veri madenciliği algoritmalarında veri seti, modeli kurabilme amacıyla eğitim verisi ve modelin doğruluğunu kontrol edebilme amacıyla test verisi olarak iki parçaya ayrılarak kullanılmaktadır. Modelin oluşturulması için kullanılacak eğitim verilerinin test verisine göre daha fazla miktarda olması modelin daha iyi eğitilmesini sağlayacaktır. Eğitim verisinin tüm verilere oranının kaç olacağına ilişkin kesin bir bilgi yoktur ancak genellikle 2/3, 3/4, %75 veya %80 gibi oranlar kullanılmaktadır.

R Programlama dilinde bu dönüşüm Caret paketi kullanılarak yapılmaktadır. Şekil 4.4'te Caret paketi kullanılarak veri setinin eğitim ve test verisi şeklinde bölünmesi gösterilmektedir.

```

> install.packages("caret")
> library("caret")
> set.seed(35)
> egitimIndisleri<-createDataPartition(y=XAPI$DURUM,p=.70,list=FALSE)
> egitimV<-XAPI[egitimIndisleri,]
> testV<-XAPI[-egitimIndisleri,]

```

Şekil 4.4. Veri setinin eğitim ve test verisi olarak ikiye bölünmesi

Bölünmenin rastgele bir indisten başlamasını sağlayabilmek için set.seed() fonksiyonu kullanılmaktadır. Verilen örnekte createDataPartition() fonksiyonu kullanılarak veri setinin %70'lik kısmı eğitim verisi için ayrılmaktadır. Bu aşamadan sonra veri setine R dilindeki veri madenciliği kütüphaneleri yüklenerek veri madenciliği çalışmaları gerçekleştirilecektir.

4.4. Algoritma Seçimi ve Model Kurma

Veri madenciliğinde sınıflandırma amacıyla Karar Ağaçları, Genetik Algoritmalar, Yapay Sinir Ağları, Naive-Bayes, Lojistik Regresyon gibi algoritma kullanılmaktadır. Bu tez kapsamında gözetilen ana unsur kullanılan algoritmaların sayısı ve çeşitliliği değildir. Bu nedenle bu tez çalışmasında C4.5, Gini ve Rastgele Orman algoritmaları kullanılmaktadır.

Sınıflandırma algoritmalarının oluşturacağı model başarıklık veya başarısızlık için gereken davranışın tespit edilebilmesi için anlamlı sonuçlar üretecektir.

4.4.1. C4.5 Karar Ağacı Algoritması

R dilinde C4.5 algoritmasından yararlanabilmek için “RWeka” kütüphanesi yüklenmelidir. RWeka kütüphanesi veri ön işleme, sınıflandırma, regresyon, kümeleme, birliktelik kuralları gibi veri madenciliğine ilişkin algoritmaları içeren bir Java kütüphanesidir. RWeka içerisinde bulunan J48 fonksiyonu C4.5 karar ağacı oluşturmak için kullanılmaktadır. Şekil 4.5'te R dilinde C4.5 modelini oluşturan kodlar gösterilmektedir.

```
> library("RWeka")
> egitimV$DURUM<-as.character(egitimV$DURUM)
> egitimV$DURUM<-as.factor(egitimV$DURUM)
> C45_model<-J48(DURUM~.,data=egitimV)
```

Şekil 4.5. Modelin oluşturulması

Oluşturulan modeli göstermek için Şekil 4.6'da gösterildiği gibi show() fonksiyonu kullanılmaktadır.

```

> show(C45_model)
J48 pruned tree
-----

DENEYIM_32 <= 0
| DENEYIM_12 <= 0.8: KALDI (10.0)
| DENEYIM_12 > 0.8: GECTI (3.0/1.0)
DENEYIM_32 > 0
| DENEYIM_52 <= 0.2: KALDI (4.0)
| DENEYIM_52 > 0.2
| | DENEYIM_9 <= 0
| | | DENEYIM_11 <= 0: KALDI (2.0)
| | | DENEYIM_11 > 0: GECTI (2.0)
| | DENEYIM_9 > 0: GECTI (22.0)

Number of Leaves :    6
Size of the tree :   11

```

Şekil 4.6. Modelin gösterimi

Sınıflandırma sonucunda oluşan istatistiksel bilgi Şekil 4.7’de gösterildiği gibi summary() fonksiyonu kullanılarak görüntülenebilmektedir.

```

> summary(C45_model)

=== Summary ===

Correctly Classified Instances          42           97.6744 %
Kappa statistic                        0.9509
Mean absolute error                    0.031
Root mean squared error                0.1245
Relative absolute error                 6.4725 %
Root relative squared error            25.4658 %
Total Number of Instances              43

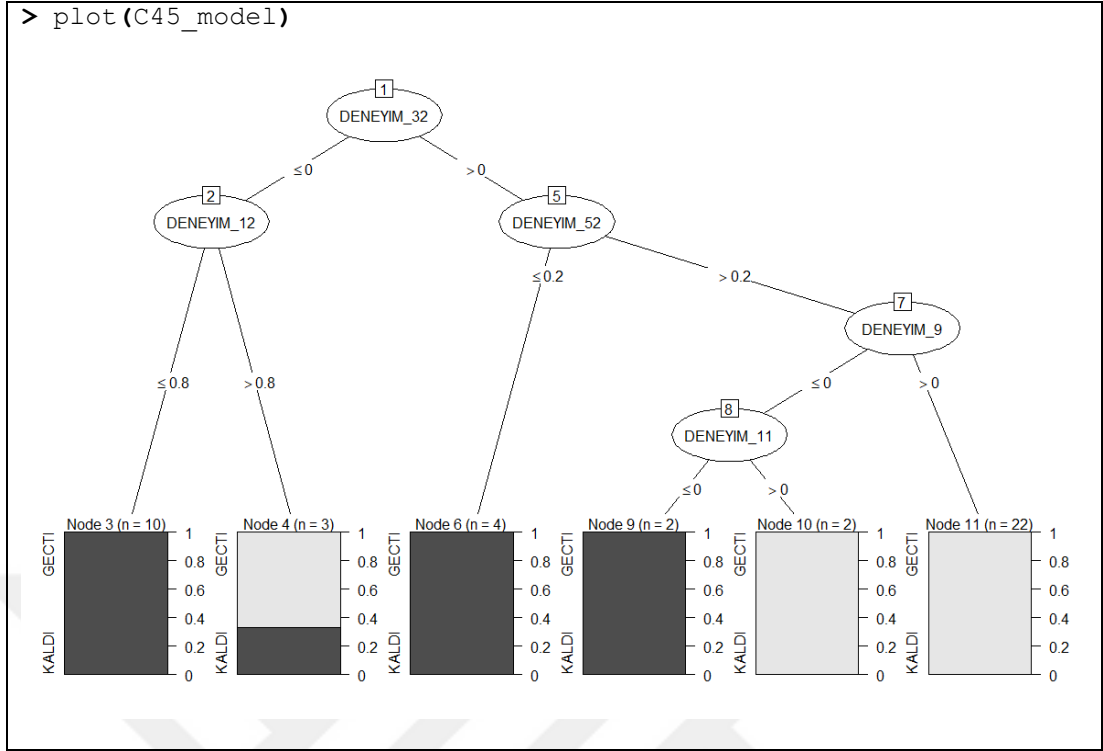
=== Confusion Matrix ===

  a  b  <-- classified as
26  0  |  a = GECTI
 1 16 |  b = KALDI

```

Şekil 4.7. Sınıflandırma sonucuna göre istatistiksel bilgi

Oluşturulan model plot() fonksiyonu ile çizdirilebilmekte ve Şekil 4.8’de oluşturulan model gösterilmektedir.



Şekil 4.8. C4.5 ile elde edilen karar ağacı

C4.5 algoritması sonucunda oluşturulan ve Şekil 4.8’de görülen karar ağacından elde edilen kurallar Şekil 4.9’da verilmiştir.

Kural 1:

Eğer 32 numaralı deneyim yapıldı ve

Eğer 52 numaralı deneyim yapıldı ve

Eğer 9 numaralı deneyim yapıldı **ise Geçti**

Kural 2:

Eğer 32 numaralı deneyim yapıldı ve

Eğer 52 numaralı deneyim yapıldı ve

Eğer 9 numaralı deneyim yapılmadı ve

Eğer 11 numaralı deneyim yapıldı **ise Geçti**

Kural 3:

Eğer 32 numaralı deneyim yapıldı ve

Eğer 52 numaralı deneyim yapıldı ve

Eğer 9 numaralı deneyim yapılmadı ve

Eğer 11 numaralı deneyim yapılmadı **ise Kaldı**

Kural 4:

Eğer 32 numaralı deneyim yapıldı ve

Eğer 52 numaralı deneyim yapılmadı **ise Kaldı**

Kural 5:

Eğer 32 numaralı deneyim yapılmadı ve

Eğer 12 numaralı deneyim yapıldı **ise Geçti**

Kural 6:

Eğer 32 numaralı deneyim yapılmadı ve

Eğer 12 numaralı deneyim yapıldı **ise Kaldı**

Şekil 4.9. Karar ağacından elde edilen kurallar

Eđitim verisi ile oluřturulan modelin dođruluđu kontrol etmek iin, ortaya ıkan ađa predict() fonksiyonu ile tahminlenir. Kestirim sonucu, test verisi ile karřılařtırılarak ‘‘Karřılık Matrisi’’ oluřturulur. Őekil 4.10’da karřılık matrisinin oluřturulması gsterilmektedir.

```
> kestirimC<-predict(C45_model,testV,type = "class")
> karMatrisC<-table(testV$DURUM,kestirimC,dnn=c("KABUL","ÖNGÖRÜ"))

> show(karMatrisC)
      ÖNGÖRÜ
KABUL  BASARILI BASARISIZ
BASARILI      9          1
BASARISIZ     5          2
```

Őekil 4.10. Karřılık matrisi

Oluřturulan modelde karřılık matrisine gre model bařarımı iin hesaplanan deđerler Tablo 4.4.’te verilmektedir.

Tablo 4.4. C4.5 model bařarımı

Dođruluk	Kesinlik	Duyarlılık	F-Ölütü
0,64706	0,64286	0,64286	1,928571

Oluřturulan model sonucunda 32, 12, 52, 9 ve 11 numaralı deneyimler đrenci bařarısı üzerinde en fazla etki eden deneyimler olduđu tespit edilmiřtir. Bu deneyimleri tecrbe eden đrencilerin bařarılı olma řansları ykselmektedir. Bu nedenle đrencilere deneyimlemeleri iin bu đrenme yařantıları nerilmelidir. Ek-B’de verilen tabloda tm deneyimlere ait aıklamalar bulunmaktadır. Tablo 4.5’te verilen deneyimler Ek-B’den elde edilmiřtir.

Tablo 4.5. Önerilen davranışlar

32	Verb	http://activitystrea.ms/schema/1.0/present
	Object ID	http://www.erdemkismet.com.tr/statement-generator/present-topic
	Kaynak	Çevrimdışı Ortam
	Veri	Değerlendirme notu (0-100 arasında)
12	Verb	http://adlnet.gov/expapi/verbs/passed
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-2-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Değerlendirme notu (0-100 arasında)
52	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-7_2-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
9	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-2-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
11	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-2-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)

Tablo 4.5’te verilen deneyimler mevcut öğrenci grubu ve mevcut başarılilik durumunu belirleyen sınavlara göre tespit edilmiştir. Bu nedenle başarılilik durumunu belirleyen sınavlar modelin tespit ettiği deneyim kodlarındaki belirtilen deneyimleri daha fazla ölçüyor olabilir. Yani diğer deneyimleri yeteri kadar ölçülüyor olabilir.

Modelin önermediği diğer deneyimler başarılı ile başarısız ayırt edebilecek duruma getirmek amacıyla geliştirilmesi gerekebilir.

Açıkça görülüyor ki öğrenme deneyimleri üzerinde yapılan eğitsel veri madenciliği çalışmaları, öğrencilerin başarısını artırmak için ve eğitim ortamını geliştirebilmek için öneriler sunabilmektedir. Bu da eğitsel veri madenciliği disiplinin amacına uygun olması anlamına gelmektedir.

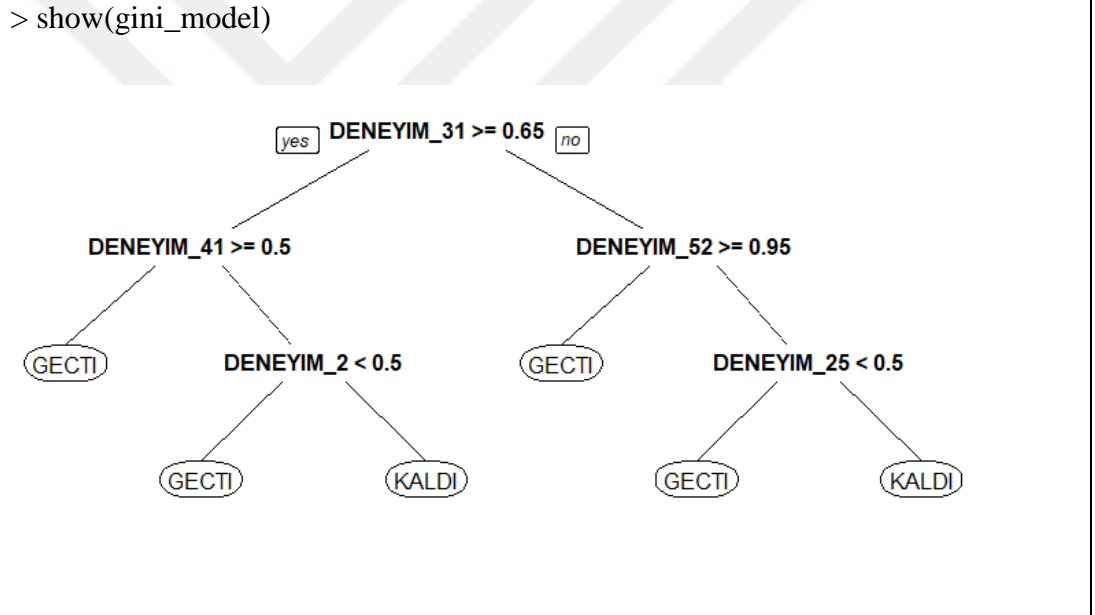
4.4.2. Gini Algoritması

Gini algoritması ikili bölünme temeline dayanan bir sınıflandırma algoritmasıdır. R dilinde gini algoritmasını çalıştırabilmek için rpart paketi kullanılmaktadır.

```
> install.packages("rpart")
> library(rpart)
> gini_model<-rpart(DURUM~.,data = egitimV,method = "class",minsplit=4,parms
= list(split="gini_model"))
```

Şekil 4.11. Gini algoritması için modelin oluşturulması

Split değeri gini olarak set edilen ağaç Show() fonksyonu ile gösterilebilmektedir.



Şekil 4.12. Gini algoritması için oluşturulan model

Gini ağacı ile oluşturulan modelde 31, 41, 52, 2 ve 25 numaralı deneyimler başarı ile başarısızlığı belirlemede en etkili olan deneyimlerdir.

Eğitim verisi ile oluşturulan modelin doğruluğu kontrol etmek için, ortaya çıkan ağaç predict() fonksiyonu ile tahminlenir. Kestirim sonucu, test verisi ile karşılaştırılarak "Karşılık Matrisi" oluşturulur. Şekil 4.12'da karşılık matrisinin oluşturulması gösterilmektedir.

```

> #Karşılık Matrisi
> kestirim<-predict(gini_model,testV,type = "class")
> karMatrisG<-table(testV$DURUM,kestirim,dnn=c("KABUL","ÖNGÖRÜ"))
> show(karMatrisG)
      ÖNGÖRÜ
KABUL  GECTI  KALDI
GECTI   9     1
KALDI   3     4

```

Şekil 4.12. Gini modeli karşılık matrisi

Oluşturulan modelde karşılık matrisine göre model başarımı için hesaplanan değerler Tablo 4.6.'te verilmektedir.

Tablo 4.6. Gini algoritması model başarımı

Doğruluk	Kesinlik	Duyarlılık	F-Ölçütü
0,7647059	0,75	0,75	2,25

Başarılı ile başarısız öğrencinin davranışlarına ilişkin olarak Gini ağacı ile oluşturulan model 31, 41, 52, 2 ve 25 numaralı deneyimleri etkili olan deneyimler olarak belirlemiştir. Bu deneyimlere ilişkin açıklamalar Ek-B'de belirtilmektedir.

5. SONUÇLAR VE ÖNERİLER

Öğrenme farklı eğitsel ortamlarda gerçekleşmektedir. Literatürde yapılan çalışmaların aksine bu çalışmada farklı eğitsel ortamlardan elde edilen öğrenci davranışları tek bir merkezde toplanmış ve eğitsel veri madenciliği çalışması için yalnızca davranışsal özellikler kullanılmıştır.

Bu tez çalışmasında öğrenme deneyimi kayıtlarının tutulabilmesi için gereken alt yapıyı sağlayan teknolojiler tanıtılarak, öğrencilerin öğrenme deneyimlerinin kayıt altına alınabileceği yerel bir LRS sunucusu kurulmuştur. Öğrencilere ait deneyimlerin kaydedilebileceği bir mobil uygulama Unity ile geliştirilmiş ve bu sistem üzerinden öğrencilere video içerik gönderilmiştir. Mobil uygulamadaki öğrenci deneyimleri Experience API aracılığıyla takip edilmiştir. Ayrıca bazı materyaller öğrenme yönetim sistemi üzerinden paylaşılmış ve bu deneyimler de Experience API üzerinden LRS üzerine kaydedilmiştir. Çevrimdışı öğrenme deneyimlerinin LRS'ye kaydedilmesi için Statement Generator yazılımı geliştirilmiş ve bu yazılım kullanılmıştır.

Standart ve merkezi bir ortamda tutulan öğrenme deneyimi kayıtları üzerinde eğitsel veri madenciliği çalışmaları yapabilmek için verilerin LRS sunucusundan veri madenciliği algoritmalarına uygun hale getirilmesi için entegrasyon yazılımı C# diliyle geliştirilmiştir. Veri seti LRS üzerinden veri madenciliği algoritmalarına uygun hale getirildikten sonra hangi deneyimin öğrenci başarısı üzerinde etkisinin olduğu araştırılmıştır.

Sınıflandırma algoritmalarıyla yapılan değerlendirme sonucunda öğrencilere önerilebilecek davranışlar tespit edilebilmiştir. Oluşturulan C4.5 karar ağacı modelinde %64 oranında Gini ağacında %76 oranında doğruluk değerine ulaşılmıştır.

Tüm bu çalışmalar sayesinde öğrencilerin başarısını artırmada hangi deneyimin daha etkili olduğunu ve daha önerilebilir olduğunu tespit etmeyi sağlayacak bir model önerilmektedir. Ayrıca üretilen sonuçlar eğitsel ortamları iyileştirebilme amacıyla kullanılabilmesi de sağlanabilmektedir.

Yapılan çalışmalar sonucunda eğitim öğretim ortamlarının takip edilebilmesini sağlamakla ilgili bir takım sıkıntılarının olduğu tespit edilmiştir. Bir eğitimcinin öğrenme ortamına ilişkin deneyimleri kaydedebileceği sistemi kurabilmesi için programlama bilgisine ihtiyacı olduğu görülmüştür. XAPI'ye entegre olan eğitim ortamları artıp veri toplama işlemi kolaylaşmadığı sürece bu alanda veri seti oluşturmak zordur.

Yeni e-öğrenme ortamları oluştururken e-öğrenme alanındaki yenilikler ve oluşturulan sistemlerde öğrenme deneyimlerinin merkezi bir yere kayıt edilebileceği göz önünde bulundurulmalıdır.

KAYNAKLAR

- [1] Dutt A., Ismail M. A., Herawan T. A., Systematic Review on Educational Data Mining, IEEE Access, DOI: 10.1109/ACCESS.2017.2654247.
- [2] Erdoğan Ş. Z., Veri Madenciliği Ve Veri Madenciliğinde Kullanılan K-Means Algoritmasının Öğrenci Veri Tabanında Uygulanması, Yüksek Lisans Tezi, İstanbul Üniversitesi, Sosyal Bilimler Enstitüsü, İstanbul, 2004, 146984.
- [3] Gülçe G., Veri Ambarı Ve Veri Madenciliği Teknikleri Kullanılarak Öğrenci Karar Destek Sistemi Oluşturma, Yüksek Lisans Tezi, Pamukkale Üniversitesi, Fen Bilimleri Enstitüsü, Denizli, 2010, 275300.
- [4] Ekim U., Veri Madenciliği Algoritmalarını Kullanarak Öğrenci Verilerinden Birliktelik Kurallarının Çıkarılması, Yüksek Lisans Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya, 2011, 302517.
- [5] Taşdemir M., Veri Madenciliği: Öğrenci Başarısına Etki Eden Faktörlerin Regresyon Analizi İle Tespiti, Yüksek Lisans Tezi, Dicle Üniversitesi, Sosyal Bilimleri Enstitüsü, Diyarbakır, 2012, 326726.
- [6] Saygılı A., Veri Madenciliği İle Mühendislik Fakültesi Öğrencilerinin Okul Başarılarının Analizi, Yüksek Lisans Tezi, Yıldız Teknik Üniversitesi, Fen Bilimleri Enstitüsü, İstanbul, 2013, 329613.
- [7] Kılınç Ç., Üniversite Öğrenci Başarısı Üzerine Etki Eden Faktörlerin Veri Madenciliği Yöntemleri İle İncelenmesi, Yüksek Lisans Tezi, Selçuk Üniversitesi, Fen Bilimleri Enstitüsü, Konya, 2015, 415460.
- [8] Güler Ç., Ortaöğretim Başarısına Etki Eden Faktörlerin Karar Ağacı İle Sınıflandırılması, Yüksek Lisans Tezi, Kahramanmaraş Sütçü İmam Üniversitesi, Fen Bilimleri Enstitüsü, Kahramanmaraş, 2017, 473120.
- [9] Altınışık U., Öğrenci Bilgi Sisteminde Veri Madenciliğinin Uygulanması, Yüksek Lisans Tezi, Kocaeli Üniversitesi, Fen Bilimleri Enstitüsü, Kocaeli, 2006, 167478.
- [10] Bresfelean P., Bresfelean M., Ghisoiu N., Determining Students' Academic Failure Profile Founded on Data Mining Methods, *Information Technology Interfaces*, Dubrovnik, Croatia, 23-26 Haziran 2008.
- [11] Yavualp N., E-Öğrenme Ortamında Kullanılan Öğrenme Stil Ve Stratejilerinin Web Kullanım Madenciliği İle Analizi, Doktora Tezi, Fırat Üniversitesi, Eğitim Bilimleri Enstitüsü, Elazığ, 2012, 187258.

- [12] Özbay Ö., Öğretim Yönetim Sistemi Üzerinde Üniversite (Lisans) Düzeyindeki Öğrenci Hareketliliğinin Veri Madenciliği Yöntemleriyle Analizi, Yüksek Lisans Tezi, Başkent Üniversitesi, Eğitim Bilimleri Enstitüsü, Ankara, 2015, 414627.
- [13] Amrieh E. A., Hamtini T., Aljarah I., Preprocessing and Analyzing Educational Data Set Using X-API for Improving Student's Performance, *Applied Electrical Engineering and Computing Technologies (AEECT)*, Amman, Jordan, 3-5 Kasım 2015.
- [14] Chakravarthy S. S., Raman A.C., Educational Data Mining on Learning Management Systems using Experience API, *Communication Systems and Network Technologies (CSNT)*, Bhopal, India, 7-8 Nisan 2014.
- [15] Blanco Á., Serrano Á., Freire M., E-Learning Standards and Learning Analytics, *Global Engineering Education Conference*, Berlin, Germany, 13-15 Mart 2013.
- [16] Quinlan J. R., C4. 5: programs for machine learning, Morgan Kaufmann Publisher, California, 1992.
- [17] Mutlu M., Öğrenme Deneyimlerinin Yorumlanması, *Eğitim ve Öğretim Araştırmaları Dergisi*, 2014, **3**(4), 21-45.
- [18] Experience API Overview. <http://experienceapi.com/overview/> (Ziyaret tarihi: 20 Eylül 2017).
- [19] Layer 1: A Modernized version of SCORM. <https://experienceapi.com/layer-1-freeing-us-from-the-constructs-of-old/> (24 Ekim 2016 tarihinde erişilmiştir.)
- [20] Mutlu M., Experience API ile Yaşam Günlüğüne Dayalı Öğrenme Deneyimlerinin Kaydedilmesi, *XIX. Türkiye'de İnternet Konferansı (İNET)*, İzmir, Türkiye, 27-29 Kasım.
- [21] Layer 2: Record Any Learning Experience (Informal Learning). <https://experienceapi.com/layer-2-record-any-learning-experience-informal-learning/> (20 Eylül 2017 tarihinde erişilmiştir.)
- [22] Serrano-Laguna Á., Martínez-Ortiz I., Haag J., Regan D., Johnson A., Applying standards to systematize learning analytics in serious games, *Computer Standards & Interfaces*, 2017, **50**(2017), 116-123.
- [23] Vidal J. C., Rabelo T., Yamaura M., Lama M., Semantic description of the Experience API Specification, *IEEE 15th International Conference on Advanced Learning Technologies*, Hualien, Taiwan, 6-9 Temmuz 2015.
- [24] ADL LRS Kütüphanesi, Github, https://github.com/adlnet/ADL_LRS (Ziyaret tarihi: 27 Ekim 2017).
- [25] XAPI Use Cases & Statements, <http://XAPI.vocab.pub/resources/index.html>. (Ziyaret tarihi: 3 Kasım 2017).

- [26] XAPI All Verbs, <http://XAPI.vocab.pub/verbs/index.html> (Ziyaret tarihi: 3 Kasım 2017).
- [27] Experience API Adapters. <http://experienceapi.com/adopters/> (Ziyaret Tarihi: 4 Kasım 2017).
- [28] Glahn C., Using the ADL Experience API for Mobile Learning, Sensing, Informing, Encouraging, Orchestrating, *Next Generation Mobile Apps, Services and Technologies (NGMAST)*, Prague, Czech Republic, 25-27 September 2013.
- [29] Experience API C# kütüphanesi. <http://rusticisoftware.github.io/TinCan.NET/> (Ziyaret Tarihi: 1 Aralık 2017).
- [30] Moodle Logstore Modülü, Moodle, https://moodle.org/plugins/logstore_XAPI (Ziyaret tarihi: 1 Aralık 2017).
- [31] Amrieh E. A., Hamtini T., Aljarah I., Mining Educational Data to Predict Student's academic Performance using Ensemble Methods, *International Journal of Database Theory and Application*, 2016, **9**(8), 119-136.
- [32] Vázquez M. M., Rodríguez M. C., Nistal M. L. Development of a XAPI Application Profile for Self-Regulated Learning, *Global Engineering Education Conference*, Tallinn, Estonia, 18-20 Mart 2015.
- [33] Experience API, ADL, <http://www.adlnet.gov/adl-research/performance-tracking-analysis/experience-api/> (Ziyaret tarihi: 2 Aralık 2017).



EKLER

EK-A

Bölüm 3.3'te açıklanan Ubuntu 16.04 üzerine LRS kurulumuna ilişkin adımlar açıklamalarıyla birlikte aşağıda verilmiştir.

```
admin:~$ wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc |
sudo apt-key add -
admin:~$ sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb_release
-cs)-pgdg main" >> /etc/apt/sources.list.d/postgresql.list'
admin:~$ sudo apt-get update
admin:~$ sudo apt-get upgrade
```

POSTGRESQL Yükleme için

```
admin:~$ sudo apt-get install postgresql-9.5 postgresql-server-dev-9.5 postgresql-
contrib-9.5
```

POSTGRESQL Yeni kullanıcı oluşturma

```
admin:~$ sudo -u postgres createuser -P -s <db_owner_name>
Enter password for new role: <db_owner_password>
Enter it again: <db_owner_password>
admin:~$ sudo -u postgres psql template1
template1=# CREATE DATABASE lrs OWNER <db_owner_name>;
template1=# \q (exits shell)
```

PYTHON ve diğer gereksinimleri yükleme

```
admin:~$ sudo apt-get install git fabric python-setuptools python-dev libxml2-dev
libxslt1-dev gcc
admin:~$ sudo easy_install pip
admin:~$ sudo pip install virtualenv
```

ADL_LRS'yi github'dan edinme

```
admin:~$ cd <wherever you want to put the LRS>
admin:~$ git clone https://github.com/adlnet/ADL_LRS.git
admin:~$ cd ADL_LRS/
```

Kurulum

```
admin:ADL_LRS$ fab setup_env
admin:ADL_LRS$ source ../env/bin/activate
(env)admin:ADL_LRS$
```

```
(env)admin:ADL_LRS$ fab setup_lrs
```

...

You just installed Django's auth system, which means you don't have any superusers defined.

Would you like to create one now? (yes/no): yes

Username (leave blank to use '<system_user_name>'):

E-mail address:

Password: <this can be different than your system password since this will just be for the LRS site>

Password (again):
Superuser created successfully.

```
(env)dbowner:ADL_LRS$ ./manage.py runserver
```

```
[Unit]
```

```
Description=LRS
```

```
[Service]
```

```
ExecStart=/path/to/env/bin/uwsgi --emperor /etc/uwsgi/vassals
```

```
Restart=on-failure
```

```
[Install]
```

```
WantedBy=multi-user.target
```

```
If your server ever goes down, uWSGI will automatically start now and can be stopped and started:
```

```
#LRS servisini başlatma
```

```
sudo systemctl start lrs
```

```
sudo systemctl stop lrs
```

```
# LRS sunucunuza dışarıdan erişebilmek için NGINX sunucusu kurulumu ve konfigürasyonu yapılmalıdır.
```

```
sudo apt-get install nginx
```

```
# Eğer sadece LRS için NGINX kurulumu yapıldıysa aşağıda NGINX konfigürasyonu kodları /etc/nginx/conf.d/default.conf üzerine yazılmalıdır.
```

```
# NGINX AYARLARI
```

```
upstream lrs {
```

```
    server localhost:8000;
```

```
    ip_hash;
```

```
}
```

```
server {
```

```
    listen 80;
```

```
    server_name localhost;
```

```
    charset utf-8;
```

```
# full path to the project dir - the dir that contains the urls.py file
```

```
root /path/to/ADL_LRS/adl_lrs;
```

```
access_log /path/to/logs/nginx/nginx_access.log;
```

```
error_log /path/to/logs/nginx/nginx_error.log;
```

```
## compression
```

```
gzip on;
```

```
gzip_buffers 16 8k;
```

```
gzip_comp_level 4;
```

```
gzip_http_version 1.0;
gzip_min_length 1280;
gzip_types text/plain text/css application/x-javascript application/json
text/javascript image/jpeg image/bmp;
gzip_vary on;
```

```
location / {
    client_max_body_size 1M;
    proxy_pass_header Server;
    proxy_set_header Host $http_host;
    proxy_redirect off;
    proxy_set_header X-Real-IP $remote_addr;
    proxy_set_header X-Scheme $scheme;
    proxy_connect_timeout 10;
    proxy_read_timeout 10;
    proxy_pass http://lrs;
}
```

```
location /static/ {
    autoindex on;
    expires 1w;
    root /path/to/ADL_LRS/adl_lrs;
}
```

```
location /static/ endless_pagination {
    autoindex on;
    expires 1w;
    alias /path/to/env/lib/python2.7/site-
packages/ endless_pagination/ static/ endless_pagination;
}
```

```
location /static/admin {
    autoindex on;
    expires 1w;
    alias /path/to/env/lib/python2.7/site-
packages/django/contrib/admin/ static/ admin;
}
}
```

EK-B

Aşağıda verilen tabloda çevrimdışı ortamdan, mobil uygulamadan ve öğrenme yönetim sisteminden elde edilen kayıtlara ilişkin XAPI verb ve XAPI Object ID bilgileri verilmiştir. Verilen sıra numarası veri madenciliği algoritmasında Deneyim numarası olarak kullanılmaktadır.

Tablo B.1. Elde edilen deneyim kayıtlarına ilişkin XAPI özellikleri

Sıra	Nesne	Değer
1	Verb	http://adlnet.gov/expapi/verbs/attended
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-1
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
2	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-1-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
3	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-1-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
4	Verb	http://adlnet.gov/expapi/verbs/completed
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-1-send-application
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
5	Verb	http://adlnet.gov/expapi/verbs/experienced
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-1-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
6	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-1-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdemesi (0-1 arasında)
7	Verb	http://adlnet.gov/expapi/verbs/attended
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-2
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
8	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-2-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)

Tablo B.1. (Devam) Elde edilen deneyim kayıtlarına ilişkin XAPI özellikleri

9	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-2-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
10	Verb	http://adlnet.gov/expapi/verbs/experienced
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-2-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
11	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-2-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdemesi (0-1 arasında)
12	Verb	http://adlnet.gov/expapi/verbs/passed
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-2-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Değerlendirme notu (0-100 arasında)
13	Verb	http://adlnet.gov/expapi/verbs/attended
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-3
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
14	Verb	http://id.tincanapi.com/verb/viewed
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-3-announcement
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
15	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-3-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
16	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-3-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
17	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-3_1-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)

Tablo B.1. (Devam) Elde edilen deneyim kayıtlarına ilişkin XAPI özellikleri

18	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-3_1-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
19	Verb	http://adlnet.gov/expapi/verbs/experienced
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-3-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
20	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-3-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
21	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-3_1-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
22	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-3_2-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
23	Verb	http://adlnet.gov/expapi/verbs/completed
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-3-send-application
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Değerlendirme notu (0-100 arasında)
24	Verb	http://adlnet.gov/expapi/verbs/attended
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-4
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
25	Verb	http://id.tincanapi.com/verb/viewed
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-4-announcement
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
26	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-4-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)

Tablo B.1. (Devam) Elde edilen deneyim kayıtlarına ilişkin XAPI özellikleri

27	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-4-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
28	Verb	http://adlnet.gov/expapi/verbs/experienced
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-4-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
29	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-4-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
30	Verb	http://adlnet.gov/expapi/verbs/completed
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-4-send-application
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Değerlendirme notu (0-100 arasında)
31	Verb	http://adlnet.gov/expapi/verbs/passed
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-4-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Değerlendirme notu (0-100 arasında)
32	Verb	http://activitystrea.ms/schema/1.0/present
	Object ID	http://www.erdemkismet.com.tr/statement-generator/present-topic
	Kaynak	Çevrimdışı Ortam
	Veri	Değerlendirme notu (0-100 arasında)
33	Verb	http://adlnet.gov/expapi/verbs/attended
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-5
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
34	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-5-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
35	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-5-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)

Tablo B.1. (Devam) Elde edilen deneyim kayıtlarına ilişkin XAPI özellikleri

36	Verb	http://adlnet.gov/expapi/verbs/experienced
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-5-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
37	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-5-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
38	Verb	http://adlnet.gov/expapi/verbs/attended
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-6
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
39	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-6-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
40	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-6-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
41	Verb	http://adlnet.gov/expapi/verbs/experienced
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-6-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
42	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-6-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
43	Verb	http://adlnet.gov/expapi/verbs/attended
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-7
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
44	Verb	http://id.tincanapi.com/verb/viewed
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-7-announcement
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)

Tablo B.1. (Devam) Elde edilen deneyim kayıtlarına ilişkin XAPI özellikleri

36	Verb	http://adlnet.gov/expapi/verbs/experienced
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-5-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
37	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-5-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
38	Verb	http://adlnet.gov/expapi/verbs/attended
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-6
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
39	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-6-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
40	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-6-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
41	Verb	http://adlnet.gov/expapi/verbs/experienced
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-6-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
42	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-6-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
43	Verb	http://adlnet.gov/expapi/verbs/attended
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-7
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
44	Verb	http://id.tincanapi.com/verb/viewed
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-7-announcement
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)

Tablo B.1. (Devam) Elde edilen deneyim kayıtlarına ilişkin XAPI özellikleri

45	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-7-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
46	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-7-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
47	Verb	http://activitystrea.ms/schema/1.0/read
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-7_1-reading
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
48	Verb	http://adlnet.gov/expapi/verbs/answered
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-7_1-questions
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
49	Verb	http://adlnet.gov/expapi/verbs/experienced
	Object ID	http://www.erdemkismet.com.tr/statement-generator/activity-7-experienced
	Kaynak	Çevrimdışı Ortam
	Veri	Deneyimlendi (1) veya Deneyimlenmedi (0)
50	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-7-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
51	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-7_1-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
52	Verb	http://activitystrea.ms/schema/1.0/watch
	Object ID	http://www.erdemkismet.com.tr/mobil/activity-7_2-course
	Kaynak	Mobil Uygulama
	Veri	İzleme yüzdesi (0-1 arasında)
53	Verb	http://adlnet.gov/expapi/verbs/completed
	Object ID	http://www.erdemkismet.com.tr/moodle/activity-7-send-application
	Kaynak	Öğrenme Yönetim Sistemi
	Veri	Değerlendirme notu (0-100 arasında)

EK-C

```
##### c4.5 Uygulaması #####
library("rJava")
library("RWeka")
egitimV$DURUM<-as.character(egitimV$DURUM)
egitimV$DURUM<-as.factor(egitimV$DURUM)
C45_model<-J48(DURUM~.,data=egitimV)
C45_model
show(C45_model)
GainRatioAttributeEval(DURUM ~., data=XAPI)
#elde edilen istatiksel bilgi
summary(C45_model)
#Gorsel
plot(C45_model)
#Dogrulugu kontrol etmek icin ortaya cikan agac predict() ile
tahminlenir.
kestirimC<-predict(C45_model,testV,type = "class")
show(kestirimC)
#Kestirim sonucu elde edilen veri,test verisi ile
karsilastirilir.KarisiklikMatrisi
karMatrisC<-table(testV$DURUM,kestirimC,dnn=c("KABUL","ÖNGÖRÜ"))
show(karMatrisC)
#Elde edilen karisiklik matrisinden dogruluk hesaplanir.
tp <- karMatrisC[1]
fp <- karMatrisC[3]
fn <- karMatrisC[2]
tn <- karMatrisC[4]
paste0("Dogruluk =", (dogrulukC <- (tp+tn) / sum(karMatrisC)))
paste0("Hata =", (hata<- 1-dogrulukC))
paste0("TPR =", (TPR <- tp/(tp+fn)))
paste0("DUYARLILIK =", duyarlilikC <- TPR)

paste0("SPC = ", (SPC <- tn/(fp+tn)))
paste0("PPV = ", (PPV <- tp/(tp+fp)))
paste0("KESİNLİK = ", PPV)
paste0("NPV = ", (NPV <- tn/(tn+fn)))
paste0("FPR = ", (FPR <- fp/(sum(karMatrisC))))
paste0("FNR = ", (FNR <- fn/(fn+tp)))
paste0("LR_p = ", (LR_p <- TPR/FPR))
paste0("LR_n = ", (LR_n <- FNR/SPC))
paste0("DOR = ", (DOR <- LR_p/LR_n))
paste0("F_measure =", (F_OlcutuC <- (2*PPV*TPR)/PPV+TPR))
```

```

##### Gini Uygulaması #####
install.packages("rpart")
library(rpart)
gini_model<-rpart(DURUM~.,data = egitimV,method =
"class",minsplit=4,parms = list(split="gini_model"))
show(gini_model)
#Elde edilen siniflandirma kurallarini agac yapisinde goruntulemek
#install.packages("rpart.plot")
library(rpart.plot)
prp(gini_model)
#Karışıklık Matrisi
kestirim<-predict(gini_model,testV,type = "class")
karMatrisG<-table(testV$DURUM,kestirim,dnn=c("KABUL","ÖNGÖRÜ"))
show(karMatrisG)
#Dogruluk hesaplama
tp <- karMatrisG[1]
fp <- karMatrisG[3]
fn <- karMatrisG[2]
tn <- karMatrisG[4]
paste0("Dogruluk =", (dogrulukG <- (tp+tn) / sum(karMatrisC)))
paste0("Hata =", (hata<- 1-dogrulukG))
paste0("TPR =", (TPR <- tp/(tp+fn)))
paste0("DUYARLILIK =", duyarlilikG <- TPR)

paste0("SPC =", (SPC <- tn/(fp+tn)))
paste0("PPV =", (PPV <- tp/(tp+fp)))
paste0("KESİNLİK =", PPV)
paste0("NPV =", (NPV <- tn/(tn+fn)))
paste0("FPR =", (FPR <- fp/(sum(karMatrisC))))
paste0("FNR =", (FNR <- fn/(fn+tp)))
paste0("LR_p =", (LR_p <- TPR/FPR))
paste0("LR_n =", (LR_n <- FNR/SPC))
paste0("DOR =", (DOR <- LR_p/LR_n))
paste0("F_measure =", (F_OlcutuG <- (2*PPV*TPR)/PPV+TPR))

```


KİŞİSEL YAYIN VE ESERLER

Kismet E., İner B., Real Time Report Application Basen On Learning Record Store, *International Conference on Education in Mathematics, Science and Technology (ICEMST)*, Marmaris, Türkiye, 28 Nisan – 1 Mayıs 2018.



ÖZGEÇMİŞ

Erdem Kısmet, 1990 yılında İstanbul'da doğdu. 2012 yılında Kocaeli Üniversitesi Bilgisayar Öğretmenliği bölümünden, 2015 yılında Sakarya Üniversitesi Bilgisayar Mühendisliği bölümünden mezun oldu. Aynı yıl içinde Kocaeli Üniversitesi Bilgisayar Mühendisliği Anabilim Dalı'nda yüksek lisans eğitimine başladı. Yüksek lisans eğitiminde eğitsel veri madenciliği konusunda çalışmaları bulunmaktadır.

