Journal of the Faculty of Engineering and Architecture of Gazi University 37:3 (2022) 1639-1654

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

A new document classification algorithm against malicious data leakage attacks

Yahya Kesenek[1]*, İbrahim Özçelik[2], Emrah Kaya[1]

[1]Computer and Informatics Engineering, Institute of Natural Sciences, Sakarya University, Sakarya, 54050, Turkey
[2]Faculty of Computer and Information Science, Department of Computer Engineering, Sakarya University, Sakarya, 54050, Turkey

Highlights:

- An algorithm about against content-based attack is present firstly
- Features extracted with NLP methods
- Content based attacks extended.

Keywords:

- Data Leakage Prevention
- Malicious DLP
- Advanced Persistent Thread
- APT
- Structural Evasion Attack

Article Info:

Research article
Received: 01.11.2019
Accepted: 24.10.2021

DOI:

10.17341/gazimmfd.641580

Correspondence:

Author:Yahya Kesenek
e-mail: yahyakesenek@gmail.com
phone: +90 506 367 1167

Graphical/Tabular Abstract

Nowadays it is important to store sensitive data and restrict its usage only to authorized people or institutions. In general, solutions for Data Leakage Prevention (DLP) ignores malicious attacks on documents and algorithms using fingerprinting and regular expressions are used. However, content-based attacks are successful evading those algorithms. We developed a novel algorithm for classifying documents successfully under scoped attack types.

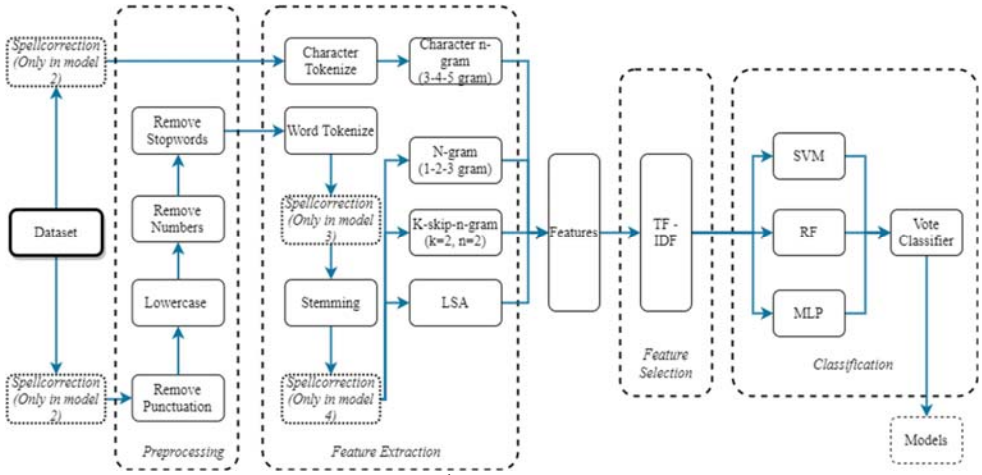


Figure A. Proposed algorithm

Purpose: In this paper, a novel algorithm against malicious content-based attacks is proposed, which is robust independent of the attack executed. Transposition, sentence structure alteration, modification, obfuscation attacks are taken into consideration within the scope of paper.

Theory and Methods:
Malware-based attacks are re-organized and attack types are shown in a schema. In this study, two type of attacks, which are structural attack and obfuscated attack are scoped. A software has been developed to carry out these attacks. With this software, the documents were attacked and then the performance of the developed method was measured. In the developed algorithm, Natural Language Processing (NLP) methods, machine learning and artificial neural networks were used. Natural language processing algorithms, which are commonly used by text-based classification systems, are used in the extraction of features. Later, Decision Support Machines (SVM), Random Forest and Multi-Layer Perceptron were used in the classification model. In these classification models, the decision mechanism is provided by Vote Classifier.

Results:
The reliability of the algorithm was compared with several methods used in data leakage prevention systems and text classification algorithms and the success of the algorithm was measured. In the tests performed, the f1 score of the classification success of the proposed method was found to be 99%.

Conclusion:
Under Advanced Persistent Thread (APT) or malware attacks usage of standard feature extraction methods may not useful. Because these attacks may corrupt document or change document as it is. In our case we investigate Transposition, sentence structure alteration, modification, obfuscation attacks and we show that under attacks text classification methods are not outperforms than our proposed method.

Journal of the Faculty of Engineering and Architecture of Gazi University 37:3 (2022) 1639-1654

Mühendislik Mimarlık Fakültesi Dergisi
Journal of The Faculty of Engineering and Architecture of Gazi University

Elektronik / Online ISSN : 1304 - 4915
Basılı / Printed ISSN : 1300 - 1884

Zararlı yazılım kaynaklı veri kaçırma ataklarına karşı yeni bir doküman sınıflandırma algoritması

Yahya Kesenek[1]*, İbrahim Özçelik[2], Emrah Kaya[1]

[1]Sakarya Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar ve Bilişim Mühendisliği ABD, 54050 Serdivan Sakarya, Türkiye
[2]Sakarya Üniversitesi, Bilgisayar ve Bilişim Fakültesi, Bilgisayar Mühendisliği Bölümü, 54050, Serdivan Sakarya, Türkiye

Ö N E Ç I K A N L A R

- İçerik tabanlı bir algoritma oluşturuldu
- NLP yöntemlerine göre içerikler çıkarıldı
- İçerik tabanlı saldırılar genişletildi

Makale Bilgileri

Araştırma Makalesi
Geliş: 01.11.2019
Kabul: 24.10.2021

DOI:
10.17341/gazimmfd.641580

Anahtar Kelimeler:
Veri sızıntısı önleme,
zararlı DLP,
Gelişmiş ısrarlı saldırılar,
APT,
Yapısal kaçırma saldırıları

ÖZ

Günümüzde değerli verilerin saklanması ve sadece yetkili şahıs veya kurumlarca kullanılması önem arz etmektedir. Genel olarak verinin korunmasına yönelik geliştirilen veri sızıntısı önleme (Data Leakage Prevention-DLP) çözümleri zararlı yazılım kaynaklı saldırıları göz ardı etmekte, parmak izi (fingerprinting) eşleştirme ve kurallı ifadeler (regular expression) benzeri yöntemler kullanan algoritmalar yer almaktadır. Oysaki doküman içeriğine yönelik yapılan saldırılar neticesinde bu algoritmalar atlatılabilmektedir. Zararlı yazılım kaynaklı veri sızıntısına karşı dayanıklı bir algoritmanın sunulduğu bu çalışmada, zararlı yazılımların saldırı türüne bağlı kalmayan bir çözüm önerilmektedir. Çalışma kapsamında, yer değiştirme, yapısal cümle saldırıları, modifikasyon saldırıları, karartma saldırıları ele alınmıştır. Bu saldırılara karşı yazım düzeltimi, kelime-gram ve karakter-gram, k-skip-n-gram ve LSA kullanılarak, saldırı altında daha iyi bir sınıflama yapılması için özellikler çıkarımı sağlanmıştır. Çıkarılan özellikler, Karar Destek Makineleri, Rasgele Orman ve Çok Katmanlı Algılayıcı kullanarak Oylamalı Sınıflandırıcı ile en çok oy alan yönteme göre sınıflama tahmini yapılmıştır. Ayrıca modifikasyon saldırılarında faydalı olan Yazım Düzeltme yönteminin etkisi farklı aşamalarda uygulanarak gösterilmiştir.

A new document classification algorithm against malicious data leakage attacks

H I G H L I G H T S

- An algorithm about against content-based attack is present firstly
- Features extracted with NLP methods
- Content based attacks extended

Article Info

Research Article
Received: 01.11.2019
Accepted: 24.10.2021

DOI:
10.17341/gazimmfd.641580

Keywords:
Data leakage prevention
malicious DLP,
advanced persistent thread,
APT
structural evasion attack

ABSTRACT

Nowadays it is important to store sensitive data and restrict its usage only to authorized people or institutions. In general, solutions for Data Leakage Prevention (DLP) ignores malicious attacks on documents and algorithms using fingerprinting and regular expressions are used. However, content-based attacks are successful evading those algorithms. In this paper an algorithm robust against malicious content-based attacks is proposed, which is independent of the attack executed. Transposition, sentence structure alteration, modification, obfuscation attacks are taken into consideration within the scope of paper. N-gram, character-gram, k-skip-n-gram and LSA methods are used in the feature extraction step, for having better classification results under attacks. The extracted features are passed to a Vote Classifier consisting of Support Vector Machine, Random Forest and Multi-Layer Perceptron classifiers. Additionally, the effects of instrumenting Spell-Correction in different steps of the algorithm is evaluated, which is effective against modification attacks.

*Sorumlu Yazar/Yazarlar / Corresponding Author/Authors : *yahyakesenek@gmail.com, ozcelik@sakarya.edu.tr, emrah.kaya@gmail.com / Tel: +90 506 367 1167

1. GİRİŞ (INTRODUCTION)

Veri sızıntısı özellikle ulusal ve kurumsal ağlarda büyük sorunlara yol açabilmektedir. Veri kaybının yaşandığı kurumlarda hem ülke hem de kurumlar büyük yara alabilmektedir. Gizlilik arz eden bilgilerin kurum dışında veya içinde yetkisiz ellere geçmesi kuruma zarar vermektedir. Bu tür veri sızıntılarını önlemek için geliştirilen Veri Sızıntısı Önleme (*Data Leakage Prevention - DLP*) sistemleri günümüzde yaygın olarak kullanılmaktadır.

DLP çözümleri içerik (*content*) veya bağlam (*context*) temelli olarak karşımıza çıkmaktadır. Bağlam tabanlı sistemler üstveri (*metadata*) bilgilerini kullanarak veri kaybını önlemeye çalışırken içerik tabanlı sistemler istatistiksel yöntemler, düzenli ifadeler gibi yaklaşımları kullanarak veri kaybını tespit etmeye çalışır [1] Bağlam tabanlı yaklaşımlar, dokümanın oluşturulma tarihi, kim tarafından oluşturulduğu, kim veya kimler tarafından kullanabileceği gibi üstverilerini (metadata) kullanarak dokümandaki veri kaybını önlemeye çalışır. Bağlam tabanlı sistemlerde dokümanda yapılacak ufak bir değişiklikle dokümanın üstveri bilgileri değiştirilebilir bu işlem sonucunda sistemin atlatılması mümkün olabilmektedir. Ayrıca üstveri kullanılarak dokümanı okuma, yazma veya her iki işlemi yapabilen yetkililerin olduğu, yetki tabanlı sistemlerde yetkisiz bir kişinin yetki ihlali ile dokümana ulaşması durumu söz konusu olabilmektedir. Bu tür sorunlardan dolayı bağlam tabanlı bir sistemin tercih edilmesi uygun olmayacaktır.

İçerik tabanlı DLP sistemleri, parmak izi çıkarma (fingerprinting), kural tabanlı yaklaşım (Rule-Based/Regular expression), istatistiksel analiz (statistical analysis (n-gram, k-skip n-gram, TF-IDF)), tam dosya eşleştirme (exact file matching), parçalı doküman eşleştirme, kavramsal analiz (Conceptual/Lexicon) ve ön tanımlı kategoriler (Pre-built categories) yöntemlerini kullanır [2].

İçerik tabanlı yöntemler, veriye yönelik bir saldırı gerçekleştirildiğinde dokümanı tanımada etkisiz kalabilmektedir [1, 3]. Parmak izi çıkarma, ön tanımlı verilere bakarak yeni verileri eşleştirmeye çalışmaktadır. Veride yapılacak ufak bir değişiklik eşleştirme oranını düşürebilmektedir. Ayrıca ön tanımlı verilerin bir veri tabanında tutulması söz konusu olduğunda veri tabanına yazma ve okuma işlemleri performans kaybına yol açabilmektedir. Kural tabanlı sistemlerde bir uzman tarafından kuralların oluşturulması gerekmektedir. Ayrıca kural tabanlı sistemlerde, genel olarak vatandaşlık numarası, sosyal güvenlik numarası gibi ön tanımlı desenler kullanılmaktadır. Desen eşleşmesini engelleyecek saldırılar ile bu tür verilerin yapılarının değiştirilmesi durumunda desenlerin eşleşmesi mümkün olamamaktadır. Ayrıca tam veya parçalı doküman eşleştirmeye dayalı sistemlerde, içeriğin bir saldırı neticesinde değişmesinden dolayı eşleşme oranı azalabilmektedir. Bu çalışmada yukarıda bahsedilen zafiyetleri giderecek daha etkili bir algoritma geliştirmek amacıyla saldırı vektörlerine göre çözüm sunulmaktadır. Yapısal saldırılar neticesinde içeriğin değiştirilmesi söz konusu olduğunda dokümana ait özellikler sağlıklı olarak çıkarılamamaktadır. Bu sebeple, bu çalışmada, N-gram kullanımının başarılı olabilmesi için n-gram'ların çıkarılma aşamasında içeriğe yönelik saldırılar dikkate alınarak önişlemler yapılmaktadır. Bu aşamada dokümandaki yazım hatalarının düzeltilmesi ile daha sağlıklı bir sınıflama yapılabilmektedir. N-gram çıkarımı buna rağmen hâlâ yapılamıyorsa karakter gramlar kullanılması etkili olmaktadır.

Doküman özeti çıkarma şeklindeki yapısal saldırılarda ise içerik azaldığı için dokümanı sınıflandırmak için yeterli veri elde edilemeyebilmektedir. Ayrıca dokümandaki kelime veya cümlelerin yer değiştirilmesi ile n-gramların özellik vektörleri değişmekte ve sınıflandırıcı atlatılabilmektedir. Bu durumda k-skip-n-gramların kullanılması faydalı olmaktadır. Eş anlamlı kelimelerin kullanıldığı karartma saldırılarında ise kelimelerin tespitinde Gizli Anlam Analizi (Latent Semantic Analysis - LSA) yöntemi kullanılması ile dokümana ait örtük bilginin açığa çıkarılması sağlanmaktadır. Bunun için çalışmamızda sunulan algoritmanın Özellik Çıkarımı aşamasında n-gram, skip-gram, LSA ve karakter-gram özellikleri çıkarılmakta ve sınıflandırıcıda bir arada kullanılmaktadır.

İçerik tabanlı saldırı vektörlerinin çeşitliliği, saldırı sonrasında oluşan doküman ile orijinal doküman arasındaki farkın çok belirgin olmaması gibi sebeplerden dolayı saldırının tespit edilmesi zor olmakla beraber, birden fazla saldırı metodunun uygulanması söz konusu olduğunda saldırı vektörünün tespiti daha da güçleşmektedir. Bu durumda tüm saldırı vektörleri için tekil çözümünün bulunması daha uygun olmaktadır. Sunulan çözümde bahsedilen yöntemler bir arada kullanılarak farklı saldırı vektörlerine karşı dayanıklı bir özellik çıkarımı sağlanmıştır. Bunun yanında farklı durumlarda sınıflandırma başarımını arttırmak için sınıflandırıcıların oylamalı olarak kullanılması tercih edilmiştir. Yazım düzeltimi yönteminin saldırılara karşı başarılı olduğu ortaya konmakla beraber, algoritmanın farklı aşamalarında uygulanmasının etkisini gösterebilmek amacıyla çalışmanın test bölümünde 4 farklı şekilde uygulanarak test edilmiştir.

Çalışma şu şekilde organize edilmiştir: 2. bölümde literatür incelemesi özetlenmekte, 3. bölümde kullanılan yaklaşım anlatılmakta, 4. bölümde yapılan testler ve sonuçları verilmektedir.

2. LITERATÜR İNCELEMESİ (LITERATURE OVERVIEW)

Mevcut veri sızıntısı tespit sistemlerinin zararlı ya da doğrudan insan müdahalesi ile karartılması durumlarında başarılı olamadıkları çeşitli çalışmalarda ele alınmıştır. Radwan ve Yousef [4] veri sızıntısı ile ilgili yaptıkları literatür çalışmasında, mevcut veri sızıntısı çözümlerinin

karartılmış veri kaçaklarını bulamadığını ifade etmektedir. Ayrıca hassas verinin diğer dosyalarla değiştirilmesi sonucu da veri kaçaklarının gerçekleşebileceğinden bahsetmektedir.

Tarique Mustafa, [3] yaptığı çalışmada mevcut veri sızıntısı çözümlerinin, zararlı yazılımların değiştirdiği veriyi yakalamadaki eksikliklerini ortaya koymaktadır. Mevcut çözümlerin yetersiz kaldığı noktalar ve çözüm önerileri sunulmaktadır. Bir gerçeklemenin sunulmadığı bu çalışmalarda vurgulanan içerik tabanlı ataklar ve çözüm önerileri çalışmamızda dikkate alınmıştır.

Veri sızıntısı engelleme alanında ticari alanda birçok firma da veri sızıntısına yönelik çözümler üretmektedir. Bu çözümlerde genel olarak kural tabanlı yaklaşımlar, parmak izi (fingerprinting) eşleştirme ve kurallı ifadeler (regular expression) benzeri yöntemler kullanan algoritmalar yer almaktadır. Sultan Alneyadi vd. [1] veri sızıntısı ile ilgili yaptıkları çalışmada, parmak izi ve kurallı ifadelerin anlamsal analizde başarısız olduğunu bu yüzden istatistiksel yöntemlerle bu başarısızlığın giderilebildiğini ifade etmektedir. Aynı çalışmada, eş anlamlı kelimeler kullanma, kelime ekleme, kelime çıkarma gibi veri değişikliklerine karşı k-skip-n-gram (skip-gram) kullanımının başarılı olduğunu belirtmişlerdir. Bu yöntemde verilerin indekslenmesi gerekmektedir. Bu da ekstra alan gerektirdiğini belirtmektedir. Bu problemi aşmak için ise skip-gramlar yerine, n-gramların ters metin frekansının (TD-IDF) kullanılmasının uygun olabileceğini belirtmektedir. Çalışmamızda bu yaklaşıma uygun olarak istatistiksel bir yöntem tercih edilmiştir.

Michael Hart vd. [5] veri sızıntısı ile ilgili yaptıkları çalışmada mevcut çözümlerin, bilgiyi açığa çıkarılması açısından tamamen gelişmiş olmamakla beraber hassas verinin yakalanması açısından da eksiklikleri olduğunu belirtmektedir.

Yavuz Canbay vd. [6] yaptığı çalışmada hassas veriler üzerinde yapılan modifikasyonları tespit etmek için bir model önermiştir. Bu modelde hassas veriler LSI yöntemiyle hassas olarak sınıflandırıldığında hassas veri içerip içermediği Smith Waterman ve Boyer Moore algoritmalarının ardışık uygulanmasıyla bulunabileceği belirtilmiştir. Burada ele alınan yöntemler önemli olmakla beraber saldırı olması durumda algoritmanın ilk aşamasında uygun n-gramların çıkarılamaması sonucu doğrudan etkileyecektir. Burada dokümanın öncelikle doğru sınıflandırılması sağlanılmalıdır. Ayrıca hassas verinin içinde bulunduğu dokümanın tamamına yönelik yapılacak bir saldırıda yine bu yöntem zayıf kalmaktadır. Sınıflandırmada karşılaşılan bir diğer sorun, metnin içerisindeki hatalı yazılmış sözcüklerin olması durumudur. Bu durumda sözcüklerin düzeltilmesi gerekir. Bruna Martins ve Mario J. Silva [7] yazım hatalarını düzeltme ile ilgili yaptıkları çalışmada yazım hatalarının oluştuğu durumları incelemiştir. Ancak ilgili çalışmada belirtilen yazım hataları dışında da çeşitli şekillerde yazım hataları olabilmektedir. Farag Ahmed vd. [8] kelime düzeltme ile ilgili yaptıkları çalışmada, dilden bağımsız bir düzeltme modeli önermektedir. Bu çalışmada n-gram tabanlı bir karşılaştırma model kullanılmaktadır. Bu yöntemde ayrıca kelime düzeltme işlemi uygulanmaktadır. Ancak bu işlem yapılırken kelimenin geçme sıklığı baz alınmaktadır. Kelime geçme sıklığına göre yazım düzeltme işlemi çokça kullanılmakta ve efektif olduğu görülmektedir [9]. Bu çalışmalardaki öneriler dikkate alınarak, çalışmamızda kullanılan özellik çıkarımı yöntemleri, yazım düzeltme algoritmalarıyla desteklenmiştir.

Dokümanda karakter gramların kullanılması kelime gramlarına göre kelime deseninin bulunması, yazım hatalarının giderilmesi ve doküman yazım dilinin tespitinde daha efektif olmaktadır. Ayrıca kelime gram kullanımı sonucunda seyrek (sparse) vektörlerin oluşması problemi ortaya çıkabilmektedir [10]. Doküman hakkında kelime gramlarıyla bilgi elde edemediğimiz durumlarda karakter-gramlar efektif olabilmektedir. Özellikle n-gramlara ayırma algoritmalarını atlatmaya yönelik yapılan saldırılarda karakter gramların kullanılması gerekmektedir. Ayrıca dokümana ait yeterli bilgi alınamayan kısa metinlerde kelime skip-gram ve karakter gramların beraber kullanılması dokümana ait daha fazla bilgi elde edilmesi sağlanmaktadır. Bu sebeple, özellikle n-gramların çıkarımının uygun olamadığı saldırılar sonucunda sınıflandırma başarımını korumak için karakter-gramların da kullanımı tercih edilmiştir.

LSA yöntemi, kelimeler arasındaki anlamsal bağlantıyı bulmak için kullanılan bir yöntemdir. LSA, bir Veri seti (corpus) kullanarak veriyi öğrenmektedir. Veriyi öğrenme aşamasında Kırpılmış SVD (Truncated SVD) kullanarak boyut azaltımı yapılmakta daha sonra uzaklık ölçümüne göre kelimeler arasındaki bağ bulunmaktadır [11]. LSA yöntemi özellikle karartılmış verinin üzerinde kullanılması ile kelimelerin anlamı belirginleştirilmektedir. Eş anlamlı ve çok anlamlı kelime saldırılarına karşı LSA özellik çıkarım yöntemi önem arz etmektedir. Bu sebeple, Özellik Çıkarımı aşamasında LSA'dan da faydalanılmıştır.

3. YAKLAŞIM (PROPOSED METHOD)

Bu bölümde, önerilen sistemin tanıtımı yapılmaktadır. Önerilen sistem, metin önişleme, özellik çıkarımı, makine öğrenimi ve yapay sinir ağları yöntemlerini içermektedir (Şekil 1, Şekil 2 ve Şekil 3) Önerilen algoritmanın eğitim aşamasında öncelikle metin üzerinde önişlemler yapılmış, daha sonrasında özellik çıkarımı sağlanmış, çıkarılan özellikler arasından özellik seçimi yapılarak sınıflandırma için daha az gürültülü bir özellik kümesi çıkarılmıştır. Seçilen özellikler sınıflandırıcılardan geçirilerek "Sınıflandırma Modeli" oluşmaktadır. Test aşamasında ise dokümanlar üzerinde saldırılar yapılmakta, daha sonrasında ise sırasıyla önişleme, özelik çıkarımı, özellik seçimi aşamalarından geçirilmekte; en son ise eğitim aşamasında çıkarılan Sınıflandırma Modeli kullanılarak sınıflandırma kararı verilmektedir (Şekil 1). Bu adımlar sonraki

bölümlerde detaylandırılmıştır. Hem eğitim hem test aşamalarında 4 farklı yöntem uygulanarak Yazım Düzeltimi işlemi farklı adımlarda uygulanmaktadır. Bu yolla Yazım Düzeltiminin hangi aşamada yapıldığında daha başarılı olacağı test edilmektedir.

3.1. Metin Önişleme (Text Preprocessing)

Bu aşamada verilen dokümandaki metinlerdeki harfler küçük karaktere dönüştürülmekte; daha sonra sayısal veriler, noktalama işaretleri ve fonksiyonel kelimeler metinden çıkarılmaktadır (Şekil 2). NLP aşamalarında genel olarak uygulanan bu yaklaşım ile hem sınıflandırma başarımı arttırılmakta hem de boyut azaltımı (dimensionality reduction) ile veri miktarı düşürülmektedir.

Algoritmanın 2. Versiyonunda önişlem adımından önce Yazım Düzeltimi yapılmaktadır.

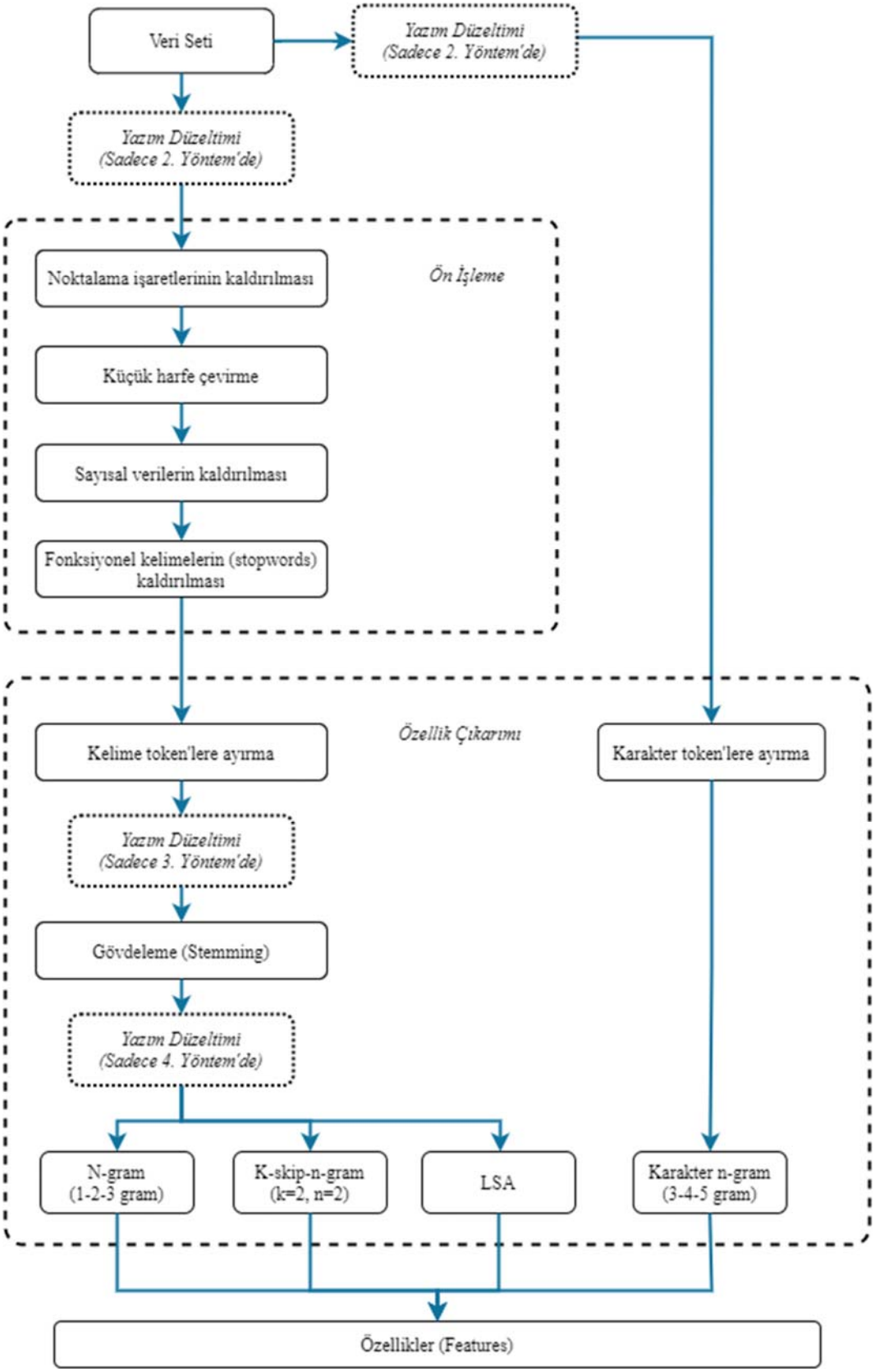
3.2. Özellik Çıkarımı (Feature Extraction)

Bu aşamada öncelikle doküman hem kelime hem de karakter token'lere ayrılmaktadır. Kelime token çıkarımı sonrasında özellik sayısının düşürülmesi için, gövdeleme (stemming) yapılarak kelimeler eklerinden arındırılmaktadır. Algoritmanın 3. ve 4. versiyonlarında gövdeleme öncesi ve sonrası Yazım Düzeltimi yapılmaktadır. Elde edilen kelime token'larından ile farklı boyutlarda n-gram ve skip-gram'lar çıkarılmakta, ayrıca SVD dönüşümü yapılarak LSA özellikleri de çıkarılmaktadır. Karakter-gram'lar ile de benzer şekilde farklı boyutlarda n-gram'lar oluşturulmaktadır.

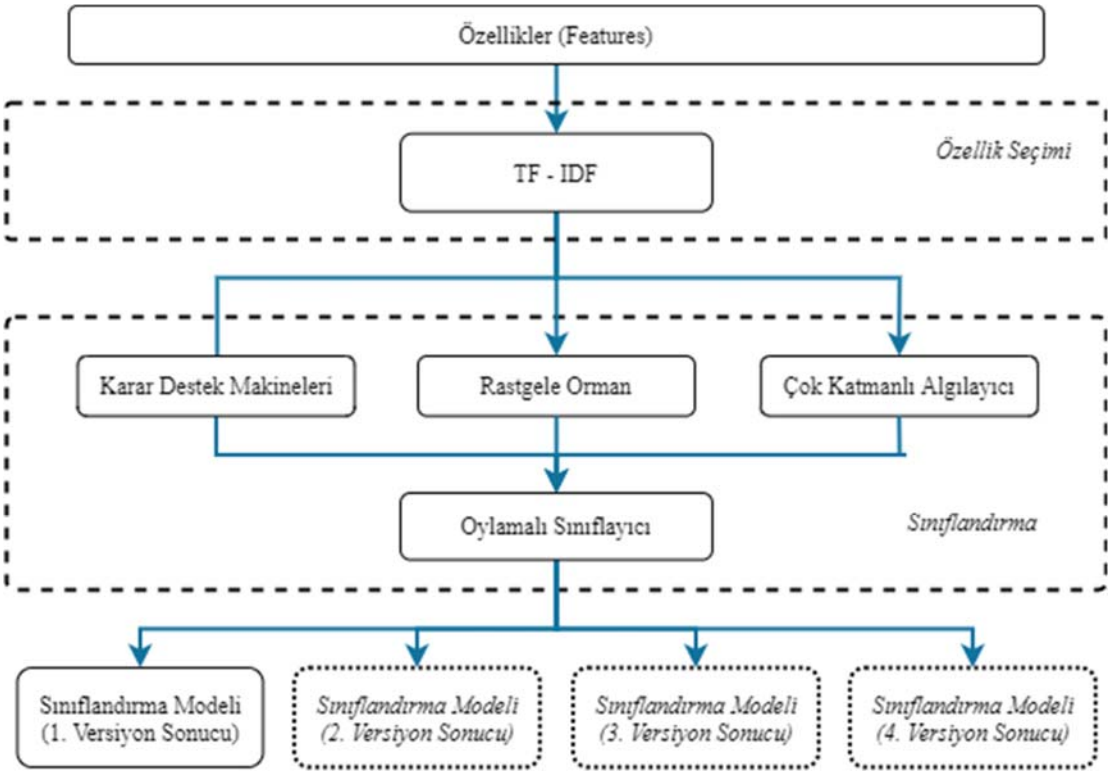
Algoritma:

```
ozellik_cikarimi_secimi( metin ) :

   if model_tipi == VERSION_2:
      metin = yazim_duzelt( metin )

   char_token = karakter_tokenlerine_ayir( metin )
   metin = noktalama_isaretlerini_kaldir( metin )
   metin = kucuk_harfe_cevir( metin )
   metin = sayisal_verileri_kaldir( metin )
   metin = fonksiyonel_kelimeleri_kaldir( metin )

   kelimeler = metini_kelimelerine_ayir( metin )
   if model_tipi == VERSION_3:
      kelimeler = yazim_duzelt( kelimeler )

   kelimeler_kok = kelimeleri_govdeye_ayir( kelimeler )
   if model_tipi == VERSION_4:
      kelime_kok = yazim_duzelt( kelimeler_kok )

   n_gram = n_gramlarina_ayir( kelimeler_kok )
   skip_gram = skip_grama_ayir( kelimeler_kok )
   lsa_model = lsa( kelimeler_kok )

   ozellikler = ozellikleri_birlestir(
                  n_gram,
                  skip_gram,
                  lsa_model,
                  char_token )

   secilmis_ozellikler = tf_idf_ozellik_secimi( ozellikler )
   return secilmis_ozellikler

metin_egitim = egitim_metinlerini_getir()
ozellikler = ozellik_cikarimi_secimi( metin_egitim )
siniflandirici_modeli = model_cikarimi_yap( ozellikler )
metin_test = test_metinlerini_getir()
saldirilmis_metin = saldiri_yap( metin_test )
ozellikler_test=ozellik_cikarimi_secimi(saldirilmis_metin)
sonuc = siniflandirici_modeli.tahmin( ozellikler_test )
```

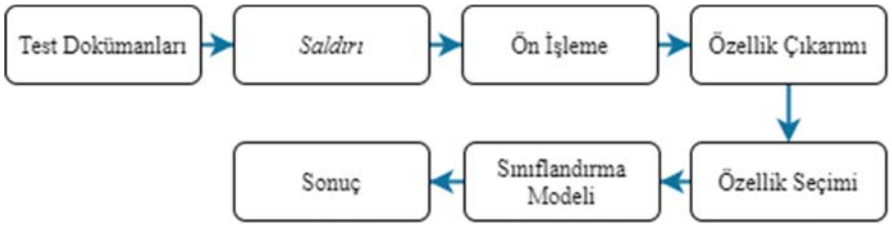
Şekil 1. Önerilen yaklaşıma ait sözde kod (Pseudo code for the proposed method)



Şekil 2. Önerilen yaklaşım - Özellik Çıkarımı (Proposed method - Feature Extraction)



Şekil 3. Önerilen yaklaşım - Sınıflandırma (Proposed method - Classification)



Şekil 4. Test aşamaları (Test stages)

Farklı saldırı türlerine karşı başarılı bir algoritma olabilmesi amacıyla Özellik Çıkarımı aşamasında n-gram, skip-gram, LSA ve karakter-gram çıkarımları bir arada kullanılmıştır. Temel olarak n-gram bazlı olan algoritmanın başarımını sağlamak için modifikasyon saldırılarına karşı karakter-gram, eş anlamlı kelimelerle değiştirme ve karartma saldırılarına karşı LSA, yer değiştirme ve yerine koyma saldırılarına karşı ise skip-gram kullanımı tercih edilmiştir.

3.3. Özellik Seçimi-Terim Ağırlıklandırma
(Feature Selection-Term Weighting)

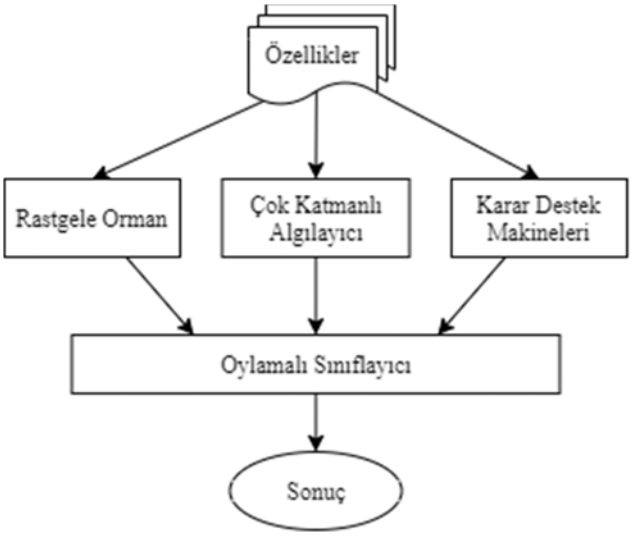
Bir kelimenin bir doküman içerisindeki geçme sıklığına o terimin frekansı denir. Bir kelimenin frekansı o dokümanın hangi sınıfa ait olduğunu gösterebilecek önemli bir kanıttır. Bir kelimenin bir dokümanda fazla geçmesi kelimenin ayrım gücünü düşürdüğü için bu kelimenin sınıflandırmaya etkisini düşürmekte fayda vardır. Ayrıca doküman içinde az geçen bir kelimenin dokümanı sınıflandırmaya etkisi olmadığı düşüncesi yanlıştır. Burada dokümanda az görülen kelimeler ile ilgili bir ölçüyü ters doküman frekansı (Inverse Document Frequency IDF) ile bulunabilir ve dokümanda sık geçen kelimelerin etkisini düşürmek için terim frekansı (Term Frequency-TF) ile çarpılarak TF-IDF değeri o kelimenin frekansı olarak alınabilir. Bu şekilde dokümanda sık geçen kelimeler ile az geçen kelimelerin dokümanı temsil etme ağırlığı eşitlenebilir. Yaptığımız çalışmada n-gramlara TF-IDF yöntemi uygulanmıştır ve terim frekansı 3'ten küçük olan terimler (n-gramlar ve LSA için kelimeler) elimine edilerek özellik sayısının azaltılması sağlanmıştır.

3.4. Sınıflandırma (Classification)

Sınıflandırma aşamasında farklı saldırılar neticesinde çıkarılan özelliklerde oluşan değişimlerin sınıflandırmayı en az bir şekilde etkilemesi için çoklu sınıflandırıcı kullanılmıştır [12].

Literatürde başarımları kanıtlanmış olan, Karar Destek Makineleri, Rasgele Orman ve Çok Katmanlı Algılayıcıdan elde edilen sonuçlar Oylamalı Sınıflandırıcı'ya aktarılarak en çok oy alan sınıf seçilmektedir. (Şekil 5).

Farklı saldırı türleri ve farklı tür dokümanlarda sınıflandırma başarımını korumak için sınıflandırıcıların oylamalı olarak kullanılması tercih edilmiştir.



Şekil 5. Oylamalı sınıflandırıcı (Vote classifier)

4. TESTLER (TESTS)

4.1. Test Veri Kümeleri (Test Datasets)

Testlerde, [5] çalışmasında da kullanılmış olan Transcendental Meditation (TM), Mormon ve Dyncorp veri setleri kullanılmıştır. Bu veri setleri, Gizli (G) ve Kurumsal Genel (KG) olarak etiketlenmiştir. Ayrıca DBPedia tarafından sağlanan veri setleri de Kurumsal Olmayan (KO) etiketine sahip olacak şekilde kullanılmıştır (Tablo 1). Gerçek bir kurumda oluşabilecek veri yapısını daha iyi benzetmek için veri setleri, birleştirilmiştir. Daha sonrasında ise %70 eğitim %30 test olacak şekilde ikiye ayrılmıştır (Tablo 2). Veri setlerinin orijinal hallerinde KG ve KO etiketli test verilerinin G etiketli olanlara göre sayıca çok daha fazla olduğu için algoritmanın aşırı uyuma gitmemesi için test dokümanlarının sayısı eşitlenmeye çalışılmıştır.

Tablo 1. Kaynak veri setleri (Datasets)

Etiket	TM	Mormon	DynCorp	DBPedia
G	85	593	2	-
KG	120	2541	198	-
KO	-	-	-	2000

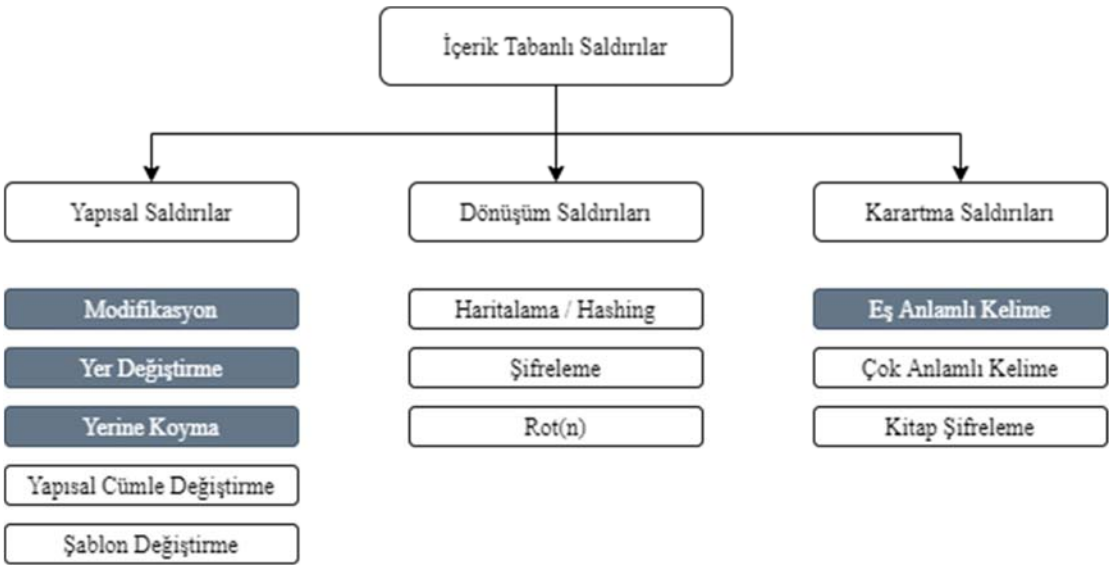
Tablo 2. Testlerde Kullanılan doküman sayıları ve etiketleri (Number of documents used in test and labels)

Etiket	Eğitim	Test	Toplam
G	466	214	680
KG	2023	836	2859
KO	1387	613	2000

4.2. Testlerin Gerçekleştirilmesi ve Sonuçlar (Execution of the Tests and Results)

Test dokümanlarına, öncelikle Şekil 4'teki saldırı adımı hariç diğer adımlar uygulanarak sınıflandırma yapılmış ve algoritmanın normal başarısı ölçülmüştür. Daha sonra da Tablo 3'teki saldırılar sırasıyla uygulanarak her bir saldırıya ait f1-skor değerleri elde edilmiştir. Saldırılar, [6] ve [3] çalışmalarındaki saldırı türleri birleştirilerek ve türlerine göre yeniden sınıflandırılarak oluşturulmuştur (Şekil 6). Bu çalışmada uygulanan saldırılar şekilde farklı olarak gösterilmiştir. Testlerde, bu saldırı tiplerine uygun saldırı vektörleri kullanılmıştır.

Yazım Düzeltimi işleminin farklı aşamalarda uygulanmasının sonuca etkisini görebilmek için Şekil 2 ve Şekil 3'te gösterildiği gibi 4 farklı yöntem ile sınıflandırma yapılmıştır. Buna göre 1. Yöntemde yazım düzeltimi yapılmazken, 2. Yöntemde Önişlem adımından önce, 3. Yöntemde Gövdeleme adımından önce, 4. Yöntemde ise Gövdeleme adımından sonra yazım düzeltimi yapılmıştır.



Şekil 6. İçerik tabanlı saldırılar (Content-based attacks)

Tablo 3. Dokümanlara yapılan saldırılar (Executed attacks on documents)

Saldırı Türü			Saldırı Kodu
Saldırı yok			S0
Yapısal Saldırılar	Modifikasyon -Kelimeyi değiştirme	Dokümandaki kelimelerin sonuna harf ekleme	S1
		Tüm metindeki kelimelerin başına rastgele harf ekleme	S2
		Tüm metindeki kelimelerin ortasına rastgele harf ekleme	S3
		Tüm metindeki boşlukları kaldırma	S4
		Tüm metindeki boşluklar yerine rasgele harf konulması	S5
		Tüm metindeki boşluklar yerine + konulması	S6
		Tüm metindeki kelimelerden rasgele bir harf çıkarılması	S7
		Tüm metindeki E ve e harfi yerine 1 sayısı konulması	S8
	Yer değiştirme	Paragrafların yerini değiştirme	S9
		Kelimelerin yerini değiştirme	S10
	Yerine koyma	Summarizer ile özet	S11
		LSASumarizer ile özet	S12
		LEXMark ile özet	S13
		TEXRank ile özet	S14
		SUMBasic ile özet	S15
		KLSummarizer ile özet	S16
		ReductionSummarizer ile özet	S17
Karartma Saldırıları		Eş anlamlı kelimeler ile değiştirme	S18

Algoritmanın başarımını karşılaştırmak için n-gram kullanan Kategori Profilleri [13] ve SGD (Stochastic Gradient Descent) [14] algoritmaları ile bu alanda yaygın olarak kullanılan çalışmada Evrişim Derin Öğrenme (Convolutional Neural Network-CNN) [2] algoritması ile aynı testler koşturulmuştur. Kategori Profilleri yönteminde eğitim dokümanına ait karakter n-gramlar 1 ile 5 arasında çıkarılmaktadır. Çıkarılan bu n-gramlar daha sonra terim frekansına göre büyükten küçüğe sıralandıktan sonra belirlenen sayıda ilk n-gram alınarak o dokümana ait kategori profili bulunmaktadır. Daha sonra yeni veriye aynı süreçler uygulanarak [15]'te belirtildiği gibi kategoriler arası mesafe ölçülmekte ve bu ölçü sonucunda bulunan en küçük dokümanın etiketi bu yeni dokümanın etiketi olarak atanmaktadır. Bu yöntemi [13] yaptığı veri sızıntısı önleme sisteminde uygulayıp çıkarılan kategori profillerinden en başarılı sonucun ilk 300 değerini alarak elde ettiğini belirtmektedir.

Veri sızıntısı önleme (DLP) sistemlerinde kullanılan N-gramların çeşitli saldırılar altında sınıflama başarısının düştüğü yapılan çalışmalarda belirtilmiştir. N-gramların etkisini ölçmek için [16] ile belirtilen N-gram bazlı çalışma ele alındı. Bu çalışmada SGD (Stochastic Gradient Descent) sınıflandırıcı ile 2-gram (Unigram) yönteminin başarılı olduğu belirtilmiştir. SGD sınıflandırıcı, Karar destek makineleri (SVM) ve Lojistik regresyon (LR) gibi dışbükey hata fonksiyonları altında dahi etkili ayırım yapabilen bir sınıflama yaklaşımıdır. Yüksek verinin çıkabileceği öngörülen modellerde kullanılmaktadır. SGD yönteminde gradient değerini hesaplamak yerine her iterasyonda rasgele seçilen örnekler üzerindeki gradient değerleri alınmaktadır. Bu şekilde riski en aza indirmeye çalışmaktadır.

Günümüzde derin öğrenme yöntemleri birçok alanda kullanılmaktadır. Başarısı yadsınamayan bu yöntemin DLP sistemlerinde doküman sınıflamada kullanılmasının gerçekleştirilen saldırılar altında başarısı için [2] çalışmasındaki yöntem kullanıldı. Bu çalışmada CNN yöntemi ile doküman sınıflandırma yapılmaktadır. CNN ileri beslemeli bir derin öğrenme yöntemidir. Genel olarak bilgisayar görmesi alanında çokça kullanılmasına rağmen son zamanlarda NLP alanında da kullanılmaktadır. Çalışma şekli olarak matrisler aracılığıyla desenler çıkararak, bulduğu desenlerden öğrenerek ilerlemektedir. Farklı saldırı durumlarında önerdiğimiz algoritmanın sınıflandırma sonuçlarına ilişkin metrik değerleri Tablo 4'te gösterilmiştir. Benzer şekilde Kategori Profilleri ve SGD ile yapılan testlerin sonuçları, Yöntem 2 ile beraber Tablo 5'te gösterilmiştir. CNN yöntemi hassaslık ölçüsü (accuracy) değeri sağladığı için Yöntem 2'nin sonuçları ile karşılaştırılması Tablo 6'da gösterilmiştir.

Tablo 4. Saldırı yöntemi bazında test sonuçları (Test results per attack method)

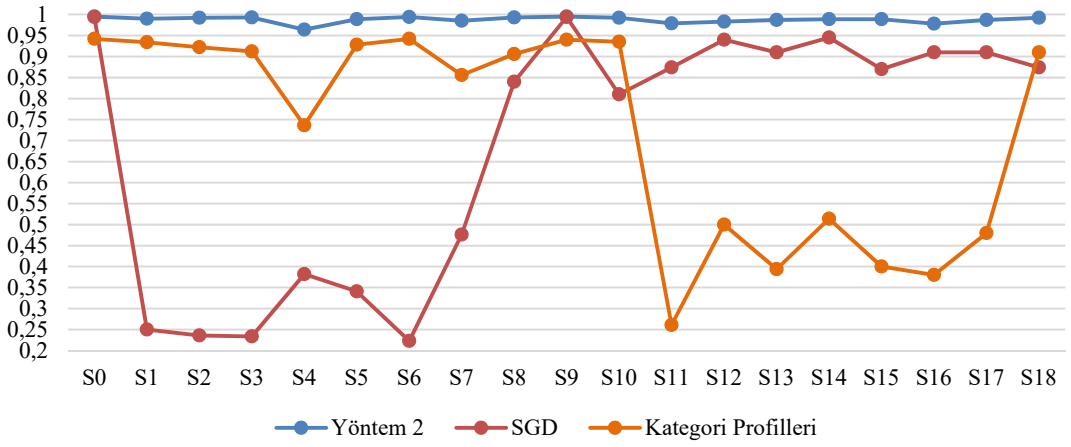
		Yöntem 1			Yöntem 2			Yöntem 3			Yöntem 4		
		G	*KG*	*KO*	*G*	*KG*	*KO*	*G*	*KG*	*KO*	*G*	*KG*	*KO*
Saldırı Yok S0	G	203	0	0	205	0	0	205	0	0	196	0	0
	KG	11	836	0	9	0	0	9	836	0	8	836	0
	KO	0	0	612	0	836	612	0	0	612	0	0	612
	F-skor	0,97	0,99	1,00	0,98	0,99	1,00	0,98	0,99	1,00	0,98	1,00	1,00
	F-sk.(μ)	0,99			0,99			0,99			0,99		
		G	*KG*	*KO*	*G*	*KG*	*KO*	*G*	*KG*	*KO*	*G*	*KG*	*KO*
Dokümandaki kelimelerin sonuna harf ekleme S1	G	171	0	0	201	0	0	187	0	0	185	0	0
	KG	41	832	0	130	834	0	27	835	0	28	825	0
	KO	2	4	612	0	2	612	0	1	612	1	1	612
	F-skor	0,89	0,97	1,00	0,76	0,93	1,00	0,93	0,98	1,00	0,86	1,00	1,00
	F-sk.(μ)	0,97			0,99			0,98			0,98		
		G	*KG*	*KO*	*G*	*KG*	*KO*	*G*	*KG*	*KO*	*G*	*KG*	*KO*
Tüm metindeki kelimelerin başına rasgele harf ekleme S2	G	167	0	0	203	0	0	181	1	0	177	0	0
	KG	39	832	0	11	836	0	31	834	0	34	833	0
	KO	8	4	612	0	0	612	2	1	612	3	3	612
	F-skor	0,88	0,97	0,99	0,97	0,99	1,00	0,91	0,98	1,00	0,91	0,98	1,00
	F-sk(μ)	0,97			0,99			0,98			0,98		
		G	*KG*	*KO*	*G*	*KG*	*KO*	*G*	*KG*	*KO*	*G*	*KG*	*KO*
Tüm metindeki kelimelerin ortasına rastgele harf ekleme S3	G	116	0	0	201	0	0	192	0	0	198	0	0
	KG	35	810	0	12	835	0	21	835	0	15	835	0
	KO	63	26	612	0	1	612	1	1	612	1	1	612
	F-skor	0,70	0,96	0,93	0,97	0,99	1,00	0,95	0,99	1,0	0,96	0,99	1,00
	F-sk(μ)	0,92			0,99			0,98			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki boşlukları kaldırma S4	G	187	0	0	185	1	0	184	0	0	182	0	0
	KG	16	812	0	20	805	0	18	799	0	20	806	0
	KO	11	24	612	9	30	612	12	37	612	12	30	612
	F-skor	0,93	0,98	0,97	0,97	0,99	1,00	0,92	0,97	0,96	0,92	0,97	0,97
	F-sk(μ)	0,97			0,99			0,96			0,96		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki boşluklar yerine rasgele harf konulması S5	G	184	0	0	197	0	0	194	0	0	198	0	0
	KG	16	812	0	17	835	0	20	834	0	15	834	0
	KO	11	24	612	0	1	612	0	2	612	1	2	612
	F-skor	0,92	0,98	0,99	0,96	0,99	1,00	0,95	0,99	1,00	0,96	0,99	1,00
	F-sk(μ)	0,98			0,99			0,99			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki boşluklar yerine + konulması S6	G	197	4	0	205	0	0	199	3	0	195	1	0
	KG	11	820	1	9	836	0	10	812	0	15	814	0
	KO	9	12	611	0	0	612	5	21	612	6	21	612
	F-skor	0,94	0,98	0,98	0,98	0,99	1,00	0,96	0,98	0,98	0,95	0,98	0,98
	F-sk (μ)	0,98			0,99			0,98			0,97		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki kelimelerden rasgele bir harf çıkarılması S7	G	158	0	0	191	0	0	176	0	0	174	0	0
	KG	48	832	0	23	835	1	37	834	1	39	834	0
	KO	8	4	612	0	1	611	1	2	611	1	2	612
	F-skor	0,85	0,97	0,99	0,94	0,99	1,00	0,90	0,98	1,00	0,90	0,98	1,00
	F-sk(μ)	0,96			0,99			0,97			0,97		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki E ve e harfi yerine 1 sayısı konulması S8	G	140	0	0	203	0	0	162	0	0	143	0	0
	KG	72	834	0	11	836	0	52	836	0	71	836	0
	KO	2	2	612	0	0	612	0	0	612	0	0	612
	F-skor	0,79	0,96	1,00	0,97	0,99	1,00	0,86	0,97	1,00	0,80	0,96	1,00
	F-sk.(μ)	0,95			0,99			0,97			0,95		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
Paragrafların yerini değiştirme S9	G	203	0	0	205	0	0	205	0	0	206	0	0
	KG	11	835	0	9	836	0	9	836	0	8	836	0
	KO	0	1	612	0	0	612	0	0	612	0	0	612
	F-skor	0,97	0,99	1,00	0,98	0,99	1,00	0,98	0,99	1,00	0,98	1,00	1,00
	F-sk.(μ)	0,99			0,99			0,99			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
Kelimelerin yerini değiştirme S10	G	202	0	0	202	0	0	205	0	0	205	0	0
	KG	12	835	0	12	836	1	9	836	0	9	836	0
	KO	0	1	612	0	0	611	0	0	612	0	0	612
	F-skor	0,97	0,99	1,00	0,97	0,99	1,00	0,98	0,99	1,00	0,98	0,99	1,00
	F-sk.(μ)	0,99			0,99			0,99			0,99		

		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
Summarizer ile özetleme S11	G	194	4	0	199	6	0	197	6	0	196	7	0
	KG	11	820	1	8	818	1	8	817	1	9	816	1
	KO	9	12	612	7	12	611	9	13	611	9	13	611
	F-skor	0,94	0,98	0,98	0,95	0,98	0,98	0,94	0,98	0,98	0,94	0,98	0,98
	F-sk.(µ)	0,98			0,98			0,98			0,98		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
LSASummarize ile özetleme S12	G	190	0	0	202	3	0	195	0	0	192	0	0
	KG	22	830	1	8	826	0	16	836	1	20	836	0
	KO	2	6	611	4	7	612	3	0	611	2	0	612
	F-skor	0,94	0,98	0,99	0,94	0,99	0,99	0,95	0,99	1,00	0,95	0,99	1,00
	F-sk.(µ)	0,98			0,98			0,99			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
LEXMark ile özetleme S13	G	199	0	0	200	0	0	198	3	0	198	1	0
	KG	13	830	0	9	832	0	13	827	0	14	832	0
	KO	2	6	612	0	4	612	3	6	612	2	3	612
	F-skor	0,96	0,99	0,99	0,96	0,99	0,99	0,95	0,99	0,99	0,96	0,99	1,00
	F-sk.(µ)	0,99			0,99			0,98			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
TEXMark ile özetleme S14	G	199	1	0	191	1	0	200	0	0	198	1	0
	KG	15	834	0	20	831	0	14	832	0	16	834	0
	KO	0	1	612	3	4	612	0	4	612	0	1	612
	F-skor	0,96	0,99	1,00	0,98	0,99	1,00	0,97	0,99	1,00	0,96	0,99	1,00
	F-sk.(µ)	0,99			0,99			0,98			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
SUMBasic ile özetleme S15	G	207	1	0	206	0	0	208	1	0	206	0	0
	KG	5	827	0	4	826	0	5	826	0	5	828	0
	KO	2	0	612	4	10	612	1	9	612	3	829	612
	F-skor	0,98	1,00	1,00	0,98	0,99	0,99	0,98	0,99	0,99	0,98	0,99	0,99
	F-sk.(µ)	0,99			0,99			0,99			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
KLSummarizer ile özetleme S16	G	195	0	0	194	1	0	190	2	0	192	1	0
	KG	17	830	1	15	820	1	21	828	1	20	829	1
	KO	2	6	611	5	15	611	3	6	611	2	6	611
	F-skor	0,95	0,99	0,99	0,95	0,98	0,98	0,94	0,98	0,99	0,94	0,98	0,99
	F-sk.(µ)	0,98			0,98			0,99			0,98		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
ReductionSummarize ile özetleme S17	G	197	3	0	201	3	0	198	3	0	198	1	1
	KG	16	831	1	10	829	1	15	829	1	15	1	1
	KO	1	2	611	3	4	611	1	4	611	1	832	610
	F-skor	0,95	0,99	1,00	0,96	0,99	1,00	0,95	0,99	1,00	0,96	0,99	1,0
	F-sk.(µ)	0,99			0,99			0,99			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO	G	KG	KO
Eş anlamlı kelime S18	G	200	0	0	203	0	0	204	0	0	207	0	0
	KG	14	835	0	11	834	0	10	834	0	7	834	0
	KO	0	1	612	0	2	612	0	2	612	0	2	612
	F-skor	0,97	0,99	1,00	0,97	0,99	1,00	0,98	0,99	1,00	0,98	0,99	1,00
	F-sk.(µ)	0,99			0,99			0,99			0,99		

Tablo 5. Yöntem 2, Kategori profilleri ve SGD test sonuçları (Test result of method2, category profiles and SGD)

		Yöntem 2			Kategori Profilleri			SGD		
		G	KG	KO	G	KG	KO	G	G	KO
Saldırı yok S0	G	205	0	0	149	20	0	206	0	0
	KG	9	0	0	64	810	1	4	833	0
	KO	0	836	612	1	6	611	4	3	612
	F-skor	0,98	0,99	1,00	0,78	0,95	0,99	0,98	1,00	0,99
	F-skor(μ)	0,99			0,94			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO
Dokümandaki kelimelerin sonuna harf ekleme S1	G	201	0	0	145	0	2	1	3	0
	KG	130	834	0	9	810	8	1	39	1
	KO	0	2	612	60	26	602	212	794	611
	F-skor	0,76	0,93	1,00	0,80	0,97	0,93	0,01	0,09	0,55
	F-skor(μ)	0,99			0,93			0,25		
		G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki kelimelerin başına rasgele harf ekleme S2	G	203	0	0	143	0	0	0	1	0
	KG	11	836	0	1	780	0	0	28	0
	KO	0	0	612	70	56	612	214	807	612
	F-skor	0,97	0,99	1,00	0,80	0,96	0,91	0,00	0,06	0,55
	F-skor(μ)	0,99			0,92			0,23		
		G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki kelimelerin ortasına rastgele harf ekleme S3	G	201	0	0	129	10	0	0	10	0
	KG	12	835	0	10	783	1	1	28	2
	KO	0	1	612	75	43	611	213	808	610
	F-skor	0,97	0,99	1,00	0,73	0,96	0,91	0,00	0,06	0,50
	F-skor(μ)	0,99			0,91			0,234		
		G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki boşlukları kaldırma S4	G	185	1	0	50	0	0	14	0	0
	KG	20	805	0	37	587	0	0	142	0
	KO	9	30	612	127	249	612	200	694	612
	F-skor	0,97	0,99	1,00	0,38	0,80	0,77	0,12	0,29	0,58
	F-skor(μ)	0,99			0,73			0,37		
		G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki boşluklar yerine rasgele harf konulması S5	G	197	0	0	138	1	5	3	1	0
	KG	17	835	0	6	787	5	4	111	0
	KO	0	1	612	60	40	602	207	724	612
	F-skor	0,96	0,99	1,00	0,79	0,97	0,92	0,03	0,23	0,57
	F-skor(μ)	0,99			0,93			0,33		
		G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki boşluklar yerine + konulması S6	G	205	0	0	147	18	0	0	0	0
	KG	9	836	0	60	810	1	0	20	0
	KO	0	0	612	5	8	611	214	816	612
	F-skor	0,98	0,99	1,00	0,78	0,95	0,99	0,38	0,20	0,22
	F-skor(μ)	0,99			0,94			0,22		
		G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki kelimelerden rasgele bir harf çıkarılması S7	G	191	0	0	123	16	0	3	0	0
	KG	23	835	1	11	680	1	0	268	2
	KO	0	1	611	70	140	611	211	568	610
	F-skor	0,94	0,99	1,00	0,72	0,89	0,85	0,03	0,48	0,61
	F-skor(μ)	0,99			0,86			0,48		
		G	KG	KO	G	KG	KO	G	KG	KO
Tüm metindeki E ve e harfi yerine 1 sayısı konulması S8	G	203	0	0	107	1	0	58	0	0
	KG	11	836	0	2	788	1	25	760	2
	KO	0	0	612	95	47	611	131	76	610
	F-skor	0,97	0,99	1,00	0,69	0,97	0,90	0,43	0,94	0,85
	F-skor(μ)	0,99			0,91			0,84		
		G	KG	KO	G	KG	KO	G	KG	KO
Paragrafların yerini değiştirme S9	G	205	0	0	147	27	0	206	0	0
	KG	9	836	0	66	808	1	4	833	0
	KO	0	0	612	1	1	611	4	3	612
	F-skor	0,98	0,99	1,00	0,76	0,94	1,00	0,98	1,00	0,99
	F-skor(μ)	0,99			0,94			0,99		
		G	KG	KO	G	KG	KO	G	KG	KO
Kelimelerin yerini değiştirme S10	G	202	0	0	143	39	0	70	0	0
	KG	12	836	1	70	805	2	2	674	1
	KO	0	0	611	1	2	610	142	162	611
	F-skor	0,97	0,99	1,00	0,72	0,93	1,00	0,49	0,89	0,80
	F-skor(μ)	0,99			0,93			0,81		

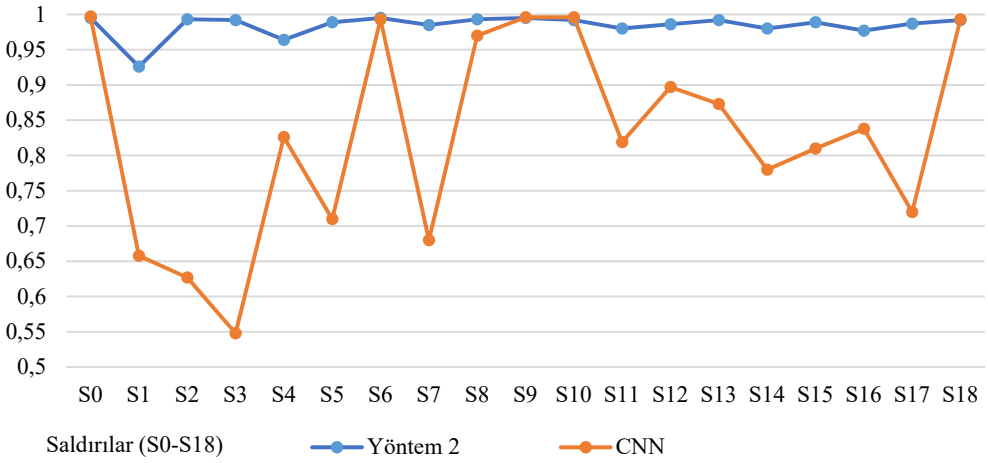
		G	KG	KO	G	KG	KO	G	KG	KO
Summarizer ile özetleme	G	199	6	0	5	1	0	150	2	0
S11	KG	8	818	1	11	47	1	6	685	0
	KO	7	12	611	198	788	612	58	149	612
	F-skor		0,95	0,98	0,98	0,05	0,11 0,55	0,82	0,90	0,86
	F-skor(μ)		0,98			0,26		0,87		
			G	KG	KO	G	KG	KO	G	KG	KO
	G		202	3	0	9	9	0	166	0	0
LSASummarize ile özetleme	KG		8	826	0	19	300	1	5	785	0
S12	KO		4	7	612	186	527	611	43	51	612
	F-skor		0,94	0,99	0,99	0,08	0,52	0,63	0,87	0,97	0,93
	F-skor(μ)		0,98			0,50			0,94		
			G	KG	KO	G	KG	KO	G	KG	KO
	G		200	0	0	12	13	0	161	1	0
LEXMark ile özetleme	KG		9	832	0	12	165	1	6	733	0
S13	KO		0	4	612	190	658	611	47	102	612
	F-skor		0,96	0,99	0,99	0,10	0,33	0,59	0,86	0,93	0,89
	F-skor(μ)		0,99			0,39			0,91		
			G	KG	KO	G	KG	KO	G	KG	KO
	G		191	1	0	18	28	0	185	0	0
TEXMark ile özetleme	KG		20	831	0	28	300	1	6	773	0
S14	KO		3	4	612	168	508	611	23	63	612
	F-skor		0,98	0,99	1,00	0,14	0,52	0,64	0,93	0,96	0,93
	F-skor(μ)		0,99			0,51			0,95		
			G	KG	KO	G	KG	KO	G	KG	KO
	G		206	0	0	5	6	0	143	1	0
SUMBasic ile özetleme	KG		4	826	0	6	136	1	2	684	0
S15	KO		4	10	612	203	694	611	69	151	612
	F-skor		0,98	0,99	0,99	0,04	0,28	0,58	0,80	0,90	0,85
	F-skor(μ)		0,99			0,40			0,87		
			G	KG	KO	G	KG	KO	G	KG	KO
	G		194	1	0	4	10	0	155	0	0
KLSummarizer ile özetleme	KG		15	820	1	18	157	1	6	742	0
S16	KO		5	15	611	192	669	611	53	94	612
	F-skor		0,95	0,98	0,98	0,04	0,31	1,00	0,84	0,94	0,89
	F-skor(μ)		0,98			0,38			0,91		
			G	KG	KO	G	KG	KO	G	KG	KO
	G		201	3	0	13	18	0	173	1	0
ReductionSummarize ile özetleme	KG		10	829	1	27	269	1	5	723	0
S17	KO		3	4	611	174	549	611	36	112	612
	F-skor		0,96	0,99	1,00	0,11	0,47	0,63	0,89	0,92	0,89
	F-skor(μ)		0,99			0,48			0,91		
			G	KG	KO	G	KG	KO	G	KG	KO
	G		203	0	0	130	0	0	193	1	0
Eş anlamlı kelime	KG		11	834	0	2	776	1	3	817	1
S18	KO		0	2	612	82	60	611	18	18	612
	F-skor		0,97	0,99	1,00	0,76	0,96	0,90	0,95	0,99	0,97
	F-skor(μ)		0,99			0,91			0,98		



Şekil 7. Yöntem 2 ile Kategori Profilleri ve SGD'nin sonuçlarını karşılaştırma grafiği
(Comparision graph of method 2 with category profile and SGD)

Tablo 6. Yöntem 2 ve CNN'nin hassaslık (accuracy) değerlerinin karşılaştırılması
(Comparison of accuracy values of method 2 and CNN)

	Yöntem 2	CNN
Saldırı yok - S0	0,995	0,997
Dokümandaki kelimelerin sonuna harf ekleme - S1	0,926	0,658
Tüm metindeki kelimelerin başına rasgele harf ekleme - S2	0,993	0,627
Tüm metindeki kelimelerin ortasına rasgele harf ekleme - S3	0,992	0,548
Tüm metindeki boşlukları kaldırma - S4	0,964	0,826
Tüm metindeki boşluklar yerine rasgele harf konulması - S5	0,989	0,71
Tüm metindeki boşluklar yerine + konulması - S6	0,995	0,993
Tüm metindeki kelimelerden rasgele bir harf çıkarılması - S7	0,985	0,68
Tüm metindeki E ve e harfi yerine 1 sayısı konulması - S8	0,993	0,97
Cümlenin yerini değiştirme - S9	0,995	0,996
Kelimelerin yer değiştirmesi - S10	0,992	0,996
Summarizer ile özet - S11	0,98	0,819
LSASumarizer ile özet - S12	0,986	0,897
LEXMark ile özet - S13	0,992	0,873
TEXRank ile özet - S14	0,98	0,78
SUMBasic ile özet - S15	0,989	0,81
KLSummarizer ile özet - S16	0,977	0,838
ReductionSummarizer ile özet - S17	0,987	0,72
Eş anlamlı kelime saldırıları - S18	0,992	0,993



Şekil 8. Yöntem 2 ile CNN hassaslık değerleri karşılaştırma grafiği
(Comparison graph of accuracy values of method 2 and CNN)

4.3. Test Sonuçlarının Değerlendirilmesi
(Evaluation of the Test Results)

Sınıflandırma sonuçlarının başarımları literatürde sıkça kullanılan f-skor (f1-skor) değerleri hesaplanarak ölçülmüştür. F-skor değeri kesinlik (precision) değeri ile duyarlılık (recall) değerlerinin harmonik ortalamasıdır. Kesinlik değeri pozitif olarak tahmin edilenlerin hangi oranda doğru olarak tespit edildiğini belirlenmektedir. Duyarlılık değeri ise pozitif olarak tahmin etmemiz gereken işlemlerin ne kadarının pozitif olarak tahmin ettiğimizi göstermektedir. F-skor hesaplaması ile bu iki önemli metrik bir arada değerlendirilmiş olmaktadır. F-skor, 0 ile 1 arasında değerler alabilmektedir ve 1'e yakın değerler alması başarılı kabul edilmektedir. Ancak CNN modelimizden f-skor hesaplaması için gerekli çıktılar alınmadığından bu yöntem ile karşılaştırmalar doğruluk (accuracy) değeri üzerinden yapılmıştır. Eğitim ve test veri kümelerimizde dengeli bir dağılım sağlandığı için bu metriğin kullanımında sorun olmayacağı kabul edilmiştir. Testlerde öncelikle saldırı olmadığı durumda ve farklı saldırılar altında yöntemin farklı varyasyonları kullanılarak sınıflandırma yapılmıştır. Bu sonuçları içeren Tablo 4 incelendiğinde, sunduğumuz çözümün saldırılar altında dahi 0,95 değeri üzerinde f-skor değerleri ürettiği ve dokümanın sınıflandırılması açısından başarılı sonuçlar alındığı görülmüştür. Ayrıca Yazım düzeltiminin başarıyı artırdığı söylenebilir. Testler sonucunda Yazım Düzeltimi'nin Yöntem 2'de önerilen şekilde kullanımının nispeten daha başarılı bir sonuç verdiği ortaya çıkmıştır.

Daha sonra, literatürde kullanılan referans yöntemler aynı dokümanlara ve aynı saldırılar altında uygulanarak karşılaştırılmalar yapılmıştır. Tablo 5 ve Şekil 7 incelendiğinde, Kategori Profilleri yönteminin maksatlı veri kaçırma saldırılarında çok başarılı olamadığı görülmüştür. Saldırı olmadığı durumlarda elde edilen 0,95 üzerindeki f-skor değerlerinin saldırılar altında 0,25 seviyelerine düşebildiği gözlemlenmiştir. Bu yöntemin özellikle modifikasyon saldırılarına karşı dayanıksız olduğu; ayrıca yerine koyma saldırılarında özetleme algoritmasının çalışma şekline bağlı olarak normal sınıflama başarısını her zaman koruyamadığı ortaya konmuştur. Bunula beraber, yer değiştirme saldırılarında normal başarımına yakın sonuçlar verdiği belirlenmiştir. Karartma saldırılarında ise kategori profillerinin başarısının düştüğü görülmüştür. Buna karşın sunduğumuz çözümün bu tip saldırılarda da 0,90 üzerinde bir f-skor ürettiği gözlenmiştir.

Yine Tablo 5 ve Şekil 7 incelendiğinde, SGD yönteminin kelime ve paragraf temelli saldırlar karşısında 0,85 civarında f-skor ürettiği görülmektedir. Saldırı olmadığı durumdaki 1'e yakın değeri elde edemese de başarılı sayılacak bir sınıflandırma sergilemiştir. Paragraf yer değiştirme saldırısında başarının düşmemesinin nedeni n-gram'ların genelde korunmasıdır. Çünkü paragraflar yer değişse bile art arda gelen kelimelerin yeri değişmemiştir. Özetleme saldırılarında da sınıflandırma yapmak için yeterli sayıda n-gram'ın korunması başarımı sağlamıştır. Özellik kümesindeki 1-gram ve 2-gramların varlığı bu tür ataklar karşısında başarımı arttırmıştır. Ancak modifikasyon saldırılarında SGD'nin başarılı olamadığı, 0,2 civarında bir f-skor ürettiği görülmüştür. SGD ve Kategori profilleri'nin başarımını gösteren Şekil 7'deki f-skorları incelendiğinde, Kategori profilleri ile SGD'nin farklı atak türlerine göre yaklaşık olarak ters şekilde etkilendiği; harf temelli saldırılarda Kategori Profilleri'nin SGD'ye göre daha başarılı olduğu, kelime temelli saldırılarda ise SGD'nin Kategori Profilleri'ne göre daha başarılı olduğu görülmektedir. Ancak her iki yöntem de sunduğumuz yönteme göre daha düşük başarım sergilemiştir.

Literatürde yer alan bir diğer yöntem olan CNN ile karşılaştırmanın yapıldığı Tablo 6 ve Şekil 8 incelendiğinde, CNN'nin sınıflandırma başarısının genelde çok düşük olmamasına rağmen önerdiğimiz modeldeki Yöntem 2'ye göre daha düşük bir başarım sergilediği görülmektedir. CNN'nin kelimelere harf ekleme türündeki saldırılarda başarımının 0,55 seviyelerine kadar düştüğü, ancak boşluklara yönelik saldırılarda nispeten daha başarılı olduğu görülmektedir. Boşluklar yerine "+" işareti konulması (S6) ve sadece bir harfe yapılan saldırılarda (S8) başarılı olabildiği görülmüştür. Bu sonuçlar ile cümle ve kelime yer değişimi saldırılarında da yüksek başarım göstererek 0,9 üzerinde sonuçlar üretmesi beraber değerlendirildiğinde CNN'nin dokümanlardaki harfler üzerinden sınıflandırma yapmaya çalıştığı ve dokümandaki harflerin korunması nispetinde başarım gösterdiği anlaşılmaktadır. Bu sebeple özetleme saldırılarında başarımı 0,8 seviyelerine düşmekte; kelimelere rasgele harflerin eklenmesi saldırılarında ise başarımın daha da düşerek 0,55 seviyelerine kadar gerilediği gözlenmektedir. Buna karşın önerdiğimiz modelin farklı türde saldırılar altında bile CNN'den daha başarılı olduğu görülmektedir.

Sunulan yöntemin ve özellik çıkarımı ve seçimi algoritmasının sonuçlar ışığında değerlendirilmesi neticesinde, algoritma içerisinde yer alan farklı adımların farklı atak türlerine karşı başarımı sağladığı değerlendirilmiştir. Buna göre,

- Yazım düzeltimi işleminin kelimelerin başına, sonuna harf eklenmesi, kelimelerdeki bazı harflerin değiştirilmesi, boşlukların "+" ile değiştirilmesi gibi saldırılara karşı başarımı arttırdığı
- Yazım düzeltiminin Yöntem 2'de, özellik çıkarımı başlamadan önce yapılmasının çıkarılan özelliklerin kalitesini arttırdığı
- N-gram ve LSA kullanımının kelime bazlı saldırılara karşı başarımı arttırdığı
- K-skip n-gram kullanımı ile özetleme saldırılarına karşı başarımın artırıldığı
- Karakter-gram kullanımı ile boşlukların kaldırılması ya da bir harf ile değiştirilmesi gibi kelime bütünlüğünün bozulduğu saldırılara karşı başarımın arttığı ortaya konulmuştur.

5. SONUÇLAR (CONCLUSIONS)

Yapısal ve karartma saldırılara karşı önerilen yöntemin başarısı, karşılaştırmalı olarak sunulmuştur. Yapılan çalışmalarda önerilen yöntemin, diğer yöntemlere göre saldırılar karşısında dokümanı başarıyla sınıflandırabildiği görülmüştür. Ayrıca modelde herhangi bir indirgeme yöntemi yapılmadığı göz önüne alındığında burada yapılacak özellik indirgeme yöntemi hem algoritmanın daha hızlı çalışmasını hem de daha iyi sınıflama yapmasını sağlayabilir. Ayrıca kullanılan yöntemlerdeki çoğu parametreler standart olarak alınmıştır. Bu yüzden algoritmanın yüksek başarım sağlaması için özellik indirgeme ve seçilim algoritmalarının kullanılması performansı yükseltebilecek bir durumdur. Diğer taraftan tüm saldırılara dayanıklı bir algoritma geliştirmenin zorluğu tekil çözüm yapılması çok da kolay değildir. Eğer yapılan saldırının tespiti yapılabilirse ele alınan algoritma çözümlerinin tümünün uygulanması yerine saldırıya dayanıklı olan metodun uygulanması hem hız hem de başarı açısından iyi olacaktır.

TEŞEKKÜR (ACKNOWLEDGEMENT)

Bu çalışma TÜBİTAK tarafından desteklenen 117E100 numaralı proje kapsamında gerçekleştirilmiştir.

KAYNAKLAR (REFERENCES)

1. Alneyadi S., Sithirasenan E., Muthukkumarasamy V., A survey on data leakage prevention systems, J. Netw. Comput. Appl., 62, 137-152, 2016.

2. Maheshwari A., Report on Text Classification using CNN, RNN & HAN. https://medium.com/jatana/report-on-text-classification-using-cnn-rnn-han-f0e887214d5f. Yayın Tarihi Temmuz 17, 2018. Erişim Tarihi Aralık 31, 2021.

3. Mustafa T., Malicious Data Leak Prevention and Purposeful Evasion Attacks: An approach to Advanced Persistent Threat (APT) management, 2013 Saudi Int. Electron. Commun. Photonics Conf. SIECPC 2013, 1-5, 2013.

4. Tahboub R., Saleh Y., Data leakage/loss prevention systems (DLP), 2014 World Congr. Comput. Appl. Inf. Syst. WCCAIS 2014, 2014.

5. Hart M., Manadhata P., Johnson R., Text Classification for Data Loss Prevention, 18-37, 2011.

6. Canbay Y., Yazici H., Sagiroglu S., A Turkish language based data leakage prevention system, 2017 5th Int. Symp. Digit. Forensic Secur. ISDFS 2017, 2017.

7. Martins B., Silva M.J., Spelling Correction for Search Engine Queries, Adv. Nat. Lang. Process., 372-383, 2004.

8. Ahmed F., Revised n-gram based automatic spelling correction tool to improve retrieval effectiveness, Res. J. Comput. …, 40, 39-48, 2009.

9. Priya M., Kalpana R., Srisupriya T., Hybrid optimization algorithm using N gram based edit distance, Proc. 2017 IEEE Int. Conf. Commun. Signal Process. ICCSP 2017, 2018-Janua, 216-221, 2018.

10. Kulmizev A., et al., The Power of Character N-grams in Native Language Identification, 2018, 382-389, 2018.

11. Altszyler E., Sigman M., Ribeiro S., Slezak D.F., Comparative study of LSA vs Word2vec embeddings in small corpora: a case study in dreams database, 1-14, 2016.

12. Catal C., Nangir M., A sentiment classification model based on multiple classifiers, Appl. Soft Comput. J., 50, 135-141, 2017.

13. Tripathy A., Agrawal A., Rath S., Classification of Sentiment Reviews using N-gram Machine Learning Approach, Expert Syst. Appl., 57, 2016.

14. Ruder S., An overview of gradient descent optimization algorithms, 1-14, 2016.

15. Topaloğlu M., Özel Anlamlı İfade İçeren Verilerde Sızıntı Önleme İçin Bir Mimari Tasarım Ve Gerçekleştirilmesi, 2012.

16. Tripathy A., Agrawal A., Rath S.K., Classification of sentiment reviews using n-gram machine learning approach, Expert Syst. Appl., March, 57, 117-126, 2016.