TÜBİTAK–2209-A ÜNİVERSİTE ÖĞRENCİLERİ ARAŞTIRMA PROJELERİ DESTEĞİ PROGRAMI

Başvuru formunun Arial 9 yazı tipinde, her bir konu başlığı altında verilen açıklamalar göz önünde bulundurularak hazırlanması ve ekler hariç toplam 20 sayfayı geçmemesi beklenir (Alt sınır bulunmamaktadır). Değerlendirme araştırma önerisinin özgün değeri, yöntemi, yönetimi ve yaygın etkisi başlıkları üzerinden yapılacaktır.

ARAŞTIRMA ÖNERİSİ FORMU

2024 Yılı

1. Dönem Başvurusu

2209/A ÜNİVERSİTE ÖĞRENCİLERİ ARAŞTIRMA PROJELERİ DESTEĞİ PROGRAMI
ARAŞTIRMA ÖNERİSİ FORMU

A. GENEL BİLGİLER

Başvuru Sahibinin Adı Soyadı: Lütfü Bedel, Yusuf Güney
Araştırma Önerisinin Başlığı: Siber Tehdit İstihbaratı için Derin Öğrenme ve Doğal Dil işleme Yöntemleri ile Zararlı URL Tespiti
Danışmanın Adı Soyadı: Arş.Gör.Hasibe CANDAN KADEM
Araştırmanın Yürütüleceği Kurum/Kuruluş: Bursa Teknik Üniversitesi, Mühendislik ve Doğa Bilimleri Fakültesi, Bilgisayar Mühendisliği Bölümü

ÖZET

Türkçe özetin araştırma önerisinin (a) özgün değeri, (b) yöntemi, (c) yönetimi ve (d) yaygın etkisi hakkında bilgileri kapsaması beklenir. Bu bölümün en son yazılması önerilir.

Özet

Bu araştırma, siber tehdit istihbaratında zararlı URL tespiti için derin öğrenme ve doğal dil işleme (NLP) tekniklerini entegre eden yenilikçi bir model geliştirmeyi hedeflemektedir. Siber saldırıların giderek karmaşıklaşması, geleneksel tespit yöntemlerinin yetersiz kaldığını gösterirken, daha güçlü analiz yöntemlerine olan ihtiyacı da artırmaktadır. Bu proje, zararlı URL tespitinde geniş veri kaynaklarından yararlanarak derin öğrenme ve NLP'nin güçlü yönlerini birleştirecektir. Derin öğrenme, büyük ve karmaşık veri setlerinden anlamlı kalıpları öğrenme imkânı sunarken; NLP, URL'lerdeki metin tabanlı özellikleri analiz ederek zararlı içeriklerin daha etkin bir şekilde tespit edilmesini sağlayacaktır.

Araştırma, veri toplama, ön işleme, özellik çıkarma, model eğitimi ve performans değerlendirme adımlarından oluşmaktadır. Çeşitli makine öğrenmesi ve derin öğrenme modelleri (LSTM, CNN, karar ağaçları, rastgele ormanlar) bu süreçte test edilecek, doğruluk, hassasiyet ve geri çağırma gibi ölçütlerle değerlendirilecektir. Ayrıca, kullanıcıların zararlı URL'leri kolayca tespit edebilmesi için bir kullanıcı arayüzü tasarımı da yapılacaktır. Bu çalışmanın, siber güvenlik alanında yenilikçi bir katkı sunarak literatürdeki boşlukları doldurması ve ulusal güvenlik ile kritik altyapıların korunmasına katkı sağlaması beklenmektedir.

Anahtar Kelimeler: Siber güvenlik, zararlı URL tespiti, derin öğrenme, doğal dil işleme, siber tehdit istihbaratı

1. ÖZGÜN DEĞER

1.1. Konunun Önemi, Araştırma Önerisinin Özgün Değeri ve Araştırma Sorusu/Hipotezi

Araştırma önerisinde ele alınan konunun kapsamı ve sınırları ile önemi literatürün eleştirel bir değerlendirmesinin yanı sıra nitel veya nicel verilerle açıklanır.

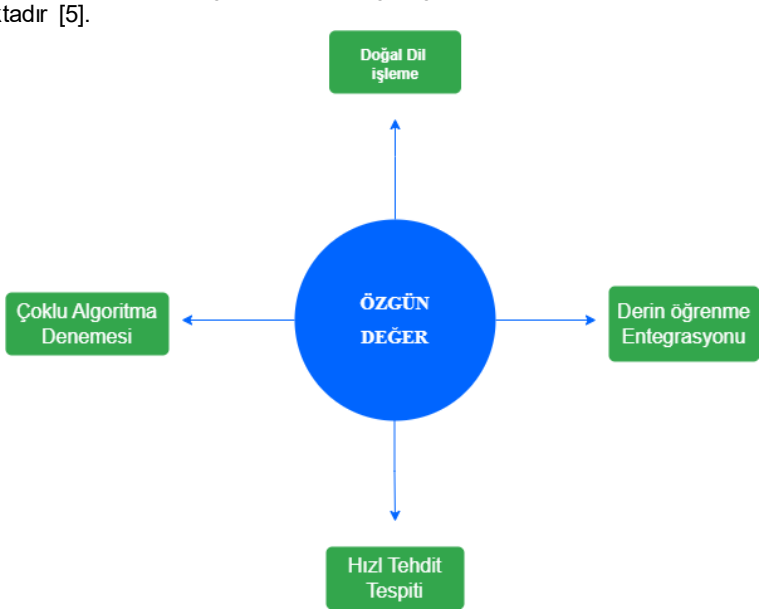
Özgün değer yazılırken araştırma önerisinin bilimsel değeri, farklılığı ve yeniliği, hangi eksikliği nasıl gidereceği veya hangi soruna nasıl bir çözüm geliştireceği ve/veya ilgili bilim veya teknoloji alan(lar)ına kavramsal, kuramsal ve/veya metodolojik olarak ne gibi özgün katkılarda bulunacağı literatüre atıf yapılarak açıklanır.

Önerilen çalışmanın araştırma sorusu ve varsa hipotezi veya ele aldığı problem(ler)i açık bir şekilde ortaya konulur.

Siber güvenlik, dijital çağın en kritik önceliklerinden biri haline gelmiştir. Özellikle zararlı URL'lerin tespiti, siber tehditlerle mücadelede önemli bir rol oynamaktadır. Kötü niyetli kişiler tarafından oluşturulan zararlı URL'lerin hızlı ve etkili bir şekilde tespit edilmesi, bireyler ve kurumlar için hayati öneme sahip bir risk yönetimi stratejisi oluşturmaktadır. Ancak geleneksel tespit yöntemleri, siber saldırı tekniklerinin artan karmaşıklığı ve tehditlerin sürekli değişimi karşısında yetersiz kalmaktadır. Bu nedenle, daha etkili ve yenilikçi tespit yöntemlerinin geliştirilmesi kaçınılmaz hale gelmiştir.

Son yıllarda derin öğrenme ve doğal dil işleme (NLP) teknikleri, zararlı URL tespiti ve siber tehdit analizi alanında umut verici çözümler sunmaktadır. Derin öğrenme, büyük ve karmaşık veri kümelerinden öğrenilen kalıpları etkili bir şekilde analiz edebilme yeteneğine sahiptir. NLP ise metin tabanlı verilerin anlamlandırılmasında önemli bir rol oynar ve zararlı URL'lerin analizinde değerli katkılar sağlar. Literatürde, zararlı URL'leri tespit etmek için birçok çalışma yapılmış olsa da çoğu çalışma yalnızca derin öğrenme veya NLP yöntemine odaklanmaktadır. Bu durum, özellikle karmaşık tehditler ve dinamik URL yapıları gibi derinlemesine analiz gerektiren senaryolarda sınırlayıcı olmaktadır.

Örneğin, derin öğrenme yöntemleri ile zararlı yazılımların tespitinde yüksek doğruluk oranları elde edilmiştir; ancak bu çalışmalarda metin tabanlı analizler sınırlı kalmıştır [1]. Diğer yandan, phishing URL'lerinin tespitinde NLP tabanlı yöntemlerin etkili sonuçlar verdiği gösterilmiş; ancak bu çalışmalarda derin öğrenme entegrasyonu sınırlı kalmıştır [2]. URL karakter dizilimlerini analiz eden evrişimli sinir ağı (CNN) yaklaşımları, zararlı içeriği tespit etmeye odaklansa da daha gelişmiş metin analizi yöntemlerini içermemektedir [3]. Türkiye'de, oltalama saldırılarının tespitinde NLP ve rastgele orman algoritmaları kullanılarak başarı sağlanmış olsa da derin öğrenme yöntemlerinin tüm potansiyeli bu çalışmalarda değerlendirilmemiştir [4]. Yapay sinir ağları ile gerçekleştirilen bir saldırı tespit çalışmasında %99,26'lık başarı oranına ulaşılmışsa da URL tabanlı tehditlerde derinlemesine bir NLP analizi bulunmamaktadır [5].



Şekil 1. Zararlı URL Tespitinde Özgün Değer

Bu araştırmanın özgün değeri, zararlı URL'lerin yol açtığı zararları önlemek amacıyla, derin öğrenme ve NLP tekniklerini bir araya getiren yenilikçi bir model geliştirmektir. Çalışmamızda, farklı algoritmalar deneyerek yüksek performans elde etmek hedeflenmektedir. Literatürdeki çalışmalar genellikle tek bir yönteme odaklanmakta ve sınırlı veri kümeleri üzerinde gerçekleştirilmektedir; çoğu çalışma derin öğrenme algoritmalarını veya NLP tekniklerini bağımsız olarak kullanmaktadır. Bu araştırmada ise iki güçlü teknolojiyi bir araya getirerek, geleneksel yöntemlerin ötesinde bir başarı elde edilmesi amaçlanmaktadır. Önerilen model, geniş veri kaynaklarıyla çalışarak çeşitli tehdit tiplerini etkili bir şekilde tespit edebilecek çok yönlü bir çözüm sunacaktır.

Bu yaklaşım, sadece akademik alanda değil, ulusal güvenlik ve kritik altyapıların korunması gibi uygulama alanlarında da değerlendirilebilecek nitelikte olacaktır. Çalışmamız, siber güvenlik alanında derin öğrenme ve NLP yöntemlerinin entegrasyonunun pratik ve teorik faydalarını ortaya koyarak literatürdeki boşlukları doldurmayı ve siber tehdit istihbaratına önemli katkılar sunmayı hedeflemektedir.

1.2. Amaç ve Hedefler

Araştırma önerisinin amacı ve hedefleri açık, ölçülebilir, gerçekçi ve araştırma süresince ulaşılabilir nitelikte olacak şekilde yazılır.

Bu projenin temel amacı, derin öğrenme ve doğal dil işleme (NLP) yöntemlerini birleştirerek, zararlı URL'lerin daha yüksek doğruluk oranıyla tespit edilmesini sağlayan yenilikçi bir model geliştirmektir. Bu sayede, siber tehdit istihbaratında mevcut yöntemlerin ötesine geçilerek, daha etkin bir güvenlik sistemi oluşturulması hedeflenmektedir.

1. **Zararlı URL tespitinde derin öğrenme ve NLP'nin entegrasyonu:** Projenin birincil hedefi, zararlı URL'lerin tespitinde kullanılan geleneksel yöntemlere kıyasla daha etkili sonuçlar verebilecek entegre bir model oluşturmaktır.

2. **Farklı veri setleriyle geniş kapsamlı analizler yapılması:** Bu hedef, yöntemin farklı veri kaynakları üzerinde test edilerek, genel geçerlilik ve etkinliğinin değerlendirilmesini içerir.

3. **Siber tehditlere karşı daha hızlı ve doğru tespit mekanizmaları geliştirilmesi:** Bu sayede, zararlı URL'ler erkenden tespit edilerek kullanıcıların ve kurumların siber saldırılara karşı daha güçlü bir şekilde korunması sağlanacaktır.

4. **Siber güvenlikte yenilikçi yöntemlerin uygulanabilirliğinin artırılması:** Geliştirilen modelin ulusal güvenlik ve kritik altyapı koruma projelerinde kullanılması, siber güvenlikteki savunma mekanizmalarını güçlendirmeyi hedeflemektedir.

5. **Siber istihbarata katkı sağlanması**: Bu çalışma, siber tehditlerin analiz edilmesi ve önceden tahmin edilmesi noktasında önemli bir adım olacaktır. Bu doğrultuda, siber tehdit istihbaratına zamanında ve yüksek doğrulukta bilgiler sağlayarak, tehditlerin önlenmesine ve karar destek mekanizmalarının iyileştirilmesine katkı sağlaması amaçlanmaktadır.

Bu hedefler, projenin araştırma süresince ulaşılabilir ve ölçülebilir çıktılar üretmesini sağlayacaktır.

2. YÖNTEM

Araştırma önerisinde uygulanacak yöntem ve araştırma teknikleri (veri toplama araçları ve analiz yöntemleri dahil) ilgili literatüre atıf yapılarak açıklanır. Yöntem ve tekniklerin çalışmada öngörülen amaç ve hedeflere ulaşmaya elverişli olduğu ortaya konulur.

Yöntem bölümünün araştırmanın tasarımını, bağımlı ve bağımsız değişkenleri ve istatistiksel yöntemleri kapsaması gerekir. Araştırma önerisinde herhangi bir ön çalışma veya fizibilite yapıldıysa bunların sunulması beklenir. Araştırma önerisinde sunulan yöntemlerin iş paketleri ile ilişkilendirilmesi gerekir.

Bu araştırma, zararlı URL'lerin tespit edilmesi için derin öğrenme ve makine öğrenimi tekniklerini uygulamayı amaçlamaktadır. Çalışma, 5 ana aşamadan oluşmaktadır: veri toplama, veri ön işleme, özellik çıkarma, model seçimi ve eğitimi, Model performansı ve değerlendirme. Aşağıda her bir aşama detaylandırılmıştır:

1. **Veri Toplama**

   Zararlı ve güvenli URL'lerin yer aldığı geniş kapsamlı ve etiketli bir veri seti oluşturulacaktır. Veri toplama kaynakları şunlardır:

   - VirusTotal API: Zararlı URL'ler hakkında bilgi sağlayan güvenilir bir kaynak olarak kullanılır.
   - URLScan.io: URL'lerin taranıp analiz edilmesini sağlayan bir diğer API'dir. Şüpheli URL'ler burada analiz edilip tespit edilebilir.
   - Açık veri setleri: Mevcut literatürdeki kötü niyetli URL veri setleri incelenecek ve bunlar araştırma için kullanılabilecektir. Örneğin, PhishTank, Malware Domain List gibi veri setleri zararlı URL örnekleri içerir.

Python kütüphaneleri:

- requests ve json modülleri, API'lerden veri çekmek için kullanılacaktır.
- Pandas ise çekilen URL'leri tablo yapısında saklamak ve işlemek için kullanılacaktır.

2. **Veri Ön İşleme**

Toplanan ham URL verileri, modelde kullanılmadan önce bir ön işleme aşamasından geçirilecektir. Bu aşama, veri temizliği ve dönüşüm işlemlerini içerir:

- **Temizleme**: URL'lerden gereksiz karakterler, boşluklar, yorumlar ve semboller çıkarılır. Bu işlem, veri setindeki gürültüyü azaltarak modelin daha doğru çalışmasını sağlar [8].
- **Tokenizasyon:** URL'ler kelimelere veya anlamlı parçalara bölünür. Bu işlem, URL'nin yapısında önemli olan anahtar kelimeleri veya alan adlarını ortaya çıkarır [6].
- **Normalizasyon:** Büyük/küçük harf duyarlılığını ortadan kaldırmak için tüm URL'ler küçük harf olarak normalleştirilir. Böylece "ÖrnekWebSitesi.com" ile "örnekwebsitesi.com" aynı şekilde ele alınır [7].

Python kütüphaneleri:

- re (regex), metin ve karakter temizleme için kullanılabilir.
- NLTK ve spaCy gibi doğal dil işleme kütüphaneleri tokenizasyon için kullanılabilir [6].

3. **Özellik Çıkarma**

Verilerin anlamlı hale getirilmesi amacıyla doğal dil işleme (NLP) yöntemleri kullanılarak URL'lerden özellikler çıkarılacaktır:

- **N-gram Analizi:** URL'lerin n-gram şeklinde (örneğin, 2-gram veya 3-gram) analiz edilmesi, URL'deki önemli karakter veya kelime dizilerini anlamaya yardımcı olur [10].
- **TF-IDF (Term Frequency - Inverse Document Frequency):** TF-IDF yöntemi, URL'lerdeki kelimelerin önemini belirlemek için kullanılacaktır. TF-IDF, URL'lerde sık kullanılan ama çok da anlamlı olmayan kelimeleri (örneğin "www" veya "com") düşük ağırlıkla değerlendirirken, daha nadir ama önemli kelimeleri yüksek ağırlıkla değerlendirir [9].

Python kütüphaneleri:

- Scikit-learn'ün TfidfVectorizer ve CountVectorizer sınıfları, özellik çıkarımı için uygundur [16].

4. **Model Seçimi ve Eğitimi**

Özellik çıkarımı tamamlandıktan sonra zararlı URL tespiti için makine öğrenmesi ve derin öğrenme modelleri kullanılacaktır. Kullanılacak algoritmalar şunlardır:

- **Lojistik Regresyon:** Basit bir sınıflandırma modeli olup, özellikle doğrusal ilişkilerin olduğu veri setlerinde etkilidir. Hızlı ve hafif bir model olarak öncelikle test edilebilir [11].
- **Karar Ağaçları ve Rastgele Ormanlar (Random Forests):** Karar ağaçları, verilerdeki daha karmaşık ilişkileri öğrenebilir. Rastgele Ormanlar, birden fazla karar ağacının birleştirilmesiyle daha güçlü sonuçlar üretebilir [14].
- **Destek Vektör Makineleri (SVM):** SVM, veriyi yüksek boyutlu bir uzaya yansıtarak daha ayrıntılı sınıflandırma yapabilir. Özellikle, veri setinde sınıflar arası ayrım zor olduğunda etkilidir [13].
- **Derin Öğrenme Modelleri:** Daha ileri seviye sınıflandırma performansı için derin öğrenme modelleri kullanılabilir:
  - LSTM (Long Short-Term Memory): URL'lerin sıralı yapılarını anlamak için zaman serisi veri analizi yapan bu model kullanılabilir. LSTM, özellikle URL'lerin sıralı karakter yapılarındaki kalıpları yakalayarak daha başarılı olabilir [12].
  - CNN (Convolutional Neural Network): CNN, URL'lerin içinde yer alan karakter dizilerini ve kombinasyonlarını inceleyerek zararlı içerik olup olmadığını öğrenebilir [13]

Python kütüphaneleri:

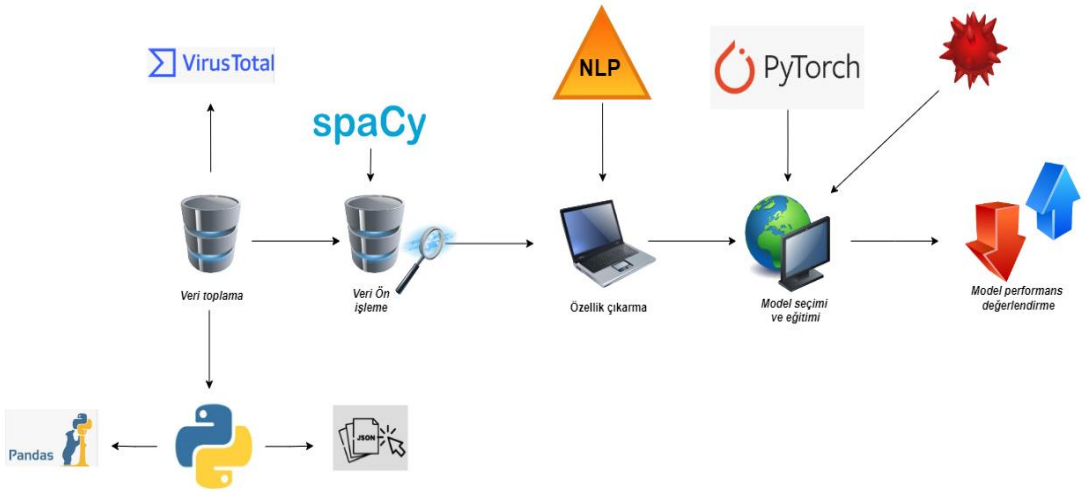
- Keras ve TensorFlow derin öğrenme modellerinin eğitimi ve test edilmesi için kullanılabilir [13].
- PyTorch, esnek ve dinamik derin öğrenme modelleri oluşturmak için bir diğer popüler framework'tür [13].

5. Model Performansı ve Değerlendirme

Modellerin performansı, doğruluk (accuracy), hata oranı (error rate), hassasiyet (precision), geri çağırma (recall) ve F1 skoru gibi ölçütlerle ölçülecektir. Ayrıca ROC eğrisi ve AUC (Area Under Curve) gibi görselleştirme araçları kullanılarak modelin genel performansı değerlendirilecektir [15].

Python kütüphaneleri:

- Scikit-learn'ün classification_report, confusion_matrix ve roc_curve fonksiyonları, bu değerlendirme metriklerini hesaplamak için kullanılabilir [16].
- Matplotlib ve Seaborn, görselleştirme işlemleri için kullanılabilir [16].



Şekil 2. Zararlı URL Tespiti İçin Yöntem

6. Kullanıcı Arayüzü Tasarımı ve Entegrasyonu

Bu aşamada, kullanıcıların zararlı URL tespiti uygulamasını kolayca kullanabilmelerini sağlayacak bir arayüz geliştirilecektir. Tasarımın temel bileşenleri şunlardır:

- **Kullanıcı Girişi**: Kullanıcılar, test etmek istedikleri URL'leri sisteme girebilir veya toplu veri yükleyebilirler.
- **Sonuç Görselleştirme**: Zararlı URL'lerin tespit sonuçları kullanıcıya açık bir şekilde gösterilecektir.
- **Geri Bildirim Mekanizması**: Kullanıcılar, tespit sonuçları hakkında geri bildirimde bulunabilir.

Bu aşamada **React** kütüphanesi kullanıcı arayüzü için, **Node.js** ve **Firebase** ise arka uç hizmetleri ve veri saklama için kullanılacaktır.

Şekil 3. Zararlı URL Tespiti İçin Web Arayüzü Tasarımı

3 PROJE YÖNETİMİ

3.1 İş- Zaman Çizelgesi

Araştırma önerisinde yer alacak başlıca iş paketleri ve hedefleri, her bir iş paketinin hangi sürede gerçekleştirileceği, başarı ölçütü ve araştırmanın başarısına katkısı "İş-Zaman Çizelgesi" doldurularak verilir. Literatür taraması, gelişme ve sonuç raporu hazırlama aşamaları, araştırma sonuçlarının paylaşımı, makale yazımı ve malzeme alımı ayrı birer iş paketi olarak gösterilmemelidir.
Başarı ölçütü olarak her bir iş paketinin hangi kriterleri sağladığında başarılı sayılacağı açıklanır. Başarı ölçütü, ölçülebilir ve izlenebilir nitelikte olacak şekilde nicel veya nitel ölçütlerle (ifade, sayı, yüzde, vb.) belirtilir.

İŞ-ZAMAN ÇİZELGESİ (*)

İP No	İş Paketlerinin Adı ve Hedefleri	Kim(ler) Tarafından Gerçekleştirileceği	Zaman Aralığı (..-.. Ay)	Başarı Ölçütü ve Projenin Başarısına Katkısı
1	Veri toplama - Zararlı ve güvenli URL'ler içeren veri seti oluşturulması	Yusuf Güney Lütfü Bedel	2 ay	Geniş kapsamlı, etiketli veri seti oluşturma ve analiz için hazır hale getirme
2	Veri ön işleme - Temizleme, tokenizasyon ve normalizasyon işlemleri	Yusuf Güney	2 ay	Veriyi model için uygun hale getirme
3	Özellik çıkarma - N-gram analizi ve TF-IDF kullanarak özellik çıkarımı yapma	Lütfü Bedel	2 ay	URL'lerden anlamlı özelliklerin çıkarılması
4	Model seçimi ve eğitimi - Lojistik regresyon, karar ağaçları, SVM ve derin öğrenme modelleri ile zararlı URL tespiti	Yusuf Güney Lütfü Bedel	2 ay	Modellerin doğruluk ve performans açısından değerlendirilmesi
5	Model değerlendirme - Performans metrikleri ile modelin test edilmesi	Lütfü Bedel	2 ay	Modelin başarısının ROC eğrisi ve AUC gibi metriklerle doğrulanması
6	Kullanıcı Arayüzü Tasarımı ve Entegrasyonu	Yusuf Güney	2 ay	Kullanıcı dostu ve modern bir arayüz oluşturulması

(*) Çizelgedeki satırlar ve sütunlar gerektiği kadar genişletilebilir ve çoğaltılabilir.

3.2 Risk Yönetimi

Araştırmanın başarısını olumsuz yönde etkileyebilecek riskler ve bu risklerle karşılaşıldığında araştırmanın başarıyla yürütülmesini sağlamak için alınacak tedbirler (B Planı) ilgili iş paketleri belirtilerek ana hatlarıyla aşağıdaki Risk Yönetimi Tablosu'nda ifade edilir. B planlarının uygulanması araştırmanın temel hedeflerinden sapmaya yol açmamalıdır.

RİSK YÖNETİMİ TABLOSU*

İP No	En Önemli Riskler	Risk Yönetimi (B Planı)
1	Veri setinin yetersiz kalması	Ek veri kaynaklarından veri toplama
2	Model performansının beklenen seviyede olmaması	Farklı model ve parametreler ile modelin yeniden eğitilmesi

(*) Tablodaki satırlar gerektiği kadar genişletilebilir ve çoğaltılabilir.

3.3. Araştırma Olanakları

Bu bölümde projenin yürütüleceği kurum ve kuruluşlarda var olan ve projede kullanılacak olan altyapı/ekipman (laboratuvar, araç, makine-teçhizat, vb.) olanakları belirtilir.

ARAŞTIRMA OLANAKLARI TABLOSU (*)

Kuruluşta Bulunan Altyapı/Ekipman Türü, Modeli (Laboratuvar, Araç, Makine-Teçhizat, vb.)	**Projede Kullanım Amacı**
Bilgisayar Laboratuvarı (Bursa Teknik Üniversitesi)	Model eğitimi ve test işlemlerinin yapılması

(*) Tablodaki satırlar gerektiği kadar genişletilebilir ve çoğaltılabilir.

4. YAYGIN ETKİ

Önerilen çalışma başarıyla gerçekleştirildiği takdirde araştırmadan elde edilmesi öngörülen ve beklenen yaygın etkilerin neler olabileceği, diğer bir ifadeyle yapılan araştırmadan ne gibi çıktı, sonuç ve etkilerin elde edileceği aşağıdaki tabloda verilir.

ARAŞTIRMA ÖNERİSİNDEN BEKLENEN YAYGIN ETKİ TABLOSU

Yaygın Etki Türleri	**Önerilen Araştırmadan Beklenen Çıktı, Sonuç ve Etkiler**
Bilimsel/Akademik (Makale, Bildiri, Kitap Bölümü, Kitap)	Bir adet bildiri yayınlanacaktır.
Ekonomik/Ticari/Sosyal (Ürün, Prototip, Patent, Faydalı Model, Üretim İzni, Çeşit Tescili, Spin-off/Start- up Şirket, Görsel/İşitsel Arşiv, Envanter/Veri Tabanı/Belgeleme Üretimi, Telife Konu Olan Eser, Medyada Yer Alma, Fuar, Proje Pazarı, Çalıştay, Eğitim vb. Bilimsel Etkinlik, Proje Sonuçlarını Kullanacak Kurum/Kuruluş, vb. diğer yaygın etkiler)	Geliştirilen model, finans, kamu ve internet sektörleri için güvenliği artıracak bir prototip veya ürün olarak ticarileştirilebilir.
Araştırmacı Yetiştirilmesi ve Yeni Proje(ler) Oluşturma (Yüksek Lisans/Doktora Tezi, Ulusal/Uluslararası Yeni Proje)	Bu çalışmanın sonuçlarının ulusal yeni Ar-Ge projelerinin ve yüksek lisans çalışmalarının da ufuklarını açacağı umut edilmektedir.

5. BÜTÇE TALEP ÇİZELGESİ

Bütçe Türü	Talep Edilen Bütçe Miktarı (TL)	Talep Gerekçesi
Sarf Malzeme	2500₺	Geliştirme aşamasında kullanılacak kaynak kitap ve yazılım lisans temini.
Makina/Teçhizat (Demirbaş)	2500₺	Zararlı URL tespiti için gerekli olan 1 TB SSD depolama birimi, veri setlerini güvenli şekilde saklamak için.
Hizmet Alımı	3500₺	Sistemin test ve geliştirme süreçlerinde bulut servis hizmet bedeli veri setlerini güvenli şekilde saklamak için
Ulaşım	500₺	Proje kapsamında yapılacak toplantılar ve saha çalışmaları için ulaşım masrafları.
TOPLAM	9000₺	

NOT: Bütçe talebiniz olması halinde hem bu tablonun hem de TÜBİTAK Yönetim Bilgi Sistemi (TYBS) başvuru ekranında karşınıza gelecek olan bütçe alanlarının doldurulması gerekmektedir. Yukardaki tabloda girilen bütçe kalemlerindeki rakamlar ile, TYBS başvuru ekranındaki rakamlar arasında farklılık olması halinde TYBS ekranındaki veriler dikkate alınır ve başvuru sonrasında değiştirilemez.

6. BELİRTMEK İSTEDİĞİNİZ DİĞER KONULAR

Sadece araştırma önerisinin değerlendirilmesine katkı sağlayabilecek bilgi/veri (grafik, tablo, vb.) eklenebilir.

7. EKLER

EK-1: KAYNAKLAR

[1]. Justin Saxe, Konstantin Berlin, Deep neural network based malware detection using two dimensional binary program features, 2017.

[2]. Andres C. Bahnsen, J. Raul Torroledo, Javier Camacho, Sergio Villegas, Classifying phishing URLs using recurrent neural networks, 2017.

[3]. Xianliang Xu, Haibing Zhu, Enhanced malicious URL detection using convolutional neural networks, 2020.

[4]. Enes Büber, Bilge Diri, DDİ yöntemleri ile oltalama saldırılarının URL'den tespit edilmesi, 2020.

[5]. Mustafa Sait Karaman, Murat Turan, Mehmet Aydın, Yapay sinir ağı kullanılarak anomali tabanlı saldırı tespit modeli, 2020.

[6] Fernandez, M., Yamagishi, L., & Leban, M. (2018). Tokenization techniques in Natural Language Processing: From regex to sub-word tokenization. Natural Language Processing for Social Media.

[7] Aggarwal, C. C., & Zhai, C. (2012). Mining text data (Vol. 4). Springer Science & Business Media.

[8] Gharib, T., Tran, T., & Sharaf, M. A. (2016). Feature extraction approaches for malicious URL detection: A review. Journal of Information Security and Applications, 32, 3-8.

[9] Ramos, J. (2003). Using tf-idf to determine word relevance in document queries. Proceedings of the First International Conference on Machine Learning.

[10] Al-Shawashreh, M., & Muheidat, F. (2019). Phishing URL detection using n-gram analysis and machine learning techniques. Applied Computing and Informatics, 15(1), 2-10.

[11] Bouazizi, M., & Ohtsuki, T. (2017). A pattern-based approach for multi-class sentiment analysis in Twitter. IEEE Access, 5, 20617-20639.

[12] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780.

[13] Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. Nature, 521(7553), 436-444.

[14] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32.

[15] Fawcett, T. (2006). An introduction to ROC analysis. Pattern Recognition Letters, 27(8), 861-874.

[16] Pedregosa, F., Varoquaux, G., Gramfort, A., et al. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830.