T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ÇEVRİMİÇİ REKLAMLARDA KULLANICININ SATIN ALMA SÜRECİNDE KANALLARIN ETKİSİNİN DERİN ÖĞRENME YÖNTEMLERİ İLE TESPİTİ

Oğuz KAHRAMAN

YÜKSEK LİSANS TEZİ
Bilgisayar Mühendisliği Anabilim Dalı
Bilgisayar Mühendisliği Programı

Danışman
Prof. Dr. M. Elif KARSLIGİL

Temmuz, 2022

T.C.
YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ÇEVRİMİÇİ REKLAMLARDA KULLANICININ SATIN ALMA SÜRECİNDE KANALLARIN ETKİSİNİN DERİN ÖĞRENME YÖNTEMLERİ İLE TESPİTİ

Oğuz KAHRAMAN tarafından hazırlanan tez çalışması 27.07.2022 tarihinde aşağıdaki jüri tarafından Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü Bilgisayar Mühendisliği Anabilim Dalı Bilgisayar Mühendisliği Programı **YÜKSEK LİSANS TEZİ** olarak kabul edilmiştir.

Prof. Dr. M. Elif KARSLIGİL
Yıldız Teknik Üniversitesi
Danışman

Jüri Üyeleri

Prof. Dr. M. Elif KARSLIGİL, Danışman
Yıldız Teknik Üniversitesi _____

Doç. Dr. M. Amaç GÜVENSAN, Üye
Yıldız Teknik Üniversitesi _____

Doç. Dr. Feza BUZLUCA, Üye
İstanbul Teknik Üniversitesi _____

Danışmanım Prof. Dr. M. Elif KARSLIGİL sorumluluğunda tarafımca hazırlanan Çevrimiçi Reklamlarda Kullanıcının Satın Alma Sürecinde Kanalların Etkisinin Derin Öğrenme Yöntemleri ile Tespiti başlıklı çalışmada veri toplama ve veri kullanımında gerekli yasal izinleri aldığımı, diğer kaynaklardan aldığım bilgileri ana metin ve referanslarda eksiksiz gösterdiğimi, araştırma verilerine ve sonuçlarına ilişkin çarpıtma ve/veya sahtecilik yapmadığımı, çalışmam süresince bilimsel araştırma ve etik ilkelerine uygun davrandığımı beyan ederim. Beyanımın aksinin ispatı halinde her türlü yasal sonucu kabul ederim.

Oğuz KAHRAMAN

Aileme
ve
tüm bilim dünyasına

TEŞEKKÜR

Bu çalışmanın gerçekleştirilmesinde çevrimiçi reklamcılık konusunda sağladıkları değerli bilgiler, altyapı ve finansal destek için Optdcom Teknoloji Yatırım AŞ'ye, proje test ve geliştirme aşamasında yardımcı olan Duygu Çobaner, Fuat Güçlüer ve Ceyla Yenilmez'e, veri kümesini sağlayan QSR Teknoloji AŞ (Lastikcim)'ye ve her türlü desteği sağlayan tez danışmanım Prof. Dr. M. Elif KARSLIGİL'e teşekkür ederim.

Oğuz KAHRAMAN

İÇİNDEKİLER

KISALTMA LİSTESİ vii

ŞEKİL LİSTESİ ix

TABLO LİSTESİ x

ÖZET xi

ABSTRACT xiv

1 GİRİŞ 1

2 İLGİLİ ÇALIŞMALAR 4

3 VERİ VE MEVCUT YAKLAŞIMLAR 8

3.1 Veri 8

3.2 Verinin Toplanması ve Saklanması 10

3.2.1 Divolte Collector İle Veri Toplanması 11

3.2.2 Kafka İle Verilerin Depolanması 11

3.2.3 Flume İle Verilerin Aktarılması 13

3.2.4 Hadoop İle Verilerin Depolanması 13

3.3 Veri Ön İşlemleri 14

3.3.1 Spark ile Veri Okunması ve Ön İşlemlerin Yapılması 14

3.4 Kullanıcı Tekilleştirme 15

3.5 Özellik Azaltımı 16

3.5.1 PCA ile Özellik Azaltma 16

3.5.2 Otokodlayıcı ile Özellik Azaltma 16

3.6 Verinin Etiketlenmesi 18

3.7 Reklam Kanal Etkisinin Mevcut Yöntemler ile Hesaplanması 18

3.7.1 İlk ve Son Tıklama İle Kanal Etkisinin Hesaplanması 18

3.7.2 Detaylı Uzman Analizi İle Kanal Etkisinin Hesaplanması 18

3.7.3 Mevcut Yaklaşımların Neden Olduğu Sorunlar 20

4 METODOLOJİ 21

4.1 Problem Tanımı 21
4.2 Model Tasarımı Öncesi Hazırlıklar 21
4.3 Derin Öğrenme Modeli Seçimi 23
4.3.1 RNN Mimarisi 23
4.3.2 LSTM Mimarisi 25
4.3.3 BiLSTM Mimarisi 27
4.4 Sistem Model Mimarisi 28

5 DENEYSEL SONUÇLAR **31**

6 SONUÇ VE ÖNERİLER **40**

KAYNAKÇA **42**

TEZDEN ÜRETİLMİŞ YAYINLAR **45**

KISALTMA LİSTESİ

AH	Additive Hazard
AMTA	Additional Multi-touch Attribution
ARNN	Attention Recurrent Neural Network
AUC	Area Under the Curve
BiLSTM	Bidirectional Long Short-Term Memory
BLR	Bagged Logistic Regression
BPTT	Backpropagation Trough Time
CPA	Cost Per Action
CPC	Cost Per Click
CR	Conversion Rate
CTR	Click-Through Rate
DARNN	Dual-Attention Recurrent Neural Network
DB	Database
GB	Gigabayt
HDFS	Hadoop Distributed File System
LR	Logistic Regression
LSTM	Long Short-Term Memory
KB	Kilobayt
MTA	Multi-touch Attribution
MSE	Mean Square Error
NN	Neural Network
RNN	Recurrent Neural Network
SP	Simple Probabilistic

SQL	Structured Query Language
YSA	Yapay Sinir Ağı

ŞEKİL LİSTESİ

Şekil 1.1 Kullanıcı satın alım yolculuğu örnekleri 2
Şekil 3.1 Verinin toplanması için kurulan büyük veri mimarisi 11
Şekil 3.2 Kafka mimarisi 12
Şekil 3.3 Flume mimarisi 13
Şekil 3.4 HDFS mimarisi 14
Şekil 3.5 Spark mimarisi 15
Şekil 3.6 Klasik otokodlayıcı mimarisi 17
Şekil 3.7 LSTM otokodlayıcı mimarisi 17
Şekil 4.1 RNN (Yinelemi Sinir Ağı) mimarisi 23
Şekil 4.2 RNN nöron mimarisi) mimarisi 24
Şekil 4.3 Sigmoid fonksiyon ve türevi 25
Şekil 4.4 Tekil bir kullanıcı için örnek satın alma yolculuğu 25
Şekil 4.5 LSTM hücre mimarisi 25
Şekil 4.6 BiLSTM mimarisi 27
Şekil 4.7 Kanalların satın almaya etkisinin tespiti için önerilen model 28
Şekil 4.8 Özellik azaltma için oluşturulan PCA modeli 29
Şekil 4.9 Özellik azaltma için oluşturulan otokodlayıcı modelleri 29
Şekil 4.10 Her kanalın satın almaya etkisinin hesaplanması 29
Şekil 5.1 LSTM tabanlı otokodlayıcı, klasik otokodlayıcı ve PCA ile özellik azaltımı sonucu elde edilen hata oranları 33
Şekil 5.2 BiLSTM ve LSTM modellerinin karşılaştırılması. 35
Şekil 5.3 LSTM tabanlı otokodlayıcı kullanan derin öğrenme modeli için reklam kanallarının etkisine göre bütçe dağılımı 36
Şekil 5.4 Özellik azaltımı kullanmayan derin öğrenme modeli için reklam kanallarının etkisine göre bütçe dağılımı 37
Şekil 5.5 Kanallara göre otokodlayıcı olan ve olmayan modeller için veri boyutu bazlı hata grafiği 38
Şekil 5.6 60 günlük veri üzerinde yapılan reklam kanallarının bütçe dağılımı . 39

TABLO LİSTESİ

Tablo 3.1 One-Hot encoding yapılan özellikler ve elde edilen yeni giriş özellikleri 9
Tablo 3.2 Reklam kanallarından satın alım yapan kullanıcı sayıları ve toplam kullanıcı sayıları 10
Tablo 4.1 Karşılaştırmalı model çıktıları 22
Tablo 5.1 Modelin eğitimi için kullanılan deney ortamı 31
Tablo 5.2 İlk ve son tıklanan reklam kanalının hesaplanması sonucu elde edilen sonuçlar 32
Tablo 5.3 Uzmanlar tarafından yapılan hesaplamalar sonucu elde edilen reklam kanalı etkileri 32
Tablo 5.4 Veri ve iterasyon sayısı 34
Tablo 5.5 PCA, otokodlayıcı ve özellik azaltımı kullanılmadan oluşturulan modeller için 60 günlük veri ile eğitilen model karşılaştırması 36
Tablo 5.6 PCA ve otokodlayıcı hata karşılaştırması 37

ÖZET

Çevrimiçi Reklamlarda Kullanıcının Satın Alma Sürecinde Kanalların Etkisinin Derin Öğrenme Yöntemleri ile Tespiti

Oğuz KAHRAMAN

Bilgisayar Mühendisliği Anabilim Dalı
Yüksek Lisans Tezi

Danışman: Prof. Dr. M. Elif KARSLIGİL

Çevrimiçi reklamcılık için kaynakların efektif şekilde dağıtılması yayıncılar için önemli bir problemdir. Şu anki durumda, çok noktalı etkileşim ve etkileşimlerin etkisinin hesaplanması bütçe dağılımı yapılırken karşılaşılan en büyük sorunlardan bir tanesidir. Çok noktalı etkileşim ile farklı reklam kanalları temsil edilmektedir. Bu nedenle, müşterilerin farklı kanallarda görüp ilgilendikleri ürünleri satın aldıklarında diğer kanalların satın almaya etkisi tespit edilememekte ve bu durum sonucunda reklam verenin, reklam bütçesini nasıl paylaştıracağı tam olarak hesaplanamamaktadır. Bu sorunu çözmek için mevcut yaklaşımlar, uzman kişiler tarafından değerlendirme yapılarak bütçenin kaynaklara dağıtılması ya da basit istatistiksel yöntemler kullanılması yönündedir. Mevcut yaklaşımlar uygulanırken karşılaşılan en önemli sorun veri boyutunun çok büyük olması ve mevcut araçların tüm veriyi toplayamamasıdır. Veri boyutu sebebiyle insan gücü ile yapılan analizler çok kısıtlı kalmakta ve birçok ayrıntı gözden kaçırılmaktadır. İstatistiksel çıktılar almak ise haftalar gerektirmektedir. Diğer bir sorun ise, eğer kullanıcılar farklı reklam kanallarından farklı zamanlarda sistem ile etkileşime geçerse, bu etkileşimlerin tüm sürece etkisi optimal bir biçimde saptanamamaktadır. Bu saptamanın yapılamamasının temel nedeni olarak etkileşimlerin çok uzun bir zaman aralığına yayılması, mevcut analiz araçlarının tüm veriyi depolayamaması ve mevcut sistemlerin ise çıktılarını son etkileşim noktasına göre oluşturmasıdır. Uzmanlar tarafından yapılan kanal etki analizi sonuçları uzmanın bilgi düzeyine göre değişmektedir. Kanal etki analizi yapılırken kullanılan somut bir yöntem olmaması bu sorunun oluşmasına

neden olmaktadır.

Bu tez çalışmasında, çevrimiçi reklamlarda farklı reklam kaynaklarının kullanıcının satın almasına etkisini tespit etmek amacıyla bir sistem gerçeklenmiştir. Kullanıcı web sitesi verilerini toplayıp işleyebilmek için web sitesi entegrasyonu sağlanmıştır. Web sitesi üzerinden alınan verilerin saklanması için büyük veri ortamı oluşturulmuştur. Büyük veri ortamına verilerin aktarılması için dağıtık mimariye uygun mesaj akışı yapabilecek sistem kurulmuştur. Toplanan veriler üzerinde ön işlemler uygulanmıştır. Verilerin doğru etiketlenebilmesi ve model başarısını arttırmak için kullanıcı tekilleştirme işlemi yapılmıştır. Kullanıcı tekilleştirme işlemi, farkı cihazlardan gelen kullanıcıların eşleştirilmesini ifade etmektedir. Bu işlemler sonucu toplam 50GB büyüklüğünde ve 2 aylık kullanıcı web sitesi hareketlerini içeren, 65 özellik ve 5 çıkış sınıfı olan veri kümesi elde edilmiştir. Giriş özellikleri kullanıcı reklam ve kampanya bilgileri, satın alım yapılıp yapılmadığı ve sistem ile ilgili bilgileri içermektedir. Çıkış sınıfları ise reklam kanallarıdır. Veri etiketleme için oturum bazlı olarak reklam kanal bilgisi bulunmuş, bu reklam kanal bilgisi ile etiketleme işlemi yapılmıştır. Reklam kanal etki tespit sistemi için, toplanan veriyi işleyebilecek LSTM(Long short-term memory- Uzun-kısa süreli bellek) tabanlı bir derin öğrenme modeli tasarlanmış ve gerçeklenmiştir. LSTM modeli seçilmesinin sebebi zaman serisi veriler üzerinde analizler yapmak üzere geliştirilmiş olmasıdır. İçinde barındırdığı bellek yapısı sayesinde geçmiş verilerden hangisinin değerli olup olmadığını tespit edebilmektedir. Özellik çıkarımı için otokodlayıcı kullanılmıştır.

Otokodlayıcı ile özellik çıkarımı olan LSTM modeli, PCA ile özellik çıkarımı olan LSTM modeli ve özellik çıkarımı olmayan LSTM ve BiLSTM modelleri eğitilmiştir. Tüm modeller için 15, 25, 45 ve 60 günlük veriler ile eğitim yapılmıştır. Eğitimler sonucu en başarılı modelin otokodlayıcı kullanan LSTM modeli olduğu gözlemlenmiştir. Tüm modeller için başarı kıstası olarak uzmanlar tarafından hesaplanan kanal etki analiz sonuçları kullanılmıştır. Eğitim sonucunda zaman aralığına ve reklam kanalına bağlı olarak %8 ve %4 aralığında hata oranı ile sonuçlar elde edilmiştir. Sonuçlar değerlendirildiğine 15 ve 60 günlük verilerde beklenen duruma en yakın sonuçlar alınmıştır. Bu durum veri kümesi üzerinde sezonsal değişikliklerin olduğunu göstermiştir.

Eklenen otokodlayıcı modeli bu tez kapsamında yapılan çalışmanın daha önce yapılmış olan LSTM destekli modellerden farklılaşmasını sağlamıştır.

Geliştirilen modelin eğitiminden sonra çıktıların anlık alınabilmesi sağlanmıştır. Haftalar süren kanal etki analizi işleminin anlık alınabilmesi bu sorunun çözümüne önemli katkı sağlamıştır. Sabit bir modelin kullanılması, farklı uzmanlar tarafından

alınan farklı sonuçların da oluşmasını önleyerek tek bir model aracılığı ile kanal etki analiz sonuçlarının alınmasını sağlamıştır.

Anahtar Kelimeler: Uzun-Kısa süreli bellek, çok noktalı etkileşim, çevrimiçi reklamcılık, reklam kanallarının satın almaya etkisi

ABSTRACT

Detection of Effect of Channels on User's Purchasing Process in Online Advertisements by Deep Learning Methods

Oğuz KAHRAMAN

Department of Computer Engineering
Master of Science Thesis

Supervisor: Prof. Dr. M. Elif KARSLIGİL

In online advertising, multi-touch attribution and distributing credits is a big challenge for publishers. In the current situation, multi-touch attribution and calculating the effect of interactions is one of the biggest problems encountered while making budget allocations. Different advertising channels are represented by multi-touch attribution. For this reason, when customers see and buy products that they are interested in in different channels, the effect of other channels on purchasing cannot be determined, and as a result, it cannot be calculated exactly how the advertiser will allocate the advertising budget. The current approaches to solving this problem are distributing the budget to the resources after evaluation by experts or using simple statistical methods. The most important problem encountered while applying the existing approaches is that the data size is too large. Due to the size of the data, analyses made with human power are very limited and many details are overlooked. Statistical outputs require weeks for getting results. Another problem is that if users interact with the system from different advertising channels at different times, the effect of these interactions on the whole process cannot be determined effectively. The main reason why this determination could not be made is that the interactions spread over a very long-time interval, the existing analysis tools cannot store all the data, and the existing systems create their outputs according to the last interaction point. The results of the channel impact analysis made by the experts can be change according to the knowledge level of the expert. The main reason of this problem is that there is no concrete method for channel effect analysis.

In this thesis, a system has been implemented for determining the effect of different advertising sources on the user's purchase in online advertisements. Website integration has been provided for collect and process user website data. Big data environment has been implemented to store website data. In order to transfer data to big data environment, a system has been established which can flow messages in accordance with a distributed architecture. Pre-processes has been applied on the collected data. In order to label the data correctly and to increase the success of the model, user singularization process has been applied. User singularization process refers to pairing users who connect website from different devices. As a result of these processes, a dataset of 65 features and 5 output classes was obtained, with a total size of 50GB and containing 2 months of user website movements. The input features contain advertisement and campaign information, conversion information and system information. Output classes are advertising channels. In order to data labeling, session-based advertising channel information has been found and tagging has been performed with this advertising channel information. A deep learning model based on LSTM (Long short-term memory) that can process the collected data has been designed and implemented for the advertising impact detection system. The reason for choosing the LSTM model is that it has developed to analyze time series data. It can determine which historical data is valuable or not with memory structure. Autoencoder has been used for feature extraction.

LSTM model with feature extraction with autoencoder, LSTM model with feature extraction with PCA, and LSTM and BiLSTM models without feature extraction has been trained. All models have been trained with 15, 25, 45 and 60 days data. Training results has been showed that LSTM model has performed best performance which use autoencoder for feature extraction. Channel effect analysis results calculated by experts has been used as success criteria for all models. As a result of the training, results have been obtained with an error rate between %8 and %4 depending on the time interval and the advertising channel. When the results were evaluated, the results closest to the expected situation has been obtained in the 15 and 60-day data. This situation has showed that there are seasonal changes on the dataset.

The added autoencoder model has been differentiated the work done within the scope of this thesis from the previous LSTM supported models.

After the training of the developed model, it has been ensured that the outputs could be taken instantly. The fact that the channel impact analysis process, which took weeks, could be taken instantly, made a significant contribution to the solution of this problem. The use of a fixed model prevented the occurrence of different results obtained by different experts, and enabled channel impact analysis results to be

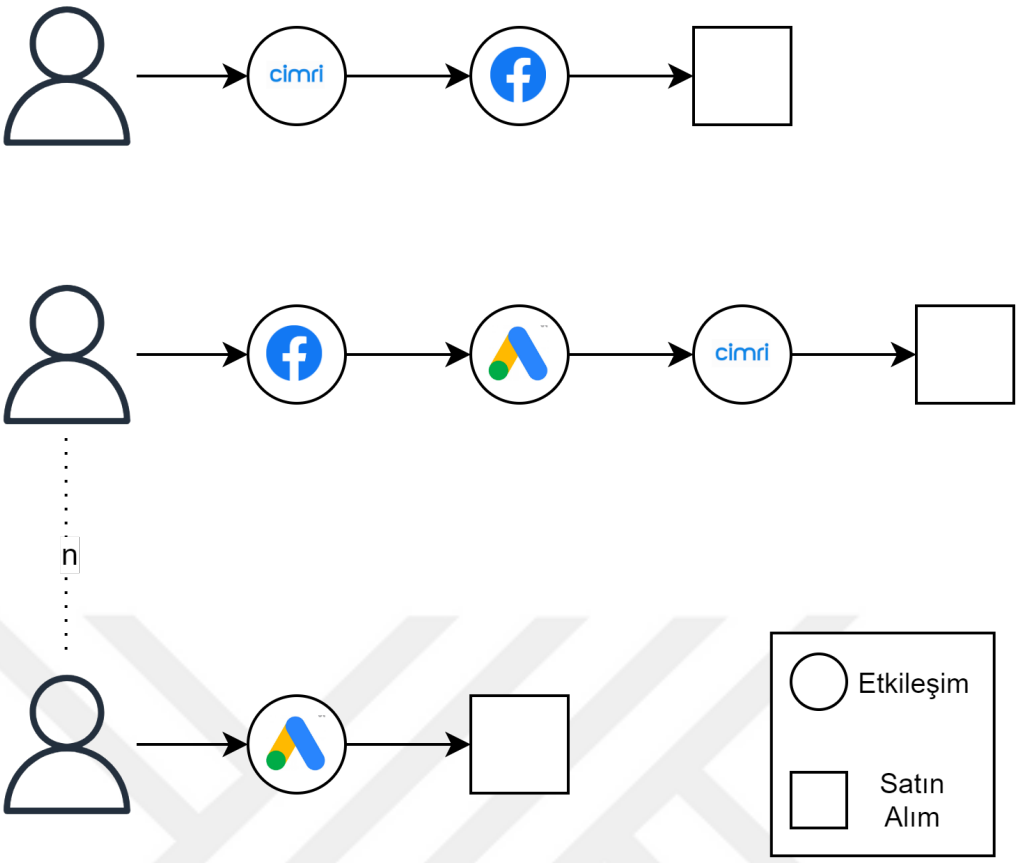
obtained through a single model.

Keywords: Long short-term memory, multi-touch attribution, online advertising, effect of advertising channels on purchasing

1
GİRİŞ

Kullanıcılara e-posta pazarlaması, dijital reklam panoları, arama motoru gibi farklı dijital reklam kanalları üzerinden reklamlar sunulur. Kullanıcı ya sayfadaki reklama tıklayarak doğrudan satışa giden yolu izler ya da yarıda keserek gün veya hafta mertebesinde bir süre sonra satışla sonuçlanacak şekilde tekrar geri dönüş yapar. Diğer taraftan kullanıcının ilgili reklama tıklamadığı, ancak bir süre sonra bu reklamın etkisi ile kendiliğinden reklamı verdiren firmanın çevrimiçi satış sayfası üzerinden satın almayı gerçekleştirdiği senaryo da mevcuttur. Şekil 1.1'de müşterilerin farklı zamanlarda farklı kanalları gezdikten sonra satın alma ile sonuçlanan hareketlerine örnekler görülmektedir. Şekilde gösterilen tüm etkileşimler, doğrudan alışverişin yapılacağı site ile etkileşime geçilen durumları temsil etmektedir. Pazarlama şirketleri, farklı kanallara reklam verme stratejilerini belirlerken her kullanıcının satın alma yolculuğunu izlemek ve her bir pazarlama kanalının etkisini ölçmek için değişik yaklaşımlar kullanmaktadırlar. Bu süreç genel olarak elle yapılmaktadır. Her kampanya veya kanalın dönüşüm üzerindeki etkileşiminin hesaplanması problemi, iç görü problemi olarak adlandırılmaktadır [1]. Kullanıcı işlemler sonucunda bir satın alım yapar ise dönüşüm olarak değerlendirilmektedir. En iyi kaynak paylaşımını ve kazancı sağlamak için reklam yayıncılarının milyonlarca veriyi, günler içinde analiz etmeleri gerekmektedir. Diğer bir problem ise, kullanıcılar bu kanallar ile çok farklı zamanlarda etkileşime geçebildikleri için bu durumlar bağımsız davranışlar gibi gözükmektedir. El yordamı ile en efektif sonucu bulmak mümkün olamamaktadır ve çapraz kanal geçişleri tespit edilememektedir. Bu sebepten dolayı reklam bütçesi dağıtımı konusunda en efektif çözümü bulmakta zorlanmaktadırlar.

Bu sorunu çözmek için birçok farklı çalışma yapılmıştır. Bu çalışmalar istatistiksel ve derin öğrenme temelli olmak üzere 2 kategori altında toplanabilir. İstatistiksel çalışmalar ile veri kümesine özgü formüller ve matematiksel çıkarımlar yapılması hedeflenmiştir. Bu yaklaşımlar el yordamı ile yapılan değerlendirmelerin daha efektif ve hızlı hale getirilmesini hedeflemiştir. İstatistiksel yöntemler ne kadar işi kolaylaştırsa da hem artan reklam kanalı sayısı hem de artan veri boyutu büyüklüğü



Şekil 1.1 Kullanıcı satın alım yolculuğu örnekleri

karşısında çok efektif olamadıkları görülmüştür. Derin öğrenme temelli yöntemler veri büyüklüğünden kaynaklı sorunları çözmede efektif olmuşlardır.

Derin öğrenme temelli modeller ile büyük veri üzerinde çalışmalar yapılmaya başlanmıştır. Bu çalışmalar ile hem farklı zaman aralıkları hem de kampanya tarzlarına göre çıkarımlar yapılmıştır. Elde edilen sonuçlar sayesinde zaman ve veri kaynaklı sorunlar giderilmiştir. Modellerin çalışma performanslarının geliştirilebilmesi için dikkat(attention) mekanizması kullanılmıştır. Daha çok özellik çıkarımı için kullanılan bu mekanizma ile geniş veri kümesi üzerinde daha hızlı çalışma sağlanabilmiştir. Yapılan çalışmaların büyük kısmı ise derin öğrenme modeli olarak RNN (Recurrent Neural Network- Yinelemeli Sinir Ağı) veya LSTM (Long Short-Term Memory-Uzun-Kısa Süreli Bellek) modelleri üzerine geliştirilmiştir.

Bu çalışmada çevrimiçi alışverişte farklı dijital reklam kanallarının müşterinin satın almasına etkisini değerlendirebilmek için, LSTM tabanlı bütçe dağılımı yapan bir derin öğrenme modeli tasarlanmış ve gerçeklenmiştir. LSTM tabanlı bir mimariye karar verilmesinin önemli sebeplerinden biri kullanıcı reklamı görüp ilk defa siteye giriş yapsa bile çoğu durumda satın almadan önce herhangi bir reklama tıklamadan direk site adresi üzerinden sisteme gelip alışveriş yapmaktadır. Bu durum da reklam

kanalının etkisinin göz ardı edilebilmesine sebep olmaktadır. Bu nedenle uzun süreli kullanıcı hareketlerini ve bu hareketler sonucu reklam etkisini saptayabilecek ve zaman serisi analizlere uygun LSTM tabanlı bir mimari gerçeklenmiştir. Mevcut çalışmalardan farklı olarak özellik çıkarımı için otokodlayıcı modeli kullanılmıştır. Bu alanda yapılan çalışmada ilk defa otokodlayıcı modeli kullanılmıştır. Modelin tahminleme performansını önemli ölçüde arttırdığı gözlemlenen yeni mimarinin gelecek çalışmalarda önemli bir çıkış noktası olması hedeflenmektedir.

Tezin 2. bölümünde literatürdeki benzer çalışmalar anlatılmıştır. 3. bölümde veri kümesi hazırlanması aşaması detaylandırılmıştır. Bu aşamada kullanılan özellik seçimi ve çıkarımı yöntemleri hakkında bilgiler verilmiştir. 4. bölümde model mimarisi anlatılmış ve otokodlayıcı ile modelin entegrasyonu hakkında bilgi verilmiştir. 5. bölümde farklı senaryolar için elde edilen sonuçlar gösterilmiştir. Son bölümde ise gelecekte yapılabilecek çalışmalar ve sonuçlar anlatılmıştır.

2
İLGİLİ ÇALIŞMALAR

Çevrimiçi reklamcılıkta, kullanıcıların reklamlara tıklaması sonucu satın alım yapması dönüşüm olarak adlandırılmaktadır. Dönüşüm oranları ise genellikle elle ve kural tabanlı metotlar kullanılarak yapılan hesaplamalara dayanmaktadır. Dönüşüm oranı ile kullanıcının bir reklama tıkladıktan sonra satın alım yapıp yapmadığı, yaptıysa kaç kullanıcının bu satın alımı yaptığı ifade edilmektedir. Berman'ın [2] önerdiği çevrimiçi reklam kampanyaları için bütçe dağılımı modeli, reklam kampanyalarının satın alım üzerindeki etkinliğinin hesaplanıp buna göre bütçenin dağıtılmasını öngörmektedir. Veri kümesi gizli olduğu için detayları paylaşılmamıştır. Elde edilen sonuçlarda %90 oranlarında efektif bütçe dağılımının yapıldığı gözlemlenmiştir. Bu sonuç için mevcut bütçe dağılımları ile karşılaştırmalar kullanılmıştır. Xuhui Shao ve arkadaşlarının [3] önerisi veri güdümlü olarak lojistik regresyon modeline dayanmaktadır. Yaptıkları önerme iki ana aşama içermektedir. İlk aşamada, kullanıcıları pozitif ve negatif olarak dönüşüm biçimine göre kümelemektedirler. İkinci aşamada ise, hangi kullanıcı için hangi reklam kaynağından daha fazla dönüşüm olabileceği hesaplanmakta, buna göre kümeleme yapılmaktadır. Bu çalışma için 300GB büyüklüğünde toplam 72,5 milyon farklı kullanıcının ve 39 farklı reklam kanalının olduğu erişime kapalı bir veri kümesi kullanılmıştır. Bu veri kümesinde kullanıcıların 4 haftalık reklam etkileşimleri kullanılmıştır. Bu çalışma ile kesin sonuçlar bulmak amaçlanmadığı için karşılaştırmalı çıktılar sunulmuştur. Sonuçlar incelendiğinde BLR modelinin LR modelinden daha iyi olduğu gözlemlenmiştir. Raghav ve arkadaşlarının [4] çalışmasında ise bir markov model önermiştir. Bu model ile farklı kullanıcı aşamaları için hesaplamalar yapılmıştır. Bu hesaplamalar için benzersiz-tek biçimli (unique-uniform) adını verdikleri bir şema oluşturmuşlardır. Bu model için milyonlarca farklı kullanıcı etkileşim yolu içeren kapalı bir veri kümesi kullanmışlardır. Sonuçlar ise sayısal olarak değil formüller ile verilmiş, bu nedenle karşılaştırmaya uygun sonuçlar elde edilememektedir. Veri kümesi gizli ve detayları paylaşılmamıştır. Brian ve arkadaşlarının [5] yaptığı çalışmada ise kullanıcının ilgili reklamı gördükten sonra reklamın referans verdiği siteyi de görüntüleme oranını tespit etmeye çalışmaktadırlar. Burada amaç reklama tıklamadan reklam görüntüleme

sonucu siteye gelen müşterilerin tespitidir. 1000 kullanıcıya ve 400bin görüntülemeye varan erişime kapalı bir veri kümesi kullanılmıştır. 2 haftalık etkileşimlerin olduğu bir veri kümesidir. Çalışma sonucunda kullanıcıların %5'inin siteyi görüntülediği fakat doğru reklam kampanyaları ile bu oranın %25'lere kadar çıkabildiği gözlemlenmiştir. Bu tür yaklaşımların genel sorunu büyük veri kümesinin analizinin zor olması ve birbiri arası etkileşimleri tam temsil edememesidir.

Diğer bir yaklaşım ise sağkalım teorisine (survival theory) [6] dayalı çözümlerdir. Bu teori çerçevesinde geliştirilen modellerde, belirli bir zamanda hangi reklam kaynağı ile etkileşimin meydana geleceğinin tespit edilmesi için öngörüye dayalı analiz modelleri kullanılmaktadır. Bunun için Wendi Ji ve arkadaşlarının [7, 8] geliştirdiği modeller ile Ya Zhang ve arkadaşlarının [9] geliştirdiği model örnek gösterilebilir. Bu modeller için 1,24 milyar kayıt içeren bir veri kümesi kullanılmıştır. Veri kümesi olarak Miaozhen System[1] tarafından açık kaynak olan bir veri kümesi kullanılmıştır. Kullanılan veri kümesi 3 aylık kullanıcı etkileşimlerini içermektedir. 46 farklı türde 2605 farklı reklam kanalı içermektedir. Elde edilen sonuçlar incelendiğinde reklam kanalından çok reklam türünün model performansını etkilediği görülmüştür. Bazı reklam türlerinde beklenen durum ile tahmin edilen arasında %90'lara varan benzerlik varken, bazı reklam türlerinde ise hata oranı %50'leri bulmaktadır. Bu tür yaklaşımların genel sorunu çok noktalı etkileşim yerine tek bir noktadaki etkileşime odaklanmasıdır.

Bu konuyla ilgili yapay sinir ağlarına dayanan modeller de geliştirilmiştir. Ning Li ve arkadaşlarının [10] önerdiği model ile farklı reklam kaynaklarından gelen kullanıcılar hedeflenmiştir. Bunun için en uygun reklam kaynağı seçilmesi hedeflenmiştir. LSTM tabanlı olan bu modeli DNAMTA (Deep Neural Net With Attention Multi-Touch Attribution Model) olarak tanımlamışlardır. LSTM tabanlı modelin geliştirilmesi için öncesine attention [11] yapısı ile desteklenmiştir. Özellik çıkarımı için kullanılan mekanizma ile sistem performansının arttırılması hedeflenmiştir. Bu çalışmada e-posta, web ve telefon gibi reklam kaynaklarına odaklanılmıştır. 426 bin kayıt ve 6 farklı kullanıcı etkileşim noktası olan bir veri kümesi kullanılmıştır. Bu veri kümesi 57 günlük kayıt içermektedir. Her kayıt bir kullanıcı yolculuğunu temsil etmektedir. Karşılaştırmalı çıktılarda DNAMTA modelinin LR modelinden daha başarılı olduğu gözlemlenmiştir. Başarı ölçütü olarak AUC değeri kullanılmış ve %87 olarak hesaplanmıştır. Bu değer hesaplanırken etiketli veriler ile eğitim ve test yapılmıştır. Hejazinia ve arkadaşları [12] DAN (Deep Average Network) adlı YSA (Yapay Sinir Ağı) modelini önermişlerdir. Bu çalışma, kullanıcı bazlı reklam hedefleme için geliştirilmiş, kullanıcıları kümeleyerek hedefe uygun reklam gösterilmesi sağlanmıştır. Milyonlarca farklı kullanıcı içeren veri kümesi 7 günlük bir periyotta toplanmıştır. %88 AUC değeri

[1] `https://apex.sjtu.edu.cn/datasets/13`

başarı ölçütü olarak kullanılmıştır. Bu değer hesaplanırken etiketli veriler ile eğitim ve test yapılmıştır. Dongdong Yang ve arkadaşları [13] ise LSTM ve lineer regresyon tabanlı DeepMTA olarak adlandırdıkları bir model geliştirmişlerdir. Bu çalışmada farklı reklam kaynaklarından gelen kullanıcılar için dönüşüm oranı hesaplanmıştır. 10 bin kullanıcı ve 742 bin kayıt içeren bir veri seti kullanılmıştır. 10 günlük kullanıcı kayıtlarını içermektedir. Başarı kriteri olarak ise AUC değeri kullanılmış ve bu değer %91 olarak elde edilmiştir. Ayrıca kullanıcıların son tıklama oranları ile de karşılaştırmalar yapılmıştır. Bu karşılaştırmalar sonucu yine %85'in üzerinde başarı alındığı görülmüştür. Bu konu üzerinde yapılan diğer bir önemli çalışma ise Kan Ren ve arkadaşları [14] tarafından yapılmıştır. Bu model RNN (Recurrent Neural Network-Yinelemeli Sinir Ağı) temelli olarak geliştirilmiştir. LSTM tabanlı modelin geliştirilmesi için öncesine attention [11] yapısı ile desteklenmiştir. Özellik çıkarımı için kullanılan mekanizma ile sistem performansının arttırılması hedeflenmiştir. Çift katmanlı olarak kullanılan attention mekanizması ile mevcuttaki çalışmalardan farklı bir yola gidilmiştir. Model kullanılarak kullanıcıların dönüşümlerinde reklam kanallarının etkisi hesaplanmıştır. Veri kümesi olarak iki farklı kaynak kullanılmıştır. İlk kaynak 1,24 milyar kayıt ve 59 milyon kullanıcı içermektedir. Miaozhen System[1] tarafından açık kaynak olan bir kaynak kullanılmıştır. 3 aylık kullanıcı etkileşimlerini içermektedir. 46 farklı türde 2605 farklı reklam kanalı içermektedir. Diğer kaynak ise 16 milyon kayıt içermektedir. Criteo[1] kaynağından alınan veri kümesi ise 30 günlük kullanıcı hareketi ve 700 kampanya içermektedir. Modelin başarı değeri olarak AUC kullanılmış ve %91 değeri elde edilmiştir. Bu değer hesaplanırken etiketli veriler ile eğitim ve test yapılmıştır. Thorton ve arkadaşlarının [15] çalışmasında ise kullanıcı seyahati pazarlama ve son etkileşim noktası açısından değerlendirilip en iyi özellik tercihinin yapılmasını hedeflenmektedir. Bu sayede bu tür çalışmalar için ortak bir özellik seçim yöntemi uygulanarak bir standart getirilmesi hedeflenmektedir. Bu çalışma için 319 farklı özelliğe sahip 10 milyon kayıt kullanılmıştır.

Son olarak, karşılaştırmalı sonuçlar içeren çalışmalardan da yararlanılabilir. Bunun için Hana Choi ve arkadaşlarının [16] hazırladığı yayın içinde birçok farklı türde yapılan çalışmalar başarılı bir biçimde incelenmiştir. İncelemeler birçok farklı kategorileri ayrılmış ve bu sayede ihtiyaca yönelik yayınların ve araçların daha kolay bulunması sağlanmıştır. Yapılan inceleme alanları reklam verenler, kullanıcılar ve kullanıcı karakteristiği gibi birçok farklı başlık altında değerlendirilmiştir.

Tüm bu çalışmalar sonucunda hem reklam kanalları hem de kanallar arası karşılaştırmalı sonuç çıkaracak bir sistemin eksikliği görülmüştür. Bu çalışmada kanalların satın alım süreci üzerinde karşılaştırmalı etkisini de ortaya çıkaracak bir model tasarlanmış ve gerçeklenmiştir. Bu tasarım için DeepMTA [13] modeli esin kaynağı olarak seçilmiş ve model oluşturulmuştur. Sistemin performansını arttırmak

için mevcuttaki çalışmalar dikkat(attention) mekanizmasına yönelmiştir.

Bu tez çalışmasında otokodlayıcı ile özellik çıkarımı yapan LSTM tabanlı model gerçeklenmiştir. Mevcut LSTM tabanlı çalışmalardan farklı olarak özellik çıkarımı için otokodlayıcı kullanılmıştır.

3
VERİ VE MEVCUT YAKLAŞIMLAR

Dijital reklam kanallarının müşterinin satın almaya etkisinin tespitine yönelik modeli geliştirirken veri kümesi hazırlanması için Lastikçim markası ile ortak bir çalışma gerçekleştirilmiştir. Bu çalışma çerçevesinde verinin toplanması için büyük veri araçları kullanılmıştır. Bu araçlar sonucu alınan veri üzerinde ise son olarak veri temizleme, one-hot encoding ve özellik azaltımı işlemleri yapılmıştır.

3.1 Veri

Bu tez çalışması için Lastikcim[1] markası ile ortak bir çalışma gerçekleştirilmiştir. Bu çalışma ile web sitesi üzerinden kullanıcının site içi hareketlerini ve siteye geldiği kaynak bilgisini içeren veri toplanmıştır. Bu veri kullanıcının site içi fare hareketleri, site içi yönlendirmeler ve gezdiği sayfalar, siteye geldiği kaynak ve kullanıcı bazında satın alım olup olmadığı gibi bilgileri içermektedir. Burada toplam 65 farklı özelliğe ve 5 sınıfa sahip bir veri kümesi hazırlanmıştır.

Veri kümesi içindeki özelliklerden 65 tanesi giriş özelliği olarak kullanılmaktadır. Bu özelliklerden 8 tanesi işletim sisteminin, 4 tanesi cihaz tipini, 4 tanesi siteye gelen trafik tipini, 17 tanesi tarayıcı tipini, 7 tanesi browser üreticisini, 10 tanesi fare hareket tipini one-hot encoding uygulanmış biçimini temsil etmektedir. Tablo 3.1'de bu özellikler detayları ile gösterilmiştir. One-hot encoding uygulanmayan özellikler için, 6 tanesi ekranda yapılan hareketin konumunu, 4 tanesi reklam kampanyasını, 2 tanesi cihazın ekran boyutunu, bir tanesi oturumun ilk giriş noktası olup olmadığını, bir tanesi tarihi nitelemektedir ve bir tanesi de satın alıma dönüşüp dönüşmediğini temsil etmektedir.

Veri kümesinde çıkış olarak kullanılan 5 sınıftan her biri bir reklam kaynağını temsil etmektedir.

Toplanan verilerin boyutu yaklaşık olarak 50gb seviyesindedir ve 60 günlük, 27 Eylül

[1] https://www.lastikcim.com.tr/

Tablo 3.1 One-Hot encoding yapılan özellikler ve elde edilen yeni giriş özellikleri

Özellik Türü	One-Hot Encoding Sonucu
İşletim Sistemi Bilgileri	Linux Windows Android OsX iOS Microsoft Google Apple
Cihaz Tipi	Kişisel Bilgisayar Akıllı Telefon Tablet Diğer
Trafik Tipi	Kişisel Bilgisayar Mobil Robot E-postaİstemcisi
Tarayıcı Tipi	Opera Mobile Mobile Safari Chrome Internet Explorer Mozilla Firerfox Safari Yandex Browser Opera Mobile Firefox Chrome Mobile Android Browser BingBot AdsBotGoogle AppleMail GoogleBot Bilinmeyen
Tarayıcı Üreticisi	Microsoft Opera Apple Yandex Mozilla Google CloudMosa
Fare Hareket Tipi	Change Keydown Click Mouseenter Mouseleave Touchstart TouchEnd TouchCancel Blur Bilinmeyen

2021 ve 26 Kasım 2021 tarih aralığını kapsayan kullanıcı hareketlerini içermektedir. Bu kullanıcı hareketleri 20 milyon satır ve 270 bin tekil kullanıcıyı temsil etmektedir. Veri içindeki reklam kanalları ise DonanimHaber, Adwords, Cimri, Facebook ve Instagram'dır. Bu kanallardan gelen toplam ve satın alım yapan kullanıcı sayıları Tablo 3.2'de gösterilmiştir. Dikkat edilmesi gereken nokta, bir kullanıcı siteyle etkileşime geçerken birden fazla kanalı kullanabilmektedir. Temsil adları sonuçlar bölümünde kullanılacak isimlerini ifade etmektedir.

Tablo 3.2 Reklam kanallarından satın alım yapan kullanıcı sayıları ve toplam kullanıcı sayıları

Reklam Kanalı	Temsil Adı	Toplam Kullanıcı	Satın Alım Yapan Kullanıcı
Adwords	Kanal 1	1614471	107508
Cimri	Kanal 2	771798	116472
DonanimHaber	Kanal 3	1614471	0
Facebook	Kanal 4	1523465	37540
Instagram	Kanal 5	32622	15

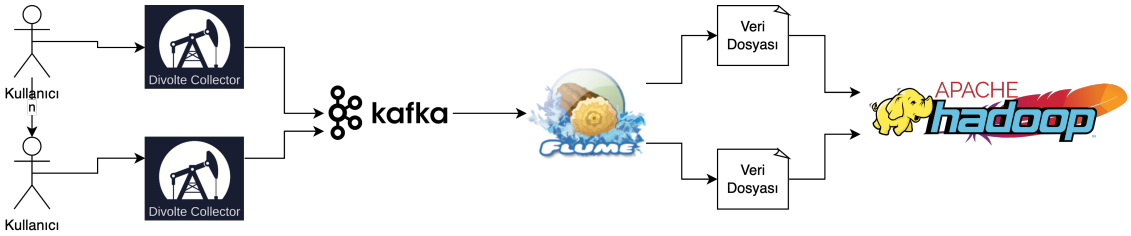
Bu tez çalışması kapsamında verinin kullanıma hazır hale getirilmesi için 3 adım uygulanmıştır. Birinci adım verinin toplanması ve depolanmasıdır. Bu adımlarda veri toplanması için gerekli alt yapı kurulmuş ve büyük veri ortamında saklanmıştır. İkinci adım verinin temizlenmesi aşamasıdır. Son adımda ise veri üzerinde özellik azaltımı işlemi uygulanmıştır.

3.2 Verinin Toplanması ve Saklanması

Verinin toplanması ve saklanması için sürekli veri akışı olan, bu veri akışı karşısında kayıpsız olarak veri saklayabilecek ve büyük veri ortamına uygun hale getirebilecek bir sistem tasarlanmıştır. Bu sistem dört temel aşamadan oluşmaktadır. Bunlar Divolte Collector[2] ile verinin toplanması, Kafka[3] ve Flume[4] ile büyük veri ortamına alınması, Hadoop[5] ile büyük veri depolama sisteminin kurulması ve Spark[6] ile verilerin modelin kullanımına hazır hale getirilmesidir.

Bu bölümde, tez için ilgili araçlar kullanılarak hazırlanan oluşturulan veri toplama mimarisi ve büyük veri ortamı anlatılmıştır. Şekil 3.1'de veri kümesi oluşturmak için tasarlanan sistem mimarisi gösterilmiştir.

[2]https://divolte.io/
[3]https://kafka.apache.org/
[4]https://flume.apache.org/
[5]https://hadoop.apache.org/
[6]https://spark.apache.org/



Şekil 3.1 Verinin toplanması için kurulan büyük veri mimarisi

3.2.1 Divolte Collector İle Veri Toplanması

Divolte Collector, kullanıcı tıklama verilerini Avro[7] dosya formatında saklamaya yarayan ve bu mesajları bir mesaj kuyruğuna yerleştirmeye yarayan açık kaynaklı JavaScript dilinde yazılmış bir araçtır. Kayıpsız olarak gerçek zamanlı verilerin aktarılmasını sağlar. Büyük veri ortamları için tasarlanmıştır. Yapısal veri formatı sayesinde yazma ve okuma işlemlerini çok hızlı bir hale getirir. Tamamen kullanıcı bilgisayarında çalışması sebebiyle ek kaynağa ihtiyaç duymadığı için asenkron veri toplamak için efektif olarak kullanılabilmektedir.

Verinin tutulması için kullanılan dosya formatı olan Avro dosya formatı, verilerin önceden belirlenen şemalara uygun olarak yapısal bir biçimde tutulması için 2009 yılında çıkarılmış açık kaynak kodlu bir dosya formatıdır. Büyük veri ortamlarında veri saklamak için tasarlanmıştır.

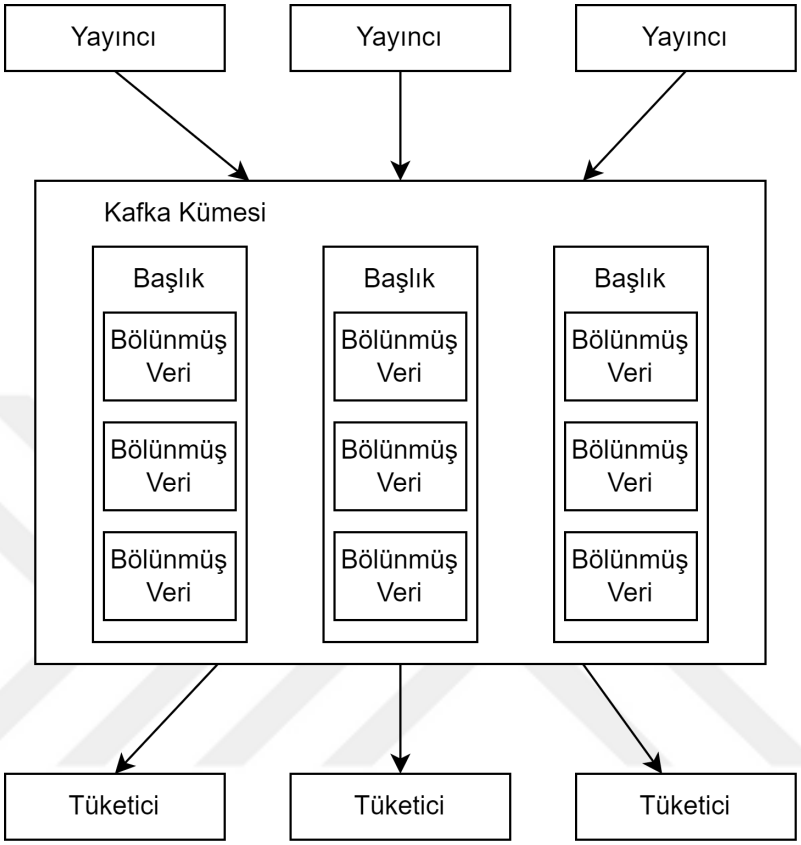
Bu tez kapsamında Divolte Collector aracı tüm kullanıcı bilgisayarlarından verinin kayıpsız ve gerçek zamanlı olarak toplanmasını ve toplanan verilerin avro dosya formatında saklanmasını sağlamak amacıyla kullanılmıştır. Saklanan dosyalar ise yine bu araç aracılığı ile Kafka ortamına aktarılmıştır.

3.2.2 Kafka İle Verilerin Depolanması

Kafka 2011 yılında çıkarılmış, olay akışı mantığında çalışan dağıtık mimariye sahip bir mesajlaşma sistemidir. Açık kaynak kodlu bir yazılımdır. Olay akışı mantığı çerçevesinde en az bir yayıncı(publisher) ve en az bir adet tüketiciye(consumer) sahiptir. Yayıncılardan alınan mesajlar bir başlık(topic) altında toplanır. Birbirinden farklı veriler farklı başlıklar altında depolanabilmektedir. Bu sayede aynı anda farklı iş ve durumlar için birbirinden bağımsız olarak üretilip tüketilebilmektedir. Tüketiciler ise ilgili başlıklara abone olarak mesajları dinler. Her bir başlık altında iletilmiş olan mesajlar bölünmüş bir biçimde(partition) tutulur. Kafka yazılımına dağıtıklığı sağlayan bu bölünmüş mimaridir. Her bir Kafka kümesi(cluster) içinde birden fazla bağımsız sunucu veya aynı sunucu içinde birbirinden farklı Kafka yazılımı olabilir.

[7]https://avro.apache.org/

Bu sayede veriler çok hızlı bir biçimde yazılıp okunabilmektedir. Bu yapının önemli yanlarından biri ihtiyaç duyulması halinde sisteme kolayca yeni bir sunucu veya yeni bir çalışan Kafka yazılımı daha eklenebilmesidir. Kafka mimarisi Şekil 3.2'de gösterilmiştir.



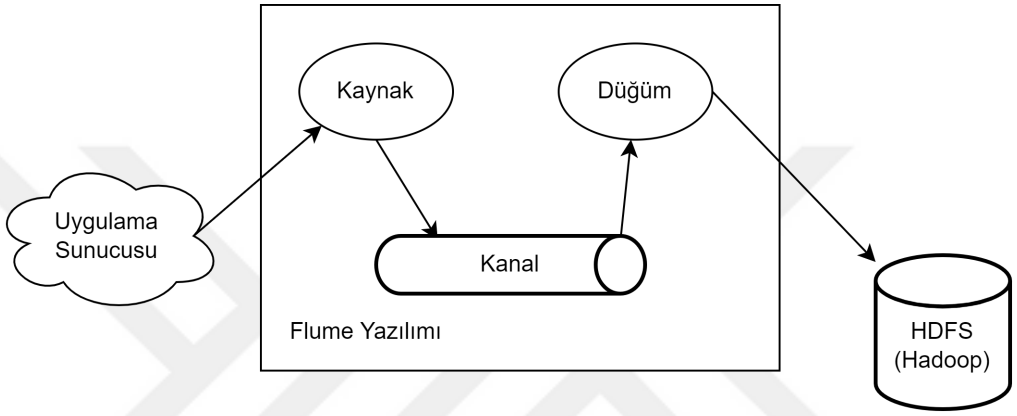
Şekil 3.2 Kafka mimarisi

Yayıncılar mesajlar Kafka ortamına iletildikten sonraki aşama ile ilgilenmez. Mesajın başarılı ulaşması yayıncı için yeterlidir. Kafka ortamına iletilen mesajları ise tüketiciler(consumers) alır ve sistemden siler. Mesajlar okunduktan sonra tüketici yazılımın hata alması Kafka mimarisinde önemli değildir. Yani senkron bir yapı değildir. Asenkron olduğu için hata durumunda mesajın tekrar sisteme eklenmesi gibi bir durum yoktur. Bu nedenle okuyan yazılım hata alması durumunda mesajı geri yazmalıdır. Kafka yazılımının en büyük dezavantajı bu yapısıdır.

Bu tez kapsamında Kafka mimarisi, verilerin depolanması ve hızlı ve dağıtık olarak aktarılması amacı ile kullanılmıştır. Divolte Collector yayıncı rolünde sisteme eklenmiştir. Üretilen Avro dosyaları Kafka aracılığı ile depolanmış ve buradan okunabilmesi için tüketicilerin kullanımına uygun duruma getirilmiştir.

3.2.3 Flume İle Verilerin Aktarılması

Flume 2012 yılında çıkarılmış, tutarlı ve dağıtık olarak verilerin taşınmasını sağlayan bir araçtır. Flume aracı ile veriler bir noktadan başka bir noktaya hızlı bir biçimde yazılabilmektedir. Bağımsız olarak çalışan her Flume yazılımı (Flume Agent), kaynaktan(source) veriyi okur. Okunan veri bir veya birden fazla kanala(channel) aktarılır. Tüketici(consumer) veriyi okuyana kadar geçici olarak kanallarda veri saklanır. Asıl hedefe ulaşması için ilgili kaynağın tüketicisinin veriyi okuması gerekmektedir. Veriyi okumak için belirlenen hedefler çıkış düğümü(sink) olarak adlandırılır. Şekil 3.3'de mimarisi gösterilmiştir.



Şekil 3.3 Flume mimarisi

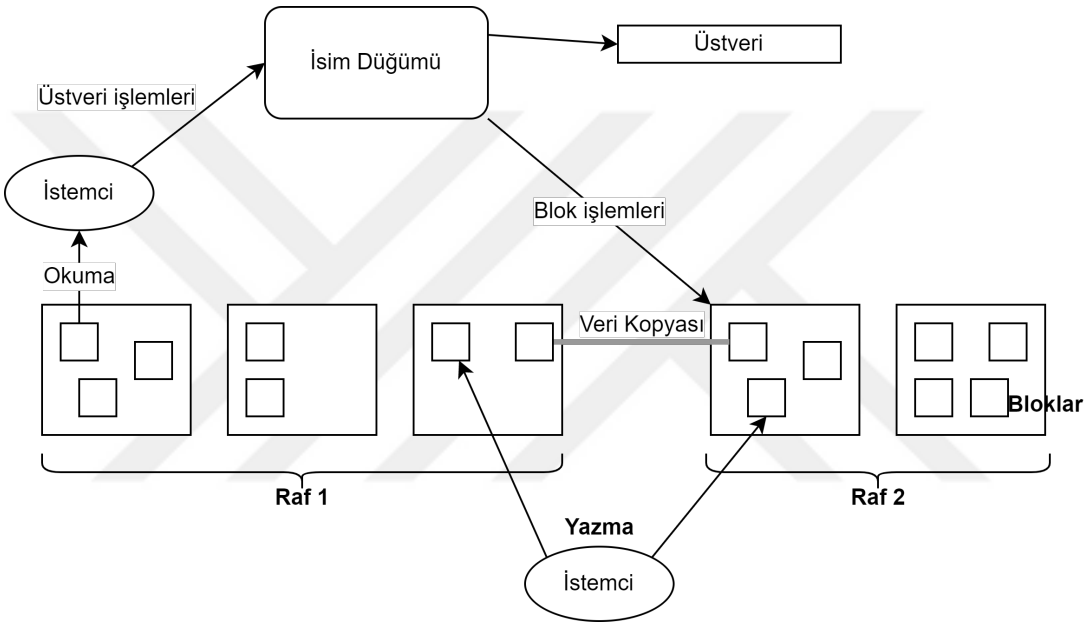
Bu tezde Flume aracı Kafka kaynağından alınan verilerin sabit ve önceden 65KB olarak belirlenmiş dosyaların büyük veri ortamına aktarılması için kullanılmıştır. Verilerin tutarlı ve sabit boyutlarda aktarılması okunması ve işlenmesi aşamasında çok önemli olduğu için Flume aracı genel sistem mimarisinde önemli bir noktada yer almaktadır.

3.2.4 Hadoop İle Verilerin Depolanması

Hadoop 2006 yılında yayınlanan, açık kaynak kodlu bir yazılım geliştirme çatısıdır. Büyük verileri işlemek için gerekli olan birçok yardımcı yazılımları ve kütüphaneleri bulundurmaktadır. Büyük veri kümeleri üzerinde basit programlama modelleri kullanılarak dağıtık mimariye sahip birden çok sunucu kümesi aracılığı ile hesaplamalar ve veri işleme yapılmasına olanak sağlamaktadır. Ölçeklenebilir ve dağıtık mimarisine rağmen verilerin tutarlı olarak saklanıp işlenebilmesine olanak sağlamaktadır. Bir çok alt modüle sahip olmasına rağmen asıl kilit noktaları HDFS(Hadoop Distributed File System - Hadoop Dağıtık Dosya Sistemi) [17] ve MapReduce(Eşle İndirge) mimarisidir.

HDFS hatalara karşı dayanıklı, ölçeklenebilir, kendi onarım mekanizmasına sahip dağıtık bir dosya sistemidir. Hadoop üzerinde depolanan tüm veriler HDFS dosya

formatında tutulmaktadır. HDFS mimarisinde dosyalara isim düğümleri(namenode) aracılığı ile erişilir. İsim düğümleri, istemcilerin verilere erişimini de denetlemektedir. Ek olarak yazılan verilerin bulunduğu blokları, boyutları ve gerekli tüm bilgilerin olduğu üstverileri de içermektedir. Veri düğümleri(datanodes) ise verilerin dağıtık biçimde tutulduğu alanlardır. Gelen okuma ve yazma işlemleri gerçekleştirmekle yükümlüdür. Veri düğümleri içinde veriler bloklar(blocks) halinde tutulur. Birden fazla veri düğümünün toplandığı alana ise raf(rack) adı verilmektedir. Hata toleransı sağlamak için veriler birden farklı veri düğümleri içinde kopyaları olacak biçimde saklanmaktadır. Burada veri düğümleri seçilirken farklı raflar içinde bulunan veri düğümleri seçilmektedir. HDFS mimarisi Şekil 3.4'de gösterilmiştir.



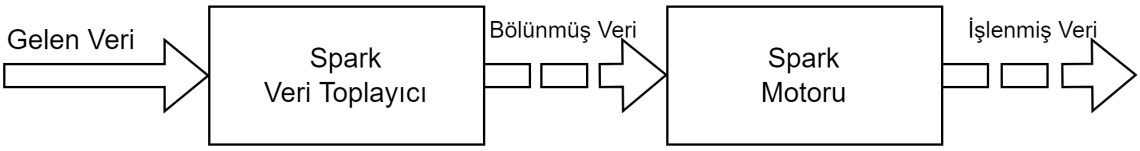
Şekil 3.4 HDFS mimarisi

3.3 Veri Ön İşlemleri

Model eğitimi öncesinde özellikler için bir takım ön işlemler uygulanmıştır. Bu ön işlemler giriş ve çıkış özelliklerine göre iki gruba ayrılabilir. Bu işlemleri hızlandırmak için Spark aracı kullanılmıştır.

3.3.1 Spark ile Veri Okunması ve Ön İşlemlerin Yapılması

Apache Spark, 2014 yılında geliştirilmiş ve büyük veri kümeleri üzerinde paralel olarak işlem yapılmasını sağlayan açık kaynak kodlu bir kütüphanedir. Bellek içi veri işleme özelliği sayesinde veri işlemeyi hızlı bir hale getirmiştir. Çalışma prensibi, gelen veriyi alıp iş birimlerine ayırdıktan sonra istenilen işlemleri yapıp çıktıyı elde etmek yönünedir. Basit çalışma mantığı Şekil 3.5'de gösterilmiştir.



Şekil 3.5 Spark mimarisi

Bu projede Spark hem girdilerin hem de çıktıların ön işlemlerden geçirilmesi için kullanılmıştır. Girdi olarak kullanılacak nümerik olan özellikler için normalizasyon işlemi uygulanmıştır. Kategorik alanlar için one-hot encoding yöntemi uygulanmıştır. One-hot encoding seçilmesinin sebebi bu yaklaşımın makine öğrenmesi metotlarında, diğer yöntemlere göre daha efektif [18] çalışmasıdır. Bu uygulama sonrası toplam özellik sayısı 65'e çıkmıştır. Kullanıcı IP bilgisi ise il, ilçe ve ülke bilgisine dönüştürülmüş ve bu bilgiler yeni birer özellik olacak biçimde veri kümesine eklenmiştir. Özellik seçimi ve özellik azaltımı işlemleri uygulanmıştır. Özellik azaltımı için uygulanan metotlar Bölüm 3.4'te anlatılmıştır.

3.4 Kullanıcı Tekilleştirme

Kullanıcı tekilleştirme, farklı zamanlarda ve farklı cihazlardan gelen aynı kullanıcının bilgilerinin birleştirilip tek bir kullanıcı gibi değerlendirilmesidir.

Reklam kanalının gerçek etkisinin hesaplanabilmesi için önemli bir adımdır. Mevcut analizler kullanıcı tekilleştirme işlemi yapılamadan yanıltıcı sonuçlar verebilmektedir. Bunun sebebi ise, farklı kaynaklardan gelen her kullanıcı için, kullanıcı satın alma seyahatinin tam olarak saptanamamasıdır. Kullanıcı satın alma seyahati ile kullanıcının satın alma adımına gelene kadar tıkladığı veya gördüğü tüm reklam kanalları, eğer reklam kanalları ile etkileşime geçti ise bu etkileşimin saptanmasıdır. Saptamanın eksik olduğu her noktada elde edilen sonuç da yanlış olacaktır.

Bu tez çalışmasında bir diğer amaç ise kullanıcı satın alım yolculuğunda kullanıcı tekilleştirme işleminin yapılabilmesidir. Mevcut araçlar ya sınırlı veri tutmakta ya da kullanıcı tekilleştirmeye imkân vermemektedir. Bu nedenle veri toplama ve ön işlemler sonrasında ek olarak kullanıcı tekilleştirme işlemi de yapılmıştır.

Kullanıcı tekilleştirme için sistem üzerinde oturum açan kullanıcı için bir kimlik kodu üretilir. Bu sayede kullanıcı farklı cihazlardan da gelse tek bir kullanıcı olarak değerlendirilir. Ayrıca makine id özelliği sayesinde de farklı cihazlar olsa dahi kullanıcılar bu cihazla eşleştirildiği an bu cihazlardan gelen tüm etkileşimler eşleştirme yapılan kullanıcı altında değerlendirilmiştir.

3.5 Özellik Azaltımı

Veri kümesinde bulunan örneklerde 65 adet özellik bulunmaktadır. Bu özelliklerin azaltılması için PCA(Principal Component Analysis- Ana bileşen analizi) ve otokodlayıcı(Autoencoder) yöntemleri uygulanarak başarıları değerlendirilmiştir. Elde edilen sonuçlar 5. bölümde anlatılmıştır.

3.5.1 PCA ile Özellik Azaltma

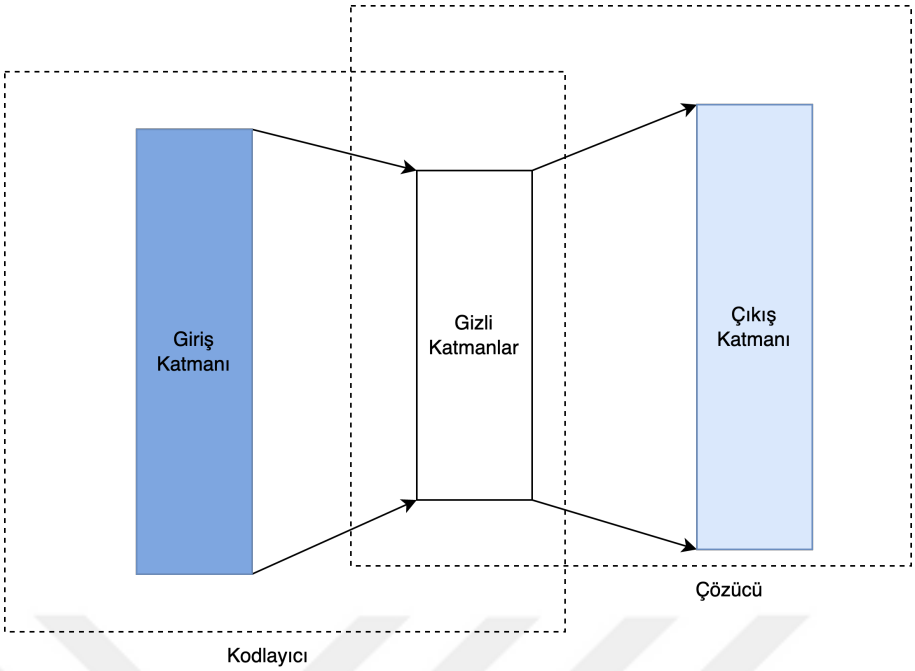
PCA [19] birbirleri ile ilişkili olan çok sayıda değişkenden meydana gelen birçok değişkenli sistemi, bu değişkenlerin doğrusal fonksiyonları şeklinde daha az sayıda ve birbirleri ile ilişkisiz olamsıyla beraber aynı zamanda önceki sisteme ait toplam değişimi mümkün olduğunca büyük oranda açıklayabilen yeni değişkenlerden meydana gelen sisteme dönüştüren çok değişkenli istatistiksel analiz tekniğidir. Bu yöntemle m«n olmak üzere, n özellik farklı m adet özelliğe dönüştürülür.

3.5.2 Otokodlayıcı ile Özellik Azaltma

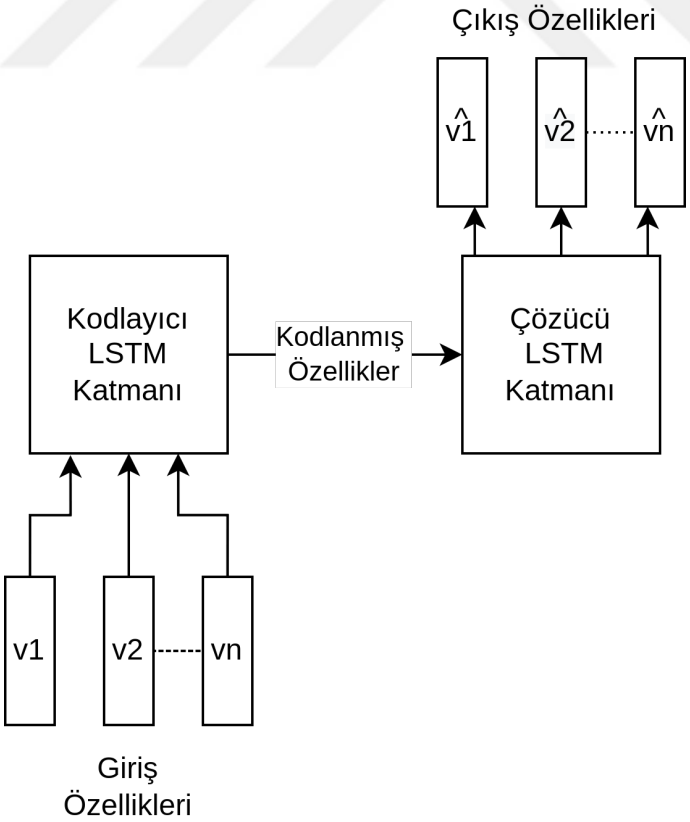
Otokodlayıcı, Rumelhart ve arkadaşlarının [20] tasarladığı derin öğrenme temelli bir yapay sinir ağı modelidir. Bu modelin temel amacı verinin sıkıştırmasıdır. Model ile veri önce sıkıştırılır ve daha sonra kayıpsız olarak geri açılabilir. Veri kümesini sıkıştırmak için yapılan işleme kodlama işlemi denir. Kodlama işlemi ile gizli katmana verinin nasıl aktarılacağı belirlenir. Çözme işlemi ile ise sıkıştırılan veri eski veriyi temsil edecek hale dönüştürülür. Dönüştürülen veri ya çok az ya da sıfır kayıp ile veri tekrar elde edilmektedir. Eğitmensiz bir yapay sinir ağıdır.

Bu tez çalışmasında özellik azaltım modeli olarak iki farklı otokodlayıcı kullanılmıştır. Kullanılan ilk otokodlayıcı modeli, Rumelhart ve arkadaşlarının [20] tasarladığı klasik otokodlayıcı modelidir. Model mimarisi Şekil 3.6'de gösterilmiştir.

İkinci model, Nitish K. ve arkadaşlarının [21] yaptığı çalışmada önerdikleri zaman serisi veriler üzerinde kullanılan LSTM mimarisine sahip bir otokodlayıcı modelidir. Bu model, özellik azaltımından daha çok verilen örneklerdeki özellikleri temsil edecek yeni bir özellik seti üretmek için tasarlanmıştır. Otokodlayıcının kodlayıcı katmanındaki LSTM modeli giriş değerlerini sırayla okur. Son değer okunduktan sonra, çözücü LSTM katmanına kodlayıcı katmanından elde edilen değerler aktarılır. Çözücü LSTM katmanı ise okunan değerler çerçevesinde çıkış üretir. Bu çıkış yeniden düzenlenmiş özellik setini ifade etmektedir. Yapılan çalışmalar sonucunda zaman serisine dayalı veriler üzerinde bu modeli kullanan sistemlerin başarı oranlarının arttığı gözlemlenmiştir. Model mimarisi Şekil 3.7'de gösterilmiştir. Şekilde $v1$, $v2$ ve vn giriş özelliklerini, $\hat{v1}$, $\hat{v2}$ ve $\hat{vn}$ çıkış özelliklerini temsil etmektedir.



Şekil 3.6 Klasik otokodlayıcı mimarisi



Şekil 3.7 LSTM otokodlayıcı mimarisi

3.6 Verinin Etiketlenmesi

Veri kullanıma hazır hale geldikten sonra sınıflandırma işleminde kullanılabilmesi için verinin etiketlenmesi gerekir. Veri büyüklüğü nedeniyle her kullanıcı için tam olarak reklam kanal etkisi hesaplanamamıştır. Bu nedenle etiketleme yapılırken her bir kullanıcı oturumu için, siteye bir reklam kanalı aracılığı ile gelindiği durumda, o reklam kanalını temsil eden çıkış sınıfı 1 olarak ve diğer çıkış sınıfları 0 olarak etiketlenmiştir. Sadece satın alınan durumların değerlendirilebilmesi için, satın alım yapan kullanıcılar için bu etiketleme yapılmaktadır. Satın alım yapmayan kullanıcılar için çıkış etiketleri 0 olarak belirlenmektedir.

3.7 Reklam Kanal Etkisinin Mevcut Yöntemler ile Hesaplanması

Mevcut durumda reklam kanalı etkisinin hesaplanması için iki yöntem izlenmektedir. Birinci yöntem basit olarak son ve ilk tıklama oranlarının hesaplanmasıdır. Diğer yöntem ise detaylı uzman analizidir.

3.7.1 İlk ve Son Tıklama İle Kanal Etkisinin Hesaplanması

En basit reklam etkisi hesaplama yöntemi olarak kullanılır. Hesaplama mantığı dönüşüm öncesi ilk tıklanan reklam kanalı ve son tıklanan reklam kanallarının yüzdesel dağılımını bulma yönündedir. Dağılımın hesaplanması için satın alım öncesi ve sonrası için toplam sayı bulunur. Bulunan sayı üzerine her kanal için toplam içindeki yüzdesel değeri hesaplanır. Bu dağılım bulunduktan sonra toplam etki hesaplanmaktadır.

3.7.2 Detaylı Uzman Analizi İle Kanal Etkisinin Hesaplanması

Sonuçlar verilirken karşılaştırma yapılabilmesi için mevcut reklamların etkisinin hesaplanması gerekmiştir. Bunun için her reklam kaynağının analiz aracı kullanılmaktadır.

Bir reklam kanalının etkisinin hesaplanması için öncelikle tıklama(click) ve gösterim(impression) sayıları alınmaktadır. Burada tıklama ile ilgili reklama kullanıcının tıklaması sonucu alınan etkileşimler temsil edilmektedir. Her tıklama sonrası bu değer bir artmaktadır. Gösterim değeri ise kullanıcı reklama tıklamasa bile o reklamı görmesini temsil etmektedir. Her reklam gösteriminde bu değer bir artmaktadır. Bu iki değerin ortak değerlendirilmesi ile önemli bir gösterge olan CTR değeri elde edilmektedir. CTR(Click-through rate- Tıklanma oranı) hesaplaması için kullanılan formül eşitlik 3.1'de gösterilmiştir. CTR değeri iiçn toplam tıklamanın

toplam etkileşime oranı temsil edilmektedir. Eşitlikteki n değeri, reklam kanal etkisinin hesaplanacağı süreyi ifade etmektedir. CTR değeri ne kadar yüksek ise reklam o kadar başarılı demektir.

$$CTR = \frac{\sum_{i=1}^{n} click}{\sum_{i=1}^{n} impression} \tag{3.1}$$

Reklamın tıklama performansını hesapladıktan sonra parasal karşılığının da hesaplanması gerekmektedir. Bunun hesaplanması için ilgili bütçe(cost) ve bu bütçe karşılığı reklama tıklanarak ne kadarlık ürün satıldığı(conversion) değerleri kullanılmaktadır. Reklam kanalının etkisi hesaplanırken CPC(Cost per click- Tıklama başı maliyet), CPA(Cost per action- Aksiyon başı maliyet) ve CR(Conversion rate- Dönüşüm oranı) değerleri de hesaplanmaktadır. Değerler hesaplanırken kullanılan formüller; CPC için eşitlik 3.2, CPA için eşitlik 3.3 ve CR için eşitlik 3.4'de gösterilmiştir. Her eşitlik için n değeri, reklam kanal etkisinin hesaplanacağı süreyi ifade etmektedir. CPC tıklama başına maliyeti, CPA dönüşüm başına maliyeti ve CR değeri de tıklama başına dönüşümü temsil etmektedir. Bu değerler ne kadar yüksek ise reklam o kadar başarılı demektir.

$$CPC = \frac{Cost}{\sum_{i=1}^{n} click} \tag{3.2}$$

$$CPA = \frac{Cost}{\sum_{i=1}^{n} conversion} \tag{3.3}$$

$$CR = \frac{\sum_{i=1}^{n} conversion}{\sum_{i=1}^{n} click} \tag{3.4}$$

Son olarak Bölüm 3.7.1'de anlatılan, satın alım yapan kullanıcıların ilk ve son tıklama değerleri hesaplanmaktadır.

CPC, CPA, CR, CTR ve ilk-son tıklama değerleri uzmanlar tarafından değerlendirilmekte, bu değerlendirme sonucu her reklam kanalı için dönüşüm etki analizi çıkarılmaktadır. Reklam kanallarının etki analizi bulunduktan sonra bu etki analizine göre bütçe dağılımı yüzdesel olarak yapılmaktadır. Bu analiz için kullanılan somut bir yöntem bulunmamaktadır. Her müşteri için ayrı değerlendirmeler yapılmakta, elde edilen değerler aynı olsa dahi müşteri bazında farklı sonuçlar çıkmaktadır. Müşterinin mevcut durumu da değerlendirildiği için bu farklılıklar

yaşanmaktadır. Bu süreç uzman bilgi birikimine göre değişebilmektedir.

3.7.3 Mevcut Yaklaşımların Neden Olduğu Sorunlar

Bir reklam kanalının dönüşüm değeri, eğer satın alım öncesi son gelinen reklam kanalı ise güncellenmektedir. Bu nedenle, önceki etkileşim olan reklam kanallarının dönüşüm değeri değişmeyeceği için ilgili reklam kanallarının gerçek etkisinin hesaplanmasında hataya sebep olmaktadır. Bu tür durumlar ise tüm verinin güvenilirliğini etkilemektedir.

İlk ve son tıklama verileri tüm reklam kanallarını ifade etmemektedir. Başka reklam kanallarıyla kullanıcı etkileşime geçse dahi satın alım yapması sonucu dönüşüme etkisi olup olmadığı veya bu etkinin ne kadar olduğu hesaplanamamaktadır.

Diğer bir sorun ise mevcutta kullanılan analiz araçları tüm kullanıcı hareketlerini saklamamaktadır ve bu da önemli bir sorun ifade etmektedir. Veri eksikliğinden kaynaklı yanlış sonuçlar çıkabilmektedir.

Son olarak ise tüm bu hesaplamaların ve analizlerin her bir reklam kanalı için yapılması gerekmektedir. Bu işlem ise her bir reklam kanalı için 1 haftalık süre gerektirmektedir. Veri eksikliği olduğu durumlarda ise bu süreç daha da uzamaktadır. Bu durum da yanlış bütçe dağıtımı sonucu oluşabilecek kayıplar için hızlı önlem alınmasını zorlaştırmaktadır. Sabit bir formül olmaması da farklı sonuçlar çıkabilmesine neden olmaktadır.

4
METODOLOJİ

Bu bölümde tez çalışmasında kullanılabilecek alternatif derin öğrenme modelleri, ne için kullanıldıkları ve hangi problemlerde başarılı oldukları ve tez kapmasında seçilen model anlatılmıştır.

4.1 Problem Tanımı

Çevrimiçi reklamcılık için kullanıcının e-ticaret sitesine gelmesini sağlayan her bir reklam kaynağı kullanıcı hedefleme için önemlidir. Buradaki ana problem, kullanıcıların bu kaynaklar ile farklı zamanda hatta farklı cihazlar ile etkileşime geçebilmesidir. Bu durum ise kullanıcının siteyle etkileşime geçtiği kaynakların doğru saptanmasını zorlaştırmaktadır. Karşılaşılan sorunu çözmek için kullanılan yöntem el yordamı ile analizler yapmaya ve bu analizler sonucu karar vermeye dayanmaktadır. Bunun için genellikle alışveriş için ilk giriş veya son giriş kaynakları değerlendirmeye alınmaktadır. Bu durum ise ara kaynakların etkisini göz ardı etmekte ve doğru sonuçların alınmasını engellemektedir. Ayrıca analiz için veri boyutundan kaynaklı sadece sınırlı bir veri kümesi kullanılabilmektedir. Bu da veri doğruluğunu etkilemekte, sonuçların tutarsızlığına neden olmaktadır.

Bu sorunlardan yola çıkarak hem büyük veriyi hızlı işleyebilecek hem de geçmiş ve gelecek durum arasında zaman serisine dayalı ilişki kurabilecek bir model ihtiyacı doğmuştur. Bunun için ise mevcut yaklaşımlar arasında bu sorunu en iyi çözebilecek LSTM mimarisinin kullanılmasına karar verilmiştir. Bu model sayesinde hem kullanıcının tüm hareketleri analize eklenebilecek hem de büyük veri üzerinde daha etkili analizler yapılabilecektir.

4.2 Model Tasarımı Öncesi Hazırlıklar

Model tasarımı öncesinde elimizde bulunan verinin geniş bir analizi yapılmıştır. Bunun için hem derin öğrenme tabanlı metotlar hem de klasik makine öğrenmesi

metotları değerlendirilmiştir. Literatürde bu konu ile ilgili daha önce yapılan karşılaştırmalı test sonuçları veren Ren ve arkadaşlarının [14] yayınladığı sonuçlar incelenmiştir. İncelemeler sonucu eğitmenli öğrenim modellerinin daha etkili olacağı anlaşılmıştır. Çıktı sonuçları Tablo 4.1'de gösterilmiştir. LR(Logistic regression- Lojistik regresyon) [3], SP(Simple probabilistic- Basit olasılıksal) [5], AH(Additive hazard- Toplamsal tehlike) [9], AMTA(Additional multi-touch attribution- İlave çoklu-etkileşim ilişkilendirme), ARNN (Attention recurrent neural network- Dikkat mekanizmalı yinelenen tekrarlı ağ) ve DARNN(Dual-attention recurrent neural network- Çift katmanlı dikkat mekanizmalı yinelenen tekrarlı ağ) [14] için çıktılar gösterilmiştir. AH ve SP metotları istatistiksel analizler sunmaktadır. LR ve AMTA modelleri ise klasik makine öğrenmesi ve istatistiksel araçları bir arada kullanmaktadır. ARNN ve DARNN ise derin öğrenme metodu kullanmaktadır. Bu sonuçlar elde edilirken birden fazla veri kümesi kullanılması performans güvenilirliğini arttırmıştır.

Tablo 4.1 Karşılaştırmalı model çıktıları

	Miaozhen Veri Kümesi		Criteo Veri Kümesi	
Modeller	AUC	Kayıp	AUC	Kayıp
LR	0.8418	0.3496	0.9286	0.3981
SP	0.7739	0.5617	0.6718	0.5535
AH	0.8693	0.6791	0.6791	0.5067
AMTA	0.8357	0.1636	0.8465	0.3817
ARNN	0.8914	0.1610	0.9793	0.1850
DARNN	**0.9123**	**0.1095**	**0.9799**	**0.1591**

Şekil 4.1'den yola çıkarak derin öğrenme temelli bir metot kullanılmasına karar verilmiştir. Toplanan verinin eğitmenli makine öğrenmesi mimarilerine uygun olduğu gözlemlenmiştir. Kullanıcının son satın alım bilgileri sınıf bilgisi olarak belirlenmiştir. Diğer kanallarının satın almaya etkisi tam olarak bilinememektedir. Elimizdeki etiketli veriler sadece son alınan satın alma kaynağı bilgisini göstermektedir. Bu nedenle diğer kaynakların da etkisini saptayabilmek için eğitmenli öğrenmeye uygun bir model geliştirilmesi uygun görülmüştür.

Derin öğrenme metodunun belirlenebilmesi için öncelikle verinin yapısı incelenmiştir. Toplanan verilerin uzun bir zaman aralığına yayıldığı ve birbirleri arasındaki ilişkilerin belirlenmesi gerektiği bilinmektedir. Bu problemin çözümü için zamana dayalı analizlerin önemli olduğu, bu analizlere uygun bir yapının geliştirilmesi gerektiğine karar verilmiştir. Bölüm 2'de incelenen çalışmalar sonucunda zaman serisine dayalı analizin problemin çözümünde ihtiyaç olunan yöntem olduğu gözlemlenmiştir. Zaman serisine dayalı analizler için ise RNN modellerinin daha başarılı olduğu literatür çalışmalarında [14][13][16] gösterilmiştir. Bu nedenle RNN tabanlı bir model ile devam edilmesi uygun görülmüştür. RNN mimarisinin geliştirilmiş ve daha

uzun süreli zaman serisi analizlere uygun hali olan LSTM modeli çalışma modeli olarak belirlenmiştir. Model detayları Bölüm 4.2.1'de anlatılmıştır.

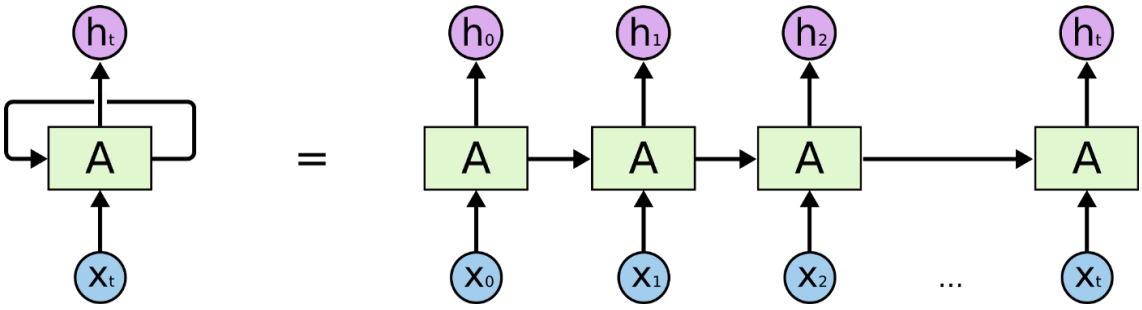
Bir kullanıcı farklı kanallardan farklı kimlik bilgileri ile gelebilmektedir. Bunun için her kullanıcıyı tekilleştirecek şekilde veri kümesi gruplanmış ve her bir kullanıcının veri kümesi içindeki hareketlerinin tamamının temsil edilmesi sağlanmıştır. Kullanıcı grupları oluşturmak için, oturum ve kimlik bilgileri kullanılmıştır. Oluşturulan bu gruplar ile modelin her bir öğrenme iterasyonu için tüm gruplanmış kullanıcı verilerinin işlenmesi sağlanmıştır. 65 özellik ve kullanıcı gruplama ile $\{\{g_l\}_{l=1}^{64}, u_k, y_k\}_{k=1}^{270000}$ parametreleri elde edilmiştir. Burada 270 bin adet kullanıcı grubu ve her bir kullanıcı u için 65 adet g (giriş özellikleri) özellikleri bulunmaktadır. Her kullanıcı grubu için 5 adet y sınıfı elde edilmektedir ve bu sınıflar reklam kanallarını temsil etmektedir.

4.3 Derin Öğrenme Modeli Seçimi

Derin öğrenme modeli için RNN yapısına sahip bir model kullanılmıştır. Bunun için klasik RNN, LSTM ve BiLSTM modellerinin mimarileri incelenmiştir.

4.3.1 RNN Mimarisi

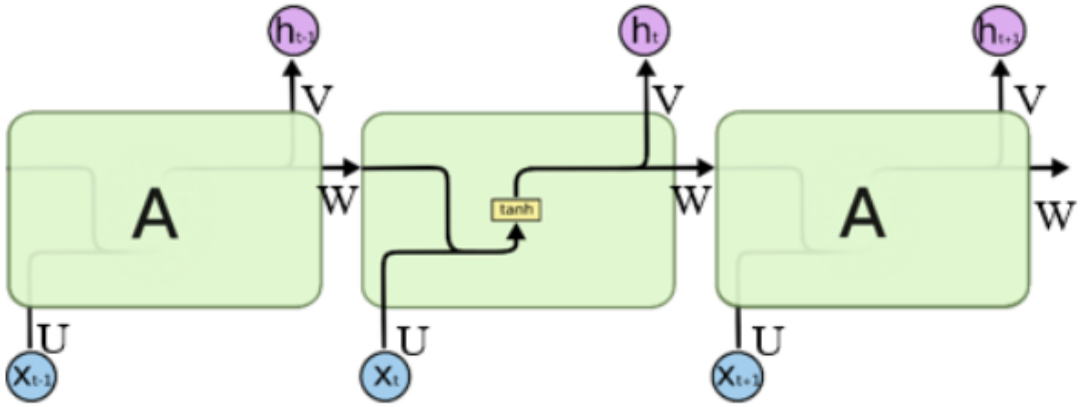
RNN [22] öğrenme modeli 1986 yılında yayınlanmış bir derin öğrenme modelidir. Klasik yapay sinir ağlarında girdi değerleri ileri yönlü çalışır. RNN modelinde ise her nöron çıkış değerini hesaplamak için hem kendi giriş değerini hem de kendisinden önceki nöronun çıkış değerini kullanır. Önceki nörondan gelen bilginin sonraki nörona taşınabilmesini sağlayan bu yapı sayesinde bir hafıza yapısı kurulur. Örnek mimari Şekil 4.1'de gösterilmiştir.



Şekil 4.1 RNN (Yinelemi Sinir Ağı) mimarisi [23]

Bu ağın en önemli noktası hafıza modelidir. Hafıza yapısı bu modeli zaman serisi analizlerine uygun hale getirmektedir. Hafıza modelinin çalışması için bir önceki nöronun çıktı değeri ve mevcut nöronun girdi değeri alınır, bu değerler bir

fonksiyondan geçirilerek sonraki nörona gönderilecek değer elde edilir. Eşitlik 4.1 ile hesaplama bağıntısı gösterilmiştir. W_{hh} bir önceki nöronun ağırlığı, h_{t-1} bir önceki nöronun çıktı değeri, W_{xh} mevcut nöronun ağırlığı ve x_t mevcut nöronun girdi değeridir. Bu değerlerin sonucu h_t ise sonraki nöronun girdi için kullanacağı hafıza yani çıkış değeridir. Bu sistem Şekil 4.2 ile gösterilmiştir. Modelin doğru çalışabilmesi için verinin sıralı olması gerekmektedir. Verinin sırası bozulursa sistemin de başarı oranı düşecektir.

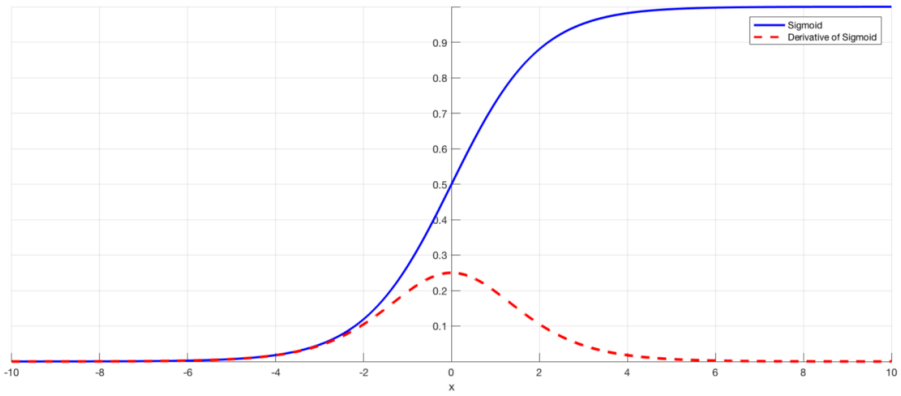


Şekil 4.2 RNN nöron mimarisi [23]

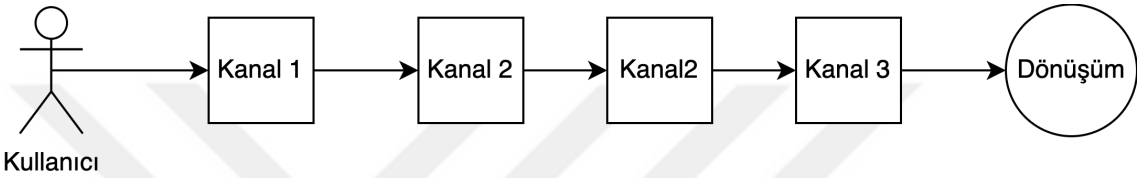
$$h_t = tanh(W_{hh}h_{t-1} + W_{xh}x_t) \tag{4.1}$$

Temel RNN modeli sadece bir önceki nöronun değerini hafızasında tutabilmektedir. Kullanılan sigmoid fonksiyonu, geçmişe yönelik bilgilerin tutulmasını zorlaştırmaktadır. Bu fonksiyon [0,1] veya [-1,1] aralığına dağılmaktadır. Değer aralığı dar bir aralık olduğu için değişim etkisi de dar bir aralıkla sınırlı kalacaktır. Değer aralığının dar bir aralıkta kalması nedeniyle türevi de küçük olacaktır. Türevin daha da küçük olması zaten azalmış olan değişim etkisinin daha da etkisiz hale getirmektedir. Bu durum Şekil 4.3 ile gösterilmiştir. Bu nedenlerden dolayı uzun aralık veya süre içeren veri kümesi üzerinde yapılacak analizlere uygun bir model değildir.

Bu tez çalışmasında kullanılan veri kümesinde bulunan kullanıcıların reklam etkileşimleri ve satın almaya giden yolculuğu çok uzun zamanlara yayılabilmektedir. Şekil 4.4'de bu duruma bir örnek gösterilmiştir. Bu sorunu aşmak için yine RNN mimarisini temel alan ve uzun süreli verileri saklayabilen LSTM mimarisi tercih edilmiştir.



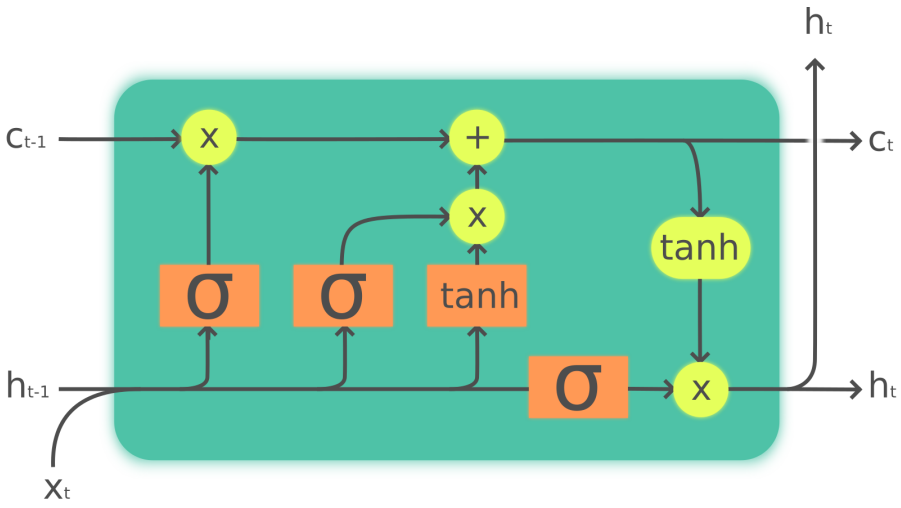
Şekil 4.3 Sigmoid fonksiyon ve türevi



Şekil 4.4 Tekil bir kullanıcı için örnek satın alma yolculuğu

4.3.2 LSTM Mimarisi

LSTM [24] mimarisi, RNN mimarisi üzerine geliştirilmiş bir modeldir. Bu mimarinin en önemli noktası sadece bir önceki durumu değil, daha önceki durumları da önemine göre değerlendirebilmesidir. Bunu yapabilmek için RNN mimarisinin en büyük dezavantajı olan büyük değişikliklerin etkisinin küçük olmasına neden olan soruna çözüm üretebilmiştir. Bunun için kapı adı verilen, neyin hatırlanıp neyin unutulacağına karar veren 3 adet yapı geliştirilmiştir. Şekil 4.5'de LSTM mimari yapısı gösterilmiştir.



Şekil 4.5 LSTM hücre mimarisi [25]

Unut kapısı (Forget Gate), bir önceki nöronun çıkış değeri ve nöronun giriş değerinden hangisinin daha değerli olduğunu belirlemektedir. Eşitlik 4.2'de çalışma bağıntısı gösterilmiştir. x_t mevcut hücrenin girdi değerini, u_f girdinin ağırlık değerini, h_{t-1} önceki gizli hücrenin değerini ve w_f ise o gizli hücreye ait nöronun ağırlığını temsil etmektedir. Çıkan sonuç 0'a ne kadar yakınsa eski bilgilerin o kadar değersiz olduğu anlamına gelmektedir.

$$f_t = \sigma(x_t * u_f * h_{t-1} * w_f) \tag{4.2}$$

Giriş kapısı (Input Gate), nöronun girdi değeri olan x_t değerinin taşıdığı bilginin ne kadar önemli olduğunu hesaplamaktadır. Eşitlik 4.3'de çalışma bağıntısı gösterilmiştir. i_t mevcut hücrenin girdi değerini, u_i girdinin ağırlık değerini, h_{t-1} önceki gizli hücrenin değerini ve w_i ise o gizli hücreye ait nöronun ağırlığını temsil etmektedir. Çıktısının hesaplanması için sigmoid fonksiyon kullanılmaktadır. Çıktı hesaplandıktan sonra ağı düzenlemek için tanh fonksiyonu ile yeni bilgi hesaplanmaktadır. Eşitlik 4.4'de bağıntı gösterilmiştir. Buradan çıkan sonuç ile hücrenin durumu güncellenmektedir. Hücre durum bilgisi, o hücrenin taşıdığı bilgiyi temsil etmektedir. Bir sonraki nörona taşınacak bilginin ne kadar değerli olduğunu temsil etmektedir. Eşitlik 4.5'de bağıntı gösterilmiştir.

$$i_t = \sigma(x_t * u_i * h_{t-1} * w_i) \tag{4.3}$$

$$n_t = tanh(x_t * u_c * h_{t-1} * w_c) \tag{4.4}$$

$$c_t = f_t * c_{t-1} * i_t * n_t \tag{4.5}$$

Çıkış kapısı (Output Gate) gizli hücrenin çıktı değerinin hesaplandığı kapıdır. Eşitlik 4.6'de bağıntı gösterilmiştir. x_t mevcut hücrenin girdi değerini, u_o girdinin ağırlık değerini, h_{t-1} önceki gizli hücrenin değerini ve w_o ise o gizli hücreye ait nöronun ağırlığını temsil etmektedir. Kapının çıktı değerinin hesaplanması için sigmoid fonksiyon kullanılmaktadır. Hücrenin çıktı değerini hesaplamak için eşitlik 4.7 kullanılır.

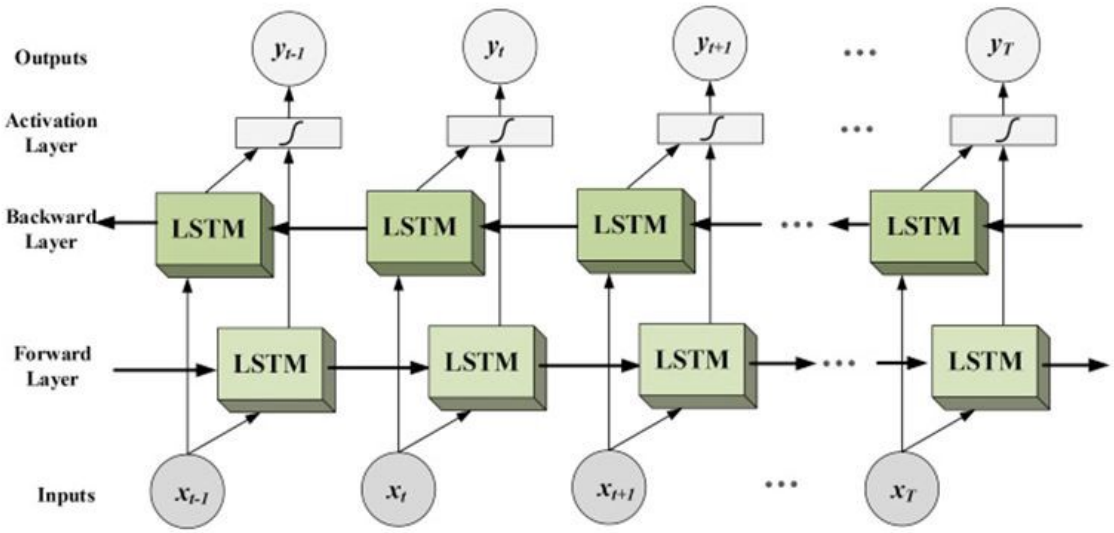
$$o_t = \sigma(x_t * u_o * h_{t-1} * w_o) \tag{4.6}$$

$$h_t = o_t * tanh(c_t) \tag{4.7}$$

LSTM modeli bu tez kapmasında yapılacak analize uygun bir modeldir. Hem uzun süreli bilgileri değerlendirebilmekte hem de zaman serisi verilere uygun bir yapı sunmaktadır.

4.3.3 BiLSTM Mimarisi

BiLSTM [26] 1997 yılında geliştirilmiş, LSTM mimarisini temel alan bir derin öğrenme modelidir. Çalışma mantığı LSTM mimarisine çok yakındır. İki adet birbirinden bağımsız LSTM katmanına sahiptir. İlk LSTM girdiyi ileriye doğru beslerken, diğer LSTM girdiyi geriye doğru besler. Yani biri geçmişten geleceğe doğru veri alır, diğeri ise gelecekten geçmiş doğru veriyi alır. Sistem mimarisi Şekil 4.6'de gösterilmiştir.



Şekil 4.6 BiLSTM mimarisi[25]

İki modelden gelen çıktı değerinin dönüştürülmesi için sigmoid fonksiyon kullanılmıştır. Eşitlik 4.8'de bağıntı verilmiştir. y_{fl} ileri yönelik besleme yapan LSTM modelinin çıktısını, y_{bl} geriye yönelik besleme yapan LSTM modelinin çıktısını, y_t ise BiLSTM model çıktısını göstermektedir.

$$y_t = \sigma(y_{fl}, y_{bl}) \tag{4.8}$$

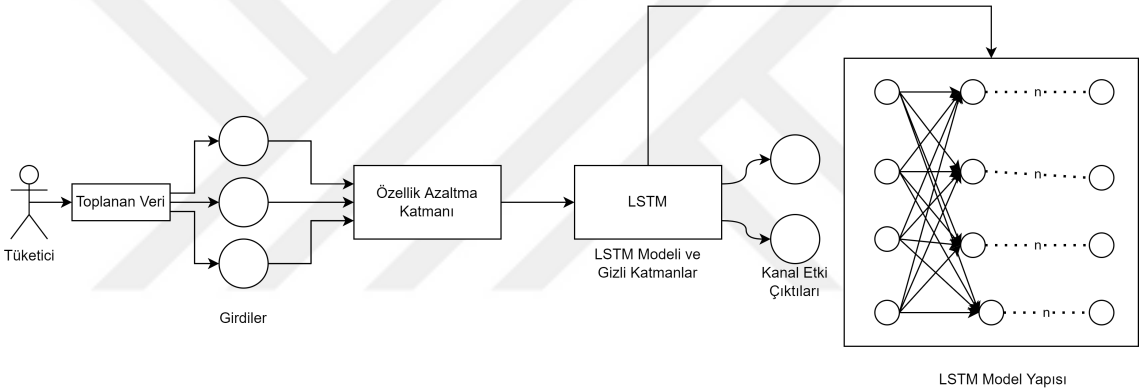
BiLSTM modeli bir verinin önünde ve arkasındaki veriler doğrudan ilişkisi varsa etkili sonuçlar verebilmektedir. Buna en iyi örnek olarak cümleden anlam çıkarma

verilebilir. Bir cümle içindeki kelimenin hem önündeki hem de ardındaki kelime doğrudan o kelimenin anlamını etkilemektedir.

Bu tez kapsamında yapılan çalışmada BiLSTM modeli ile denemeler yapılmış ve sonuçları etkilemediği gözlemlenmiştir. Elde edilen sonuçlar bölüm 5'te gösterilmiştir.

4.4 Sistem Model Mimarisi

Bu tez çalışmasında derin öğrenme modeli olarak LSTM mimarisi seçilmiştir. LSTM mimarisinin seçilmesinin sebebi geçmişe yönelik daha uzun belleğe sahip olmasıdır. Geliştirilen model kullanıcı girdilerini alıp bu girdileri özellik azaltma katmanına gönderecek, özellik azaltma katmanından elde edilen değerleri ise LSTM katmanına girdi olarak aktaracak biçimde tasarlanmıştır. Kullanılan model Şekil 4.7'de gösterilmiştir.

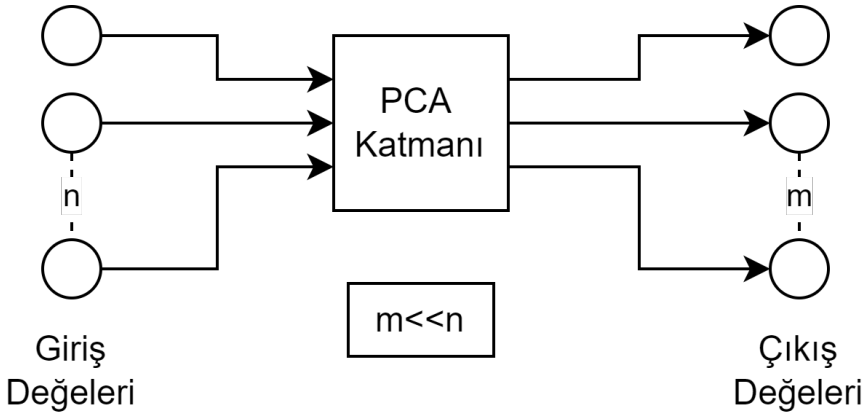


Şekil 4.7 Kanalların satın almaya etkisinin tespiti için önerilen model

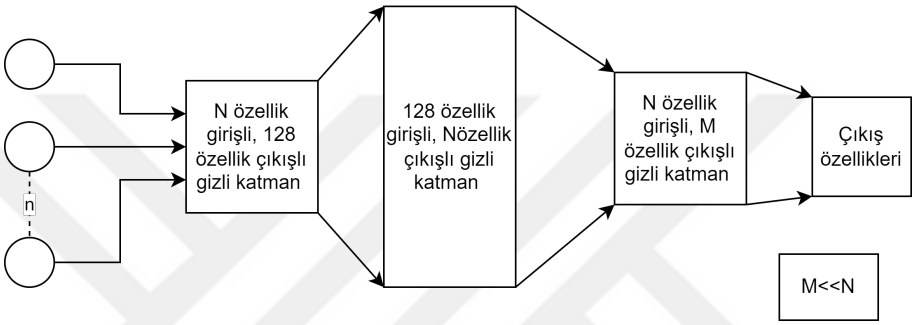
Özellik azaltma için 3 farklı model kullanılmıştır. Bu modeller PCA, klasik otokodlayıcı ve LSTM otokodlayıcı modelleridir. PCA için kullanılan model mimarisi Şekil 4.9'de ve otokodlayıcılar için kullanılan model mimarileri de Şekil 4.8'de gösterilmiştir. Farklı çıkış değerleri için alınan sonuçlar Bölüm 5'de değerlendirilmiştir.

Çıktı birimi her reklam kanalını temsil edecek ve onun etkisini gösterecek biçimde belirlenmiştir. Geliştirilen bu model sayesinde tüm reklam kanallarının etkisi tek bir model üzerinden alınabilmiştir.

Bütçe dağılımının hesaplanması için satın alma ile sonuçlanan işlemleri içeren veri kümesi bu çalışmada tasarladığımız öğrenme modeline girdi olarak verilmiştir. Girdi olarak kullanılacak değerler öncelikle kullanıcı kimlik bilgisine göre gruplanmıştır. Bunun amacı, her kullanıcı için kanal etkisinin ayrı ayrı hesaplanarak karşılaştırmaya uygun biçime dönüştürülebilmesidir. LSTM model çıktısı, kullanıcı için satın alma kanal etkisini temsil etmektedir. Her grup için elde edilen bu çıktılar, ilgili kanal için



Şekil 4.8 Özellik azaltma için oluşturulan PCA modeli



Şekil 4.9 Özellik azaltma için oluşturulan otokodlayıcı modelleri

toplam etkiyi gösteren değere eklenmiştir. Tüm gruplar için işlem tamamlandıktan sonra son etki değeri ve kanalların tüm satın almaya etkisi ayrı ayrı hesaplanmıştır. Yapılan bu işlemlere ait yarı-kod Şekil 4.10'de verilmiştir.

```
Çıktı: Her bir kanalın satın alıma etkisi {c1, c2, .., cn}

Initialize channel effectt array C[] = 0
inputs = group_by_client_id(user_ads_data)
for each input(i) in the inputs do
    new_features = feature reduction(i)
    channels = lstm(new_features)
    for each channel(index)  in channels do
        c_effect = channels[index]
        C[index] = C[index] + c_effect
    end for
end for

C = normalize(C)
percenteges = calculate_percentege(C)
```

Şekil 4.10 Her kanalın satın almaya etkisinin hesaplanması

Modelin her eğitimi sonrası hata düzeltmeleri için kayıp fonksiyonu olarak mean-squared error (MSE) kullanılmış ve güncellemeler yapılmıştır. Eşitlik 4.9'de bağıntı verilmiştir. LSTM modelinin performansını karşılaştırmak için uzmanlar tarafından elde edilen sonuçlar kullanılmış ve bu sonuçlar ile karşılaştırılmıştır. N özellik sayısını, i seçilen özelliği, Y beklenen çıktıyı ve $\hat{Y}$ elde edilen çıktıyı temsil

etmektedir.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(Y_i - \hat{Y}_i)^2 \tag{4.9}$$

5
DENEYSEL SONUÇLAR

Bu tez çalışmasında çevrimiçi alışverişte farklı reklam kanallarının tüketicinin ürün satın almasına etkisini belirlemeye yönelik derin öğrenme tabanlı bir model tasarlanmış ve gerçeklenmiştir.

Geliştirilen modelin eğitilmesi ve test edilmesi için toplam 50GB büyüklüğünde, kullanıcı web sitesi etkileşimlerini içeren veri kümesi kullanılmıştır. Bu veri kümesi içinden, zaman serisine dayalı analizlerin karşılaştırılabilmesi için 15, 25, 45 ve 60 günlük veri alt kümeleri oluşturulmuştur. Model performansının ölçülmesi için ilk ve son tıklama sayısına göre ve uzman analizine göre reklam etki değerleri elde edilmiştir.

Model eğitimi ve testi için veriyi bellekte tutup işleyebilecek bir sistem üzerinde çalışılmıştır. İlgili sistem özellikleri Tablo 5.1'de gösterilmiştir.

Tablo 5.1 Modelin eğitimi için kullanılan deney ortamı

Çekirdek Sayısı	8
Bellek(RAM)	32GB
Disk	1TB
Grafik Kartı	RTX2080
İşletim Sistemi	Windows 11

Model geliştirimeleri Python[1] dilinde, Pytorch[2] ve scikit-learn[3] kütüphaneleri kullanılarak yapılmıştır.

Elle yapılan değerlendirme ile modelin performansının karşılaştırılması için ilk ve son satın alınan reklam kanalları bulunmuştur. Bu yaklaşımlar çerçevesinde kullanıcıların reklamlar ile ilk ve son etkileşim noktaları yüzdesel olarak Tablo 5.2'de gösterilmiştir. Tabloya dikkat edildiğinde kanal1 ve kanal2 için oranlar çok yüksek iken kalan kanallar için oranlar ya çok düşük ya da sıfır noktasındadır. Yapılan uzman analizlerinde kanal4'ün etkisinin daha yüksek çıkması beklenmektedir. Bu

[1]https://www.python.org/
[2]https://pytorch.org/
[3]https://scikit-learn.org/

gözlemden yola çıkarak, mevcutta olan ilk ve son etkileşim noktalarını göz önüne alan yaklaşımların yanlış olduğu anlaşılmaktadır.

Tablo 5.2 İlk ve son tıklanan reklam kanalının hesaplanması sonucu elde edilen sonuçlar

Reklam Kanalı	Etkileşim Oranı(%)
Kanal1	%41.8
Kanal2	%54.7
Kanal3	0
Kanal4	%3.5
Kanal5	0

Diğer bir karşılaştırma değeri uzmanlar tarafından hesaplanan ve elde edilen sonuçlardır. Kanal etki değerleri hesaplanırken her reklam platformunun analiz araçları kullanılmıştır. Reklam kanalı etki değerleri hesaplanırken CPC, CPA, CR ve CTR değerleri kullanılmıştır. Son olarak uzman analizi yapılmış ve Tablo 5.3'de gösterilen sonuçlar elde edilmiştir.

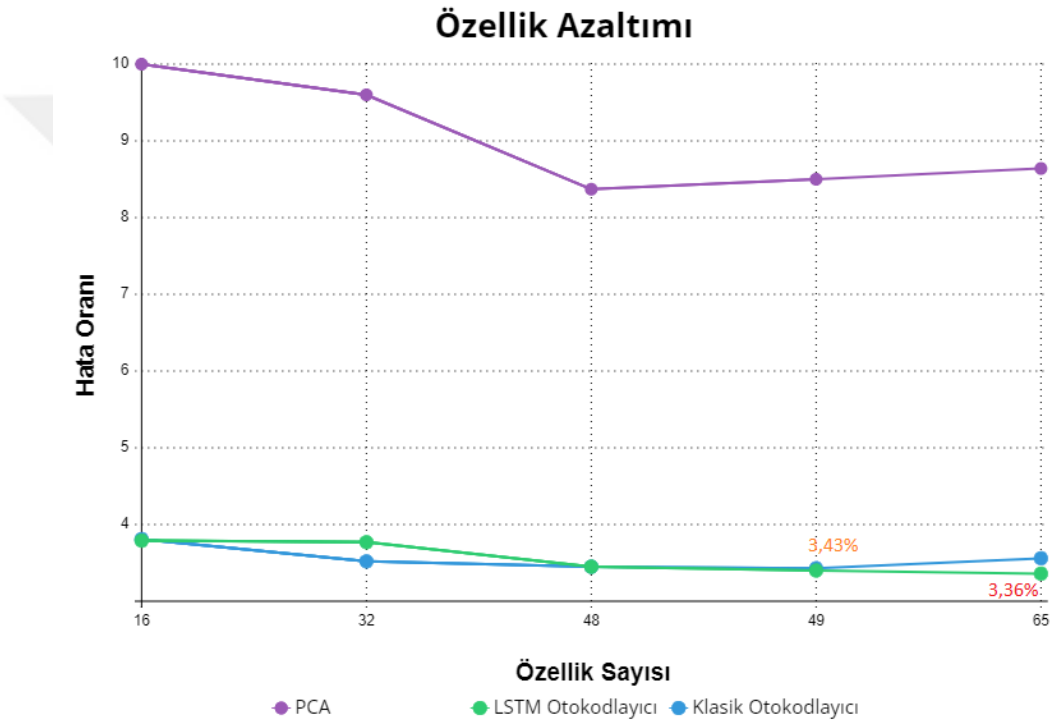
Tablo 5.3 Uzmanlar tarafından yapılan hesaplamalar sonucu elde edilen reklam kanalı etkileri

Reklam Kanalı	Etkileşim Oranı(%)
Kanal1	%38
Kanal2	%35
Kanal3	0
Kanal4	%27
Kanal5	0

Bu tez çalışması için geliştirilen LSTM tabanlı öğrenme modeli 5 çıkış sınıfı olan bir ağdır. Giriş değerleri için özellik azaltımı modellerinden elde edilen değerler kullanılmıştır. Çıktılardan reklam kanallarını temsil etmektedir. DonanimHaber, Adwords, Cimri, Facebook ve Instagram çıktı olarak kullanılan reklam kanallarıdır. Kullanılan veri kümesi tüm verinin %30'u test verisi olacak biçimde bölünmüştür. Veri kümesi bölünürken hem test hem de eğitim veri kümesi için tüm zaman aralığını kapsamasına dikkat edilmiştir. Geliştirilen modele girdi olarak kullanılan özellikler kullanıcıların site içeresindeki tüm etkileşimlerini içermektedir.

Bu tez çalışmasında özellik azaltım modeli seçimi için LSTM tabanlı otokodlayıcı, klasik otokodlayıcı ve PCA modelleri geliştirilmiştir. Bunun için 60 günlük veri seti kullanılmıştır. Özellik azaltımı modelleriyle yapılan denemler sonucu elde edilen hata oranları Şekil 5.1'de gösterilmiştir. LSTM tabanlı otokodlayıcı, özellik sayısının azaltılmadığı durumda daha iyi sonuç vermiştir fakat özellik sayısı azaltıldıkça hata oranının klasik otokodlayıcıya göre daha hızlı arttığı gözlemlenmiştir. LSTM tabanlı

otokodalyıcı için özellik azaltılmadığı durumda %3.36 hata oranı elde edilmiştir. Klasik otokodlayıcı için ise başlangıçta hata oranı daha yüksek olmasına rağmen özellik sayısı 48 ve 32'ye düşürüldüğü durumda, başlangıç durumuna göre daha iyi sonuçlar vermiştir. Klasik otokodlayıcı için hatanın nereye kadar azaldığını gözlemlemek adına 48 - 65 ve 32 - 48 özellik sayısı aralıkları için detaylı testler yapılmış ve hatanın 49 özellik sayısında en düşük seviye olan %3,43 hata oran seviyesine indiği gözlemlenmiştir. PCA ise otokodlayıcılara göre daha yüksek hata oranına sebep olmuştur. Elde edilen bu sonuçlar, özellik azaltımının kullanılabileceğini ve otokodlayıcıların bu problem özelinde PCA'e göre daha iyi sonuçlar verdiği gözlemlenmiştir.



Şekil 5.1 LSTM tabanlı otokodlayıcı, klasik otokodlayıcı ve PCA ile özellik azaltımı sonucu elde edilen hata oranları

En iyi modelin seçilebilmesi için veri kümesi farklı iterasyon ve farklı büyüklüklerde eğitilmiştir. Tablo 5.4'de denenen tüm durumlar için toplam veri, kullanıcı grup ve iterasyon sayısı gösterilmiştir. Eğitim sürecinde LSTM otokodlayıcı kullanan model tercih edilmiştir. Bu tercihin yapılma sebebi, en iyi sonuçların LSTM tabanlı otokodlayıcı modeliyle alınmasıdır. İterasyon sayısı arttıkça hata oranları azaldığı fark edilmiştir. 10 iterasyondan sonra hata oranında azalmanın sınırlı bir alanda kaldığı gözlemlenmiştir. Eğitim sürelerinin 22 saate kadar çıkması sebebiyle, sezonsal analiz ihtiyacı durumlarında daha hızlı eğitim yapabilmek adına en uygun modeller olarak 10 iterasyonla eğitilmiş modeller seçilmiştir. Tüm başarı karşılaştırmaları 10 iterasyona sahip modeller ile yapılmıştır.

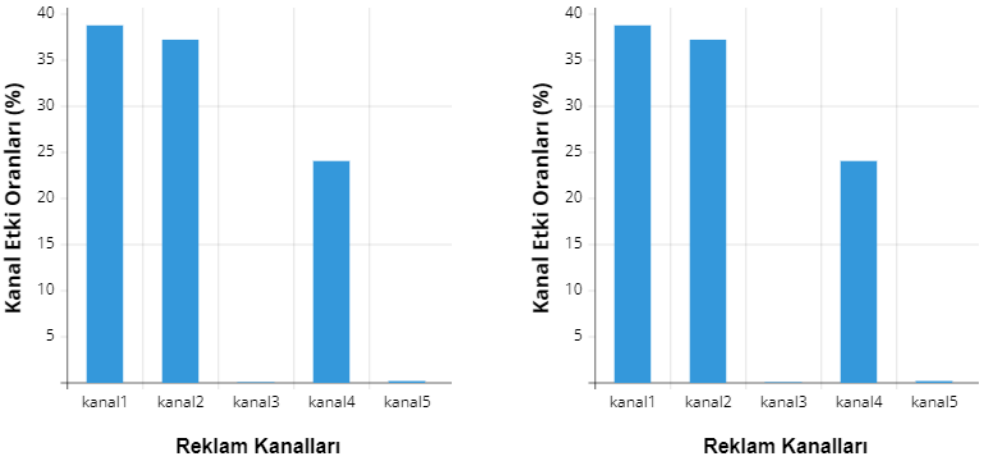
Tablo 5.4 Veri ve iterasyon sayısı

Veri Sayısı	Tekil Kullanıcı Sayısı	İterasyon	Hata Oranı	Eğitim Süresi
5,000,000	68,000	20	2,05	7.5 Saat
5,000,000	68,000	15	2,13	6 Saat
5,000,000	68,000	10	2,27	4.5 Saat
10,000,000	131,000	20	5,03	13 Saat
10,000,000	131,000	10	5,29	7.5 Saat
15,000,000	201,000	20	5.20	19 Saat
15,000,000	201,000	10	5.34	11 Saat
20,617,395	**271,000**	**20**	**3.13**	**24.5 Saat**
20,617,395	271,000	15	3.27	19 Saat
20,617,395	271,000	10	3.36	13.5 Saat

Temel derin öğrenme modeli seçimi için BiLSTM ve LSTM temelli modeller üzerinde aynı veri kümesi ile denemeler yapılmıştır. Her iki model için de otokodlayıcı özellik azaltımı modeli olarak kullanılmıştır. Çalıştırılan modeller sonucu çok yakın değerler elde edilmiştir. Bu sonuçlardan yola çıkarak tek yönlü verilerde BiLSTM modelinin, LSTM modelinden çok küçük farklara sahip olduğu ve neredeyse aynı sonucu elde ettiği çıkarımı yapılabilir. Sonuçlar Şekil 5.2'de gösterilmiştir. Şekil 5.2a ve Şekil 5.2c BiLSTM çıktılarını göstermektedir. Şekil 5.2b ve Şekil 5.2d LSTM çıktılarını göstermektedir.

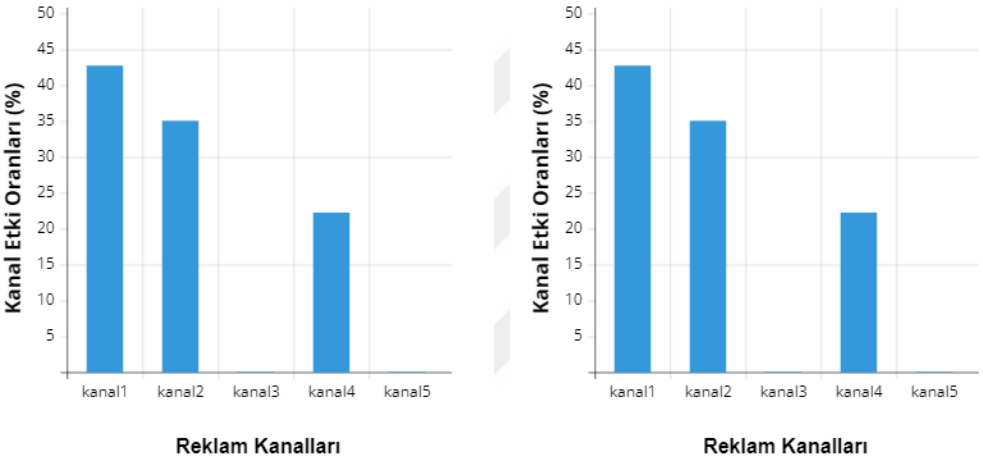
Sistemin ürettiği reklam kanallarının dönüşüme etkisini gösteren grafik Şekil 5.4 ve Şekil 5.3'de gösterilmiştir. Farklı zaman ve farklı modellerin sonuç üzerindeki etkileri gösterilmiştir. Buradan iki sonuç çıkarılabilmektedir. İlk olarak otokodlayıcı gibi derin öğrenme temelli öznitelik çıkarma modelleri kullanılması, veri üzerindeki önemli noktaların daha ön plana çıkarılmasını sağlamakta ve bu sayede üretilen modellerin daha iyi çıktılar vermesini sağlamaktadır. 5.3a'da bu durum gösterilmiştir. Diğer bir konu ise veri zaman aralığıdır. Zamana dayalı analizlerde veri uzunluğu doğrudan etkilidir. Bu nedenle analiz yaparken çıktı alınmak istenen zaman aralığının seçilmesi çok önemlidir. 5.3a ve 5.4a 60 günlük veri kümesi için üretilen modellerin sonuçlarını, 5.3b ve 5.4b 45 günlük bir zaman aralığı için üretilen modellerin sonuçlarını, 5.3c ve 5.4c 25 günlük bir zaman aralığı için üretilen modellerin sonuçlarını, 5.3d ve 5.4d 15 günlük bir zaman aralığı için üretilen modellerin sonuçlarını göstermektedir. Bu sonuçlar çerçevesinde elde edilen net çıkarım otokodlayıcı kullanılan modeller en iyi sonucu verdiğidir. Zaman dağılımı ise sonuçları iyileştirmek için önemlidir.

Şekil 5.5'de elde edilen sonuçlar ise kanallara göre sistemin hata çıktılarını göstermektedir. Kanal 3 ve kanal 5 etkiye sahip olmadığı için değerlendirmeye alınmamıştır. Bu çıktılardan elde edilen sonuçlar ise kanal bazlı olarak farklı zaman aralığında farklı sonuçlar alındığı yönündedir. En iyi sonuçlar 15 günlük ve 60 günlük



(a) 15 Günlük Veri

(b) 15 Günlük Veri



(c) 60 Günlük Veri

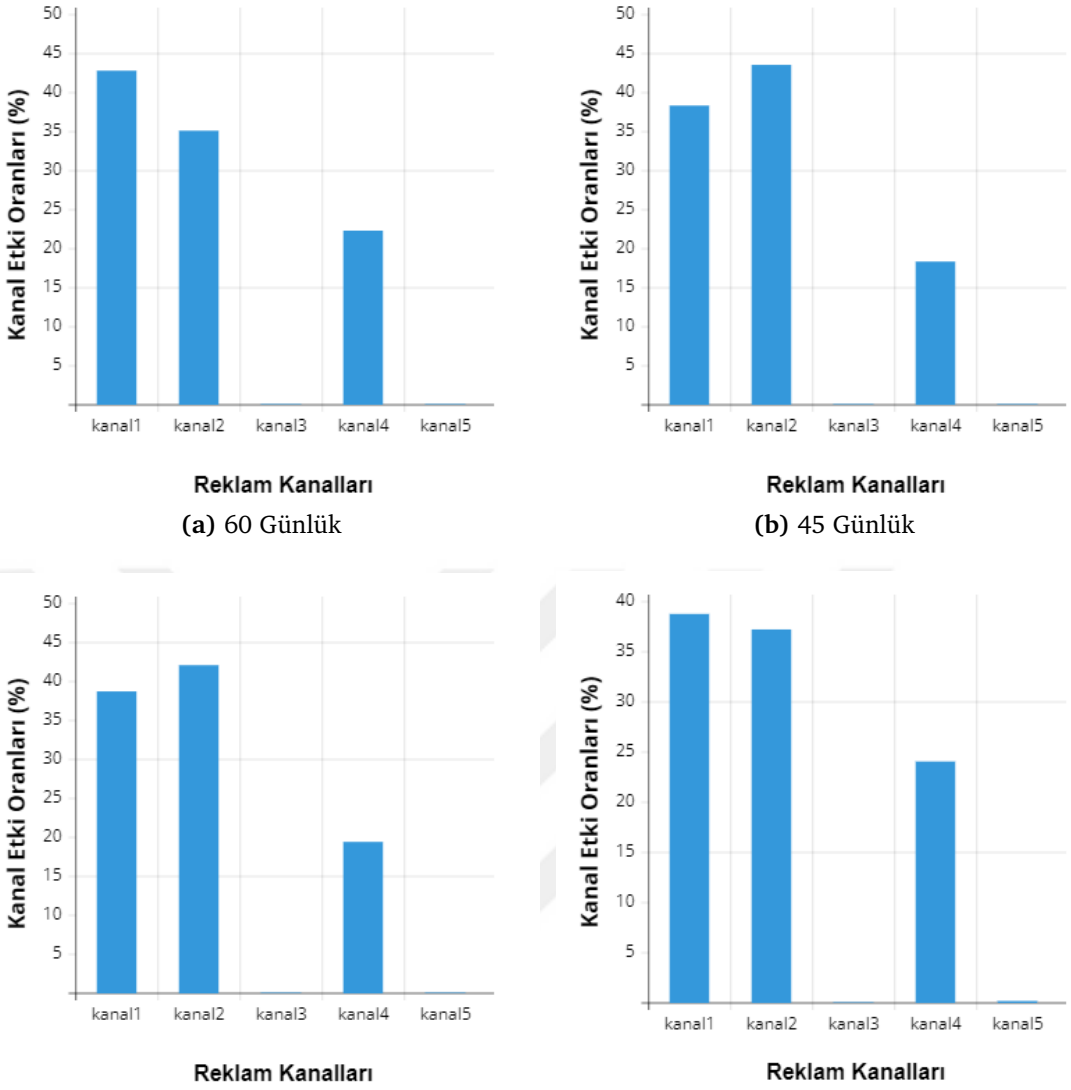
(d) 60 Günlük Veri

Şekil 5.2 BiLSTM ve LSTM modellerinin karşılaştırılması. a ve c BiLSTM, b ve d LSTM kanal etki sonuçlarını göstermektedir

veri kümeleri ile eğitilen modeller ile alınmıştır. Buradan çıkarılacak sonuç, reklamın verileceği zaman aralığına göre modelin eğitilip o aralık için sistemin çalıştırılması yönündedir.

Tablo 5.5'de PCA, LSTM tabanlı otokodlayıcı ve özellik azaltımı kullanılmadan oluşturulan LSTM modelleri için hata ve başarı detayları verilmiştir. En iyi sonuçları LSTM tabanlı otokodlayıcı verdiği için karşılaştırmada kullanılmıştır. Karşılaştırmalar için en iyi sonuçları veren veri kümelerinden biri olan 60 günlük veri kümesi kullanılmıştır. Özellik azaltımı kullanan modellerin hepsi özellik azaltımı kullanmayan modele göre daha iyi sonuçlar vermiştir.

PCA, LSTM tabanlı otokodlayıcı ve klasik otokodlayıcı kullanan modellerin karşılaştırılması için hem 60 günlük veri kümesi üzerinde hem de 15 günlük veri kümesi üzerinde denemeler yapılmıştır. İki durumda da en iyi sonucu LSTM tabanlı



(a) 60 Günlük **(b)** 45 Günlük

(c) 25 Günlük **(d)** 15 Günlük

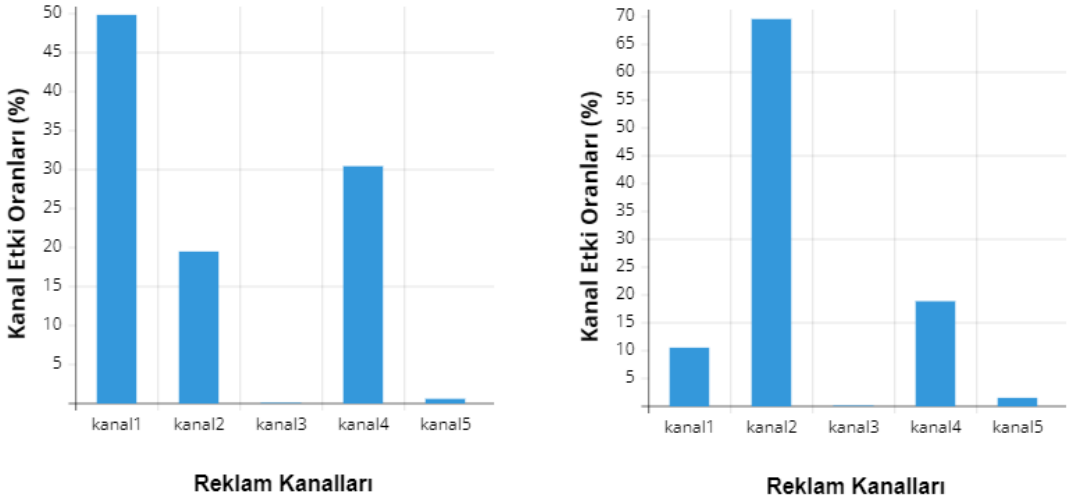
Şekil 5.3 LSTM tabanlı otokodlayıcı kullanan derin öğrenme modeli için reklam kanallarının etkisine göre bütçe dağılımı

Tablo 5.5 PCA, otokodlayıcı ve özellik azaltımı kullanılmadan oluşturulan modeller için 60 günlük veri ile eğitilen model karşılaştırması

Kanal	Uzman Değerlemesi	Model Tahmini				Hata Oranı			
		PCA	LSTM Otokodlayıcı	Klasik Otokodlayıcı	LSTM	PCA	LSTM Otokodlayıcı	Klasik Otokodlayıcı	LSTM
Kanal1	38	56,05	42,55	39,35	50	18,05	4,55	1,35	12
Kanal2	35	31,95	34,04	38,96	19	3,05	0,96	3,96	15
Kanal3	0	0	0	0	0	0	0	0	0
Kanal4	27	11,77	22,41	21,66	28	15,23	4,59	5,34	1
Kanal5	0	0,33	0	0,03	2	0,33	0	0,03	2

otokodlayıcı kullanan model vermiştir. Tablo 5.6'de alınan sonuçlar ve hata oranları gösterilmiştir. Her iki model için de karşılaştırma yapılabilmesi için, özellik azaltımı aşamasında elde edilecek özellik sayısı 32 olacak biçimde yapılandırılmıştır.

Tüm bu sonuçlar en iyi modelin otokodlayıcı ile özellik azaltımı yapan derin



(a) 60 Günlük **(b)** 45 Günlük

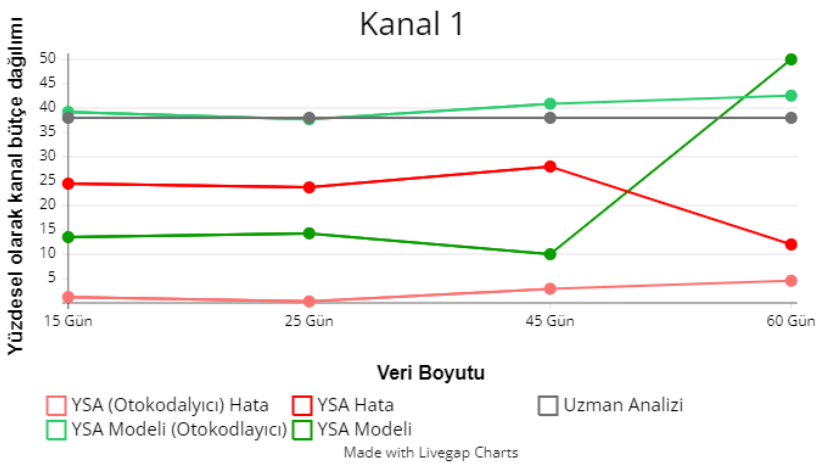
(c) 25 Günlük **(d)** 15 Günlük

Şekil 5.4 Özellik azaltımı kullanmayan derin öğrenme modeli için reklam kanallarının etkisine göre bütçe dağılımı

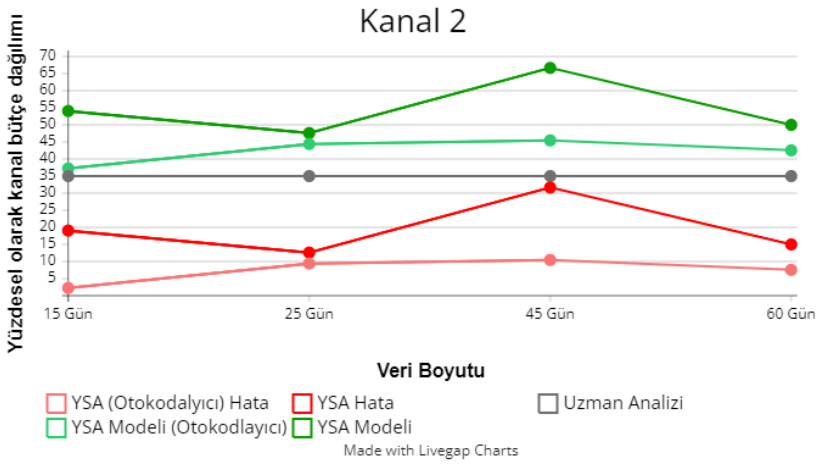
Tablo 5.6 PCA ve otokodlayıcı hata karşılaştırması

Kanal	Uzman Değerlemesi	Model Tahmini						Toplam Hata		
		60 Gün			15 Gün					
		PCA	LSTM Otokodlayıcı	Klasik Otokodlayıcı	PCA	LSTM Otokodlayıcı	Klasik Otokodlayıcı	PCA	LSTM Otokodlayıcı	Klasik Otokodlayıcı
Kanal1	38	56,05	42,55	39,35	54,94	38,91	39,00	34,99	4,46	2,35
Kanal2	35	31,95	34,04	38,96	26,92	37,35	38,16	11,13	3,31	7,12
Kanal3	0	0	0	0	2,19	0	0	2,19	0	0
Kanal4	27	11,77	22,41	21,66	15,95	23,73	22,00	26,28	7,86	10,34
Kanal5	0	0,33	0	0,03	0	0,01	0,84	0,33	0,01	0,87

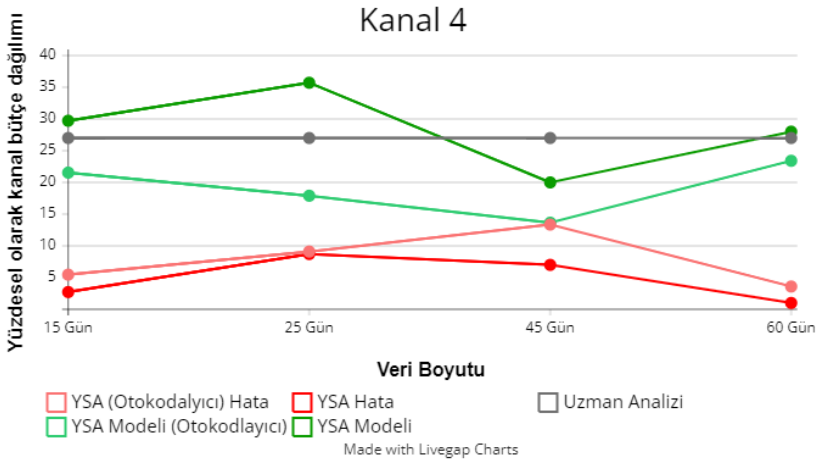
öğrenme modeli olduğunu göstermiştir. El yordamı ile yapılan uzman ve ilk veya son gelinen reklam kanalı analizleri sonuçları ile derin öğrenme modeli sonuçları karşılaştırılmıştır. Elde edilen sonuçlar Şekil 5.6'de verilmiştir. Grafikten elde edilen sonuçlar neticesinde hem otokodlayıcı kullanılan hem de kullanılmayan derin öğrenme modellerinin başarılı sonuçlar verebildiği gözlemlenmiştir. Fakat, otokodlayıcı kullanılarak üretilen modelin çok daha başarılı olduğu ve kanal 5 gibi



(a) Kanal 1



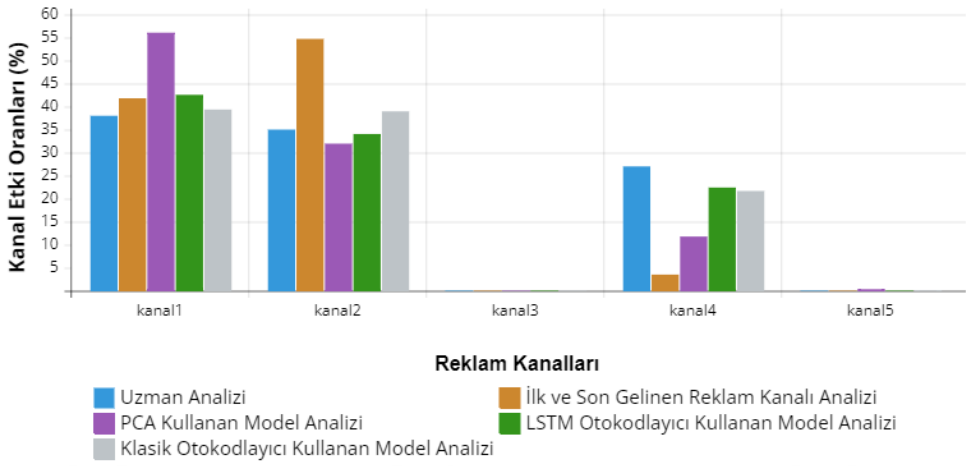
(b) Kanal 2



(c) Kanal 4

Şekil 5.5 Kanallara göre otokodlayıcı olan ve olmayan modeller için veri boyutu bazlı hata grafiği

aslında önemi olmayan reklam kanallarının da göz ardı edilebilmesini sağlamıştır. Bu durum da genel başarı oranını arttırmıştır.



Şekil 5.6 60 günlük veri üzerinde yapılan reklam kanallarının bütçe dağılımı. Önerilen model analizi çubuğu tez kapsamında geliştirilen model sonucunu, uzman analizi çubuğu konunun uzmanlarının elde ettikleri sonucu ve ilk veya son gelinen reklam kanalı analizi çubuğu da satın alım öncesi ilk ve son etkileşim olan kanal etki sonucunu göstermektedir.

Yapılan değerlendirmeler sonucunda kanal etki analizi yapılırken sezonsal değişiklikler olduğu gözlemlenmiştir. Bunun için Şekil 5.5'de elde edilen sonuçlar değerlendirilmiştir. 15 günlük ve 60 günlük veri kümesi ile eğitilen model üzerinde yapılan analizlerde uzman değerlendirmesine en yakın sonuçlar elde edilmiştir. 25 ve 45 günlük veri kümesi ile eğitilen model üzerinde yapılan analizlerde ise uzman değerlendirmesi sonucu beklenen değerlere göre hataların arttığı gözlemlenmiştir. Buradan elde edilen sonuç ise belirli aralıklarda kullanıcı davranışlarında farklılıklar olduğu yönündedir.

Tüm bu sonuçlar özellik azaltma modeli olarak otokodlayıcı kullanan derim öğrenme modellerinin daha başarılı sonuçlar verdiğini, bu problem için kullanılabilecek bir yaklaşım olduğunu göstermiştir.

6
SONUÇ VE ÖNERİLER

Bu tez çalışmasında yapay zekâ çalışmalarının son yıllarda önem kazandığı, fakat veri kümesi eldesinin güçlüğü nedeniyle çok fazla çalışma yapılamayan çevrimiçi reklamcılık alanında, kullanıcının satın almasına reklam kanallarının etkisini tespit etmeye yönelik bir sistem tasarlanmış ve gerçeklenmiştir. Elde edilen sonuçlar umut verici olmakla birlikte, sonuçlardan yola çıkarak bütçe dağılım önerilerinin gerçek anlamda ne kadar etkili olduğunu gözlemlemek için uzun süreli denemeler yapılmaya devam edilmesi gerekmektedir.

Mevcut durumda çevrimiçi reklamlarda kullanıcının satın alma sürecinde kanalların etkisinin hesaplanması elle yapılan ve günlerce süren analizlere dayalı bir işlemdir. Elde edilen sonuçlar sayesinde dakikalar içinde ve çok büyük veri üzerinde çalışabilecek bir model gerçeklenmiştir ve bu model mevcut sorunlar karşısında alternatif bir yöntem olarak sunulmuştur. Bu sayede zaman ve veriden kaynaklı sorunların önüne geçilmesi sağlanmıştır. Bu sonuçlar sayesinde buraya ayrılan gereksiz kaynağın önüne geçilmesi için zemin hazırlanmıştır.

Bu tez çalışmasının diğer çalışmalardan farklılaştığı önemli noktalardan biri otokodlayıcı ile özellik çıkarımı yapılmasıdır. Derin öğrenme modelinin otokodlayıcı ile desteklenmesi sayesinde sistem performansı arttırılmıştır ve daha iyi sonuçlar elde edilebilmiştir.

Bu tez çalışmasıyla model geliştirilmesi üzerinde elde edilen temel iki sonuç vardır. İlk olarak zamana dayalı çalışmalarda verinin toplandığı zaman aralığı sistem performansına doğrudan etki etmektedir. Bu nedenle sezonsal analizlerin de yapılması gerektiği anlaşılmıştır. Veri büyüklüğü yanında reklamın verileceği zaman aralığına göre de sistemi eğitmek gerekmektedir. Diğer bir sonuç ise otokodlayıcı gibi özellik çıkarımı yöntemleri ile sistem performansı önemli ölçüde geliştirilebilmiştir. Bu tarz sistemlere özel geliştirmeler yapılması daha iyi sonuçlar elde edilmesini sağlayacaktır.

Bu çalışmadan yola çıkarak gelecek çalışmalarda aşağıdaki geliştirmeler yapılabilir;

- Mevcut sistemde sadece kullanıcı site etkileşimi verisi kullanılabilmiştir. Buna ek olarak mobil uygulamalar için de geliştirme yapmak gereklidir.
- Toplanan veride satın alınmış ve daha sonra iptal edilmiş olan durumların bilgisi bulunmamaktadır. Bu verinin de eklenmesi sistem doğruluğunu arttıracaktır.
- Satın alım yapan kullanıcıların harcadığı para miktarı da kanal seçiminde önemli rol oynamaktadır. Bunun için fiyat bilgisi dahilinde bir model geliştirilmesi faydalı olacaktır.
- Önemli sezonlarda dönüşüm oranlarının tespiti için daha geniş bir veri kümesi ile çalışma yapılması önemli bir nokta olacaktır.
- KVKK kaynaklı anonim tutulmak zorunda kalınan veri kümeleri üzerinde kullanıcı tekilleştirme işlemi yapılamamaktadır. Bu nedenle anonim veriler üzerinde analiz yapabilecek derin öğrenme modelleri de geliştirilmelidir.

KAYNAKÇA

[1] A. Li, P. K. Kannan, "Attributing conversions in a multichannel online marketing environment: An empirical model and a field experiment," *Journal of Marketing Research*, vol. 51, pp. 40–56, Feb. 2014. DOI: 10.1509/jmr.13.0050.

[2] R. Berman, "Beyond the last touch: Attribution in online advertising," *SSRN Electronic Journal*, Jan. 2013. DOI: 10.2139/ssrn.2384211.

[3] X. Shao, L. Li, "Data-driven multi-touch attribution models," in *KDD '11: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, Aug. 2011, pp. 258–264. DOI: 10.1145/2020408.2020453.

[4] R. Singal, O. Besbes, A. Desir, V. Goyal, G. Iyengar, "Shapley meets uniform: An axiomatic framework for attribution in online advertising," *Management Science*, Jan. 2022. DOI: 10.1287/mnsc.2021.4263.

[5] B. Dalessandro, C. Perlich, O. Stitelman, F. Provost, "Causally motivated attribution for online advertising," in *Proceedings of the Sixth International Workshop on Data Mining for Online Advertising and Internet Economy*, ser. ADKDD '12, Beijing, China: Association for Computing Machinery, 2012, ISBN: 9781450315456. DOI: 10.1145/2351356.2351363.

[6] D. B. Paul, "The selection of the "survival of the fittest"," *Journal of the History of Biology*, vol. 21, no. 3, pp. 411–424, 1988.

[7] W. Ji, X. Wang, D. Zhang, "A probabilistic multi-touch attribution model for online advertising," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, ser. CIKM '16, Indianapolis, Indiana, USA: Association for Computing Machinery, 2016, pp. 1373–1382, ISBN: 9781450340731. DOI: 10.1145/2983323.2983787.

[8] W. Ji, X. Wang, "Additional multi-touch attribution for online advertising," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, ser. AAAI'17, San Francisco, California, USA: AAAI Press, 2017, pp. 1360–366.

[9] Y. Zhang, Y. Wei, J. Ren, "Multi-touch attribution in online advertising with survival theory," in *2014 IEEE International Conference on Data Mining*, 2014, pp. 687–696. DOI: 10.1109/ICDM.2014.130.

[10] N. Li, S. K. Arava, C. Dong, Z. Yan, A. Pani, "Deep neural net with attention for multi-channel multitouch attribution," in *KDD Conference, London*, 2018.

[11] A. Vaswani *et al.*, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, Long Beach, California, USA: Curran Associates Inc., 2017, pp. 6000–6010, ISBN: 9781510860964.

[12] M. Hejazinia, P. Mitsoulis-Ntompos, S. Zhang, "Deep personalized re-targeting," in *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ser. ASONAM '19, Vancouver, British Columbia, Canada: Association for Computing Machinery, 2019, pp. 1148–1154, ISBN: 9781450368681. DOI: 10.1145/3341161.3345617.

[13] D. Yang, K. Dyer, S. Wang, *Interpretable deep learning model for online multi-touch attribution*, 2020. arXiv: 2004.00384 [cs.IR]. [Çevrimiçi]. Erişilebilir: https://arxiv.org/abs/2004.00384 (Erişim Tarihi: 26.07.2022).

[14] K. Ren *et al.*, "Learning multi-touch conversion attribution with dual-attention mechanisms for online advertising," *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, Oct. 2018. DOI: 10.1145/3269206.3271677.

[15] T. Thornton, S. Thorne, A. Calderon, "Modelling user behaviour in market attribution: Finding novel data features using machine learning," May 2021. [Çevrimiçi]. Erişilebilir: https://aisel.aisnet.org/ukais2021/18/ (Erişim Tarihi: 22.08.2022).

[16] H. Choi, C. F. Mela, S. R. Balseiro, A. Leary, "Online display advertising markets: A literature review and future directions," *Information Systems Research*, vol. 31, no. 2, pp. 556–575, 2020.

[17] K. Shvachko, H. Kuang, S. Radia, R. Chansler, "The hadoop distributed file system," in *2010 IEEE 26th symposium on mass storage systems and technologies (MSST)*, Ieee, 2010, pp. 1–10.

[18] C. Seger, *An investigation of categorical variable encoding techniques in machine learning: Binary versus one-hot and feature hashing*, 2018. [Çevrimiçi]. Erişilebilir: https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A1259073 (Erişim Tarihi: 22.08.2022).

[19] H. Abdi, L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[20] D. E. Rumelhart, G. E. Hinton, R. J. Williams, "Learning internal representations by error propagation," California Univ San Diego La Jolla Inst for Cognitive Science, Tech. Rep., 1985.

[21] N. Srivastava, E. Mansimov, R. Salakhutdinov, "Unsupervised learning of video representations using lstms," in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML'15, Lille, France: JMLR.org, 2015, pp. 843–852.

[22] D. E. Rumelhart, G. E. Hinton, R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[23] L. Tai, J. Zhang, M. Liu, J. Boedecker, W. Burgard, *A survey of deep network solutions for learning control in robotics: From reinforcement to imitation*, 2016. DOI: 10.48550/ARXIV.1612.07139. [Çevrimiçi]. Erişilebilir: https://arxiv.org/abs/1612.07139 (Erişim Tarihi: 22.08.2022).

[24] S. Hochreiter, J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, pp. 1735–80, Dec. 1997. DOI: 10.1162/neco.1997.9.8.1735.

[25] Z. Cui, R. Ke, Z. Pu, Y. Wang, *Deep bidirectional and unidirectional lstm recurrent neural network for network-wide traffic speed prediction*, 2018. DOI: 10.48550/ARXIV.1801.02143. [Çevrimiçi]. Erişilebilir: https://arxiv.org/abs/1801.02143 (Erişim Tarihi: 22.08.2022).

[26] M. Schuster, K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

TEZDEN ÜRETİLMİŞ YAYINLAR

Konferans Bildirisi

1. O. Kahraman ve M. E. Karslıgil, "Derin Öğrenme ile Çevrimiçi Reklamlarda Kullanıcı Dönüşüm Oranları Üzerindeki Kanal Etkilerinin Belirlenmesi", 2022 30th Signal Processing and Communications Applications Conference (SIU), 2022