

Güvenli Belge Anonimleştirme Sistemi

Anıl Engin Keretli

Bilgisayar Mühendisliği Bölümü
Kocaeli Üniversitesi
Kocaeli, Türkiye
230201128
anilengin.keretli@gmail.com

Bilge Çeşme

Bilgisayar Mühendisliği Bölümü
Kocaeli Üniversitesi
Kocaeli, Türkiye
220201045
bilge.cesme@gmail.com

I. GİRİŞ

A. Projenin Amacı

Bu projenin amacı, akademik makalelerin anonimleştirilmesini sağlayan bir sistem geliştirmektir. Akademik değerlendirme süreçlerinde çift-kör hakemlik (double-blind review) yöntemi kullanılarak, yazar ve hakem bilgileri gizlenerek objektif değerlendirme sağlanmaktadır. Bu proje, kullanıcıların sisteme makale yükleyebilmesini, makalelerin anonimleştirilmesini ve hakemler tarafından incelenerek geri bildirim verilmesini mümkün kılmaktadır.

Geliştirilen sistem, Python programlama dili kullanılarak oluşturulmuş olup, metin işleme ve anonimleştirme işlemleri için doğal dil işleme (NLP) tekniklerinden faydalanmaktadır. Aynı zamanda, güvenli veri saklama ve yetkilendirme mekanizmaları ile makalelerin korunmasını hedeflemektedir.

B. Projenin Kapsamı

Bu proje, akademik makalelerin anonimleştirilmesi sürecini dijital ortamda yönetmek için tasarlanmıştır. Sistemin temel bileşenleri şunlardır:

Kullanıcı Modülü: Yazarlar, sisteme giriş yapmadan veya giriş yaparak makalelerini yükleyebilir.

Anonimleştirme Modülü: Makaleler otomatik olarak analiz edilerek yazar isimleri, kurum bilgileri ve iletişim bilgileri anonim hale getirilir.

Hakem Modülü: Hakemler, anonimleştirilmiş makaleleri inceleyerek değerlendirme yapar.

Yönetici Modülü: Editörler, makale süreçlerini yönetir ve hakem atamalarını gerçekleştirir.

Veritabanı ve Güvenlik: Makaleler güvenli bir şekilde saklanır ve sadece yetkilendirilmiş kişiler tarafından erişilebilir.

Sistem, akademik değerlendirme süreçlerinin şeffaflığını ve tarafsızlığını artırarak, bilimsel yayınların daha güvenilir bir şekilde incelenmesini sağlamayı amaçlamaktadır.

II. YÖNTEM

A. Geliştirme Ortamı ve Temel Teknolojiler

Projenin geliştirilmesinde Python programlama dili tercih edilmiş olup, PDF dosyalarının işlenmesi ve metin analizi için çeşitli kütüphaneler kullanılmıştır. Öncelikle, PDF dosyalarının okunması ve işlenmesi için PyMuPDF (fitz) kütüphanesi tercih edilmiştir. Bu kütüphane, PDF dosyalarından metin çıkarma, sayfa düzeni analizi ve içerik manipülasyonu gibi temel işlevleri sağlamaktadır. Doğal dil işleme görevleri için spaCy kütüphanesi kullanılmıştır. spaCy'nin tercih edilmesinin nedeni, özellikle Named Entity Recognition (NER) konusundaki başarısı ve Türkçe dil desteği sunmasıdır. Bu kütüphane sayesinde metinlerdeki kişi isimleri, kurum isimleri ve diğer özel isimler başarıyla tespit edilebilmektedir.

B. Web Uygulaması ve Veritabanı

Web uygulamasının geliştirilmesinde Flask framework'ü kullanılmıştır. Flask'ın tercih edilmesinin nedeni, hafif yapısı, kolay entegre edilebilirliği ve Python ekosistemi ile uyumlu çalışmasıdır. Veritabanı işlemleri için SQLAlchemy



Fig. 1. Ana Panel

ORM kullanılmış, bu sayede veritabanı işlemleri daha güvenli ve yönetilebilir hale getirilmiştir. Ayrıca, kullanıcı arayüzünün geliştirilmesinde Bootstrap framework'ü kullanılarak modern ve responsive bir tasarım elde edilmiştir.

C. Güvenlik ve Performans Optimizasyonu

Sistemin güvenliği için bcrypt kütüphanesi ile şifreleme işlemleri gerçekleştirilmiş, JWT (JSON Web Token) kullanılarak güvenli oturum yönetimi sağlanmıştır. Asenkron işlemler ve performans optimizasyonu için asyncio ve aiohttp kütüphaneleri kullanılmış, bu sayede sistem yüksek yük altında bile verimli çalışabilir hale getirilmiştir. Ayrıca, logging modülü kullanılarak sistem logları tutulmuş ve hata ayıklama süreci kolaylaştırılmıştır.

D. Problem Tanımı

Akademik makalelerin anonimleştirilmesi sürecinde karşılaşılan temel zorluklar şunlardır:

- Yazar ve kurum bilgilerinin tutarlı bir şekilde tespit edilmesi
- Dolaylı referansların (self-citation, acknowledgments vb.) belirlenmesi
- Metin içindeki gizli tanımlayıcıların tespiti
- PDF formatındaki belgelerin yapısal bütünlüğünün korunması
- Anonimleştirme işleminin belge formatını bozmaması
- İşlem süresinin optimize edilmesi

E. Literatür Taraması

Akademik makalelerin otomatik anonimleştirilmesi konusunda yapılan önceki

Makale Listesi				
Araştırma kelime ile ara...		Tüm Durumlar	Tarihe Göre (Yeni-Eski)	
Makale ID	Başlık	Durum	Tarih	İşlemler
4	Emotion Recognition Using Temporally Localized Emotional Events in EEG With Naturalistic Context: DENSEP Dataset	Kaydedildi	2 Nisan 2025 20:30	Gör
3	deneme	Kaydedildi	2 Nisan 2025 15:42	Gör
2	diğer	Kaydedildi	2 Nisan 2025 15:16	Gör
1	erit	Onaylandı	2 Nisan 2025 15:00	Gör

Fig. 2. Editör

çalışmalar incelendiğinde, çeşitli yaklaşımlar ve metodolojiler göze çarpmaktadır:

1) *Doğal Dil İşleme Yaklaşımları*: Literatürde, NLP tabanlı anonimleştirme çözümleri genellikle şu teknikleri kullanmaktadır:

- Named Entity Recognition (NER)
- Pattern Matching
- Machine Learning based Classification
- Rule-based Systems

F. Maskeleye İşlemi ve Veri Anonimleştirme

PDF dokümanlarındaki hassas bilgilerin maskelenmesi için çok aşamalı bir yaklaşım benimsenmiştir. İlk olarak, spaCy kütüphanesinin NER (Named Entity Recognition) modeli kullanılarak metindeki kişi isimleri, kurumlar ve e-posta adresleri tespit edilmektedir. Tespit edilen bu hassas bilgiler, önceden tanımlanmış kalıplarla değiştirilmektedir; örneğin kişi isimleri [AUTHOR], e-posta adresleri [EMAIL], kurum isimleri [INSTITUTION] şeklinde maskelenmektedir. Ayrıca, self-citation tespiti için özel bir algoritma geliştirilmiş olup, yazarın kendi çalışmalarına yaptığı atıflar da [SELF-CITATION] etiketi ile maskelenmektedir. PyMuPDF kütüphanesi kullanılarak, maskelenen bu bilgiler PDF dosyasına geri yazılmakta ve orijinal formatı korunarak yeni bir anonim PDF dokümanı oluşturulmaktadır.

1) *PDF İşleme Teknikleri*: PDF dokümanlarının işlenmesinde kullanılan başlıca yaklaşımlar:

- OCR (Optical Character Recognition)
- PDF parsing
- Layout analysis
- Content extraction

III. SİSTEM MIMARISI

A. Teknoloji Yığını

Sistem, modern ve güvenilir teknolojiler üzerine inşa edilmiştir:

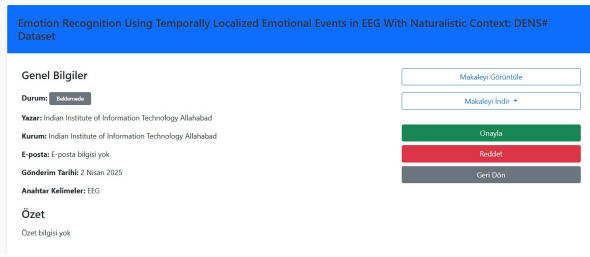


Fig. 3. Admin Makale Paneli

The screenshot shows the 'Hakem Dashboard' (Reviewer Dashboard) with a blue header 'Makale Değerlendirme Sistemi' and a user profile 'Hoş geldiniz, Hakem-3' with a 'Çıkış Yap' (Logout) button. Below the header, there's a 'Size Atanan Makaleler' (Articles Assigned to You) section. It contains a table with columns: 'Takip Kodu' (Tracking Code), 'Başlık' (Title), 'Anahtar Kelimeler' (Keywords), 'Yükleme Tarihi' (Upload Date), 'Durum' (Status), and 'İşlemler' (Actions). The table lists four articles with their respective tracking codes, titles, keywords, upload dates, and status. Each row has buttons for 'Görüntüle' (View), 'Onayla' (Approve), and 'Değerlendir' (Evaluate).

Takip Kodu	Başlık	Anahtar Kelimeler	Yükleme Tarihi	Durum	İşlemler
MXL-20250402-0004	Emotion Recognition Using Temporally Localized Emotional Events in EEG With Naturalistic Context: DENS# Dataset	EEG	-	Beklemede	Görüntüle Değerlendir
MXL-20250402-0001	deneme	deneme	-	Beklemede	Görüntüle Değerlendir
MXL-20250402-0002	diğer	ıgh	-	Beklemede	Görüntüle Değerlendir
MXL-20250402-0001	erfj	kjh	-	Beklemede	Görüntüle Değerlendir

Fig. 4. Hakem Dashboard

1) Backend Teknolojileri:

- Python 3.8+: Yüksek performanslı ve geniş kütüphane desteği
- Spacy 3.0: State-of-the-art NLP özellikleri
- PyMuPDF 1.19.0: PDF manipülasyonu için güçlü araçlar
- Flask 2.0: Hafif ve esnek web framework
- SQLAlchemy: ORM ve veritabanı yönetimi
- Redis: Önbellekleme ve iş kuyruğu yönetimi

2) Frontend Teknolojileri:

- HTML5/CSS3: Modern web standartları
- JavaScript/jQuery: Dinamik kullanıcı arayüzü
- Bootstrap 5: Responsive tasarım
- AJAX: Asenkron veri transferi

IV. TEST VE HATA AYIKLAMA

Projenin test ve hata ayıklama sürecinde, PDF dosyalarının işlenmesi ve metin çıkarma fonksiyonları üzerinde kapsamlı testler gerçekleştirilmiştir. Özellikle Türkçe karakterler ve özel semboller içeren akademik makalelerde karakter kodlama sorunlarıyla karşılaşmış, bu sorunlar PyMuPDF kütüphanesinin gelişmiş özellikleri kullanılarak çözülmüştür. Veritabanı tarafında yaşanan yabancı anahtar kısıtlamaları ve veri tipi uyumsuzlukları, şema optimizasyonu ile giderilmiştir. Yapılan stres testleri sonucunda sistem performansı iyileştirilmiş ve çoklu kullanıcı senaryolarında başarılı sonuçlar elde edilmiştir. Bu süreç, ekibe PDF işleme ve veritabanı yönetimi konularında değerli deneyimler kazandırmıştır.

V. YALANCI KOD

A. ANA UYGULAMA YAPISI

- **Uygulama Başlatma**
 - Flask uygulaması oluştur
 - Rotaları tanımla

- Hata ayıklama modunda çalıştır

• Takip Numarası Üretme

- Rastgele 7 haneli benzersiz numara oluştur
- Veritabanında var olup olmadığını kontrol et
- Benzersiz numarayı döndür

B. Rotalar ve İşlevler

• Ana Sayfa (/)

- index.html Şablonunu göster

• Editör Paneli (/editor)

- Tüm makaleleri veritabanından al
- Her makale için:
 - * Uygun hakemleri listele
 - * Makale durumuna göre bilgileri hazırla
- editor.html Şablonunu makale listesiyle göster

• Makale Görüntüleme (/makale/<id>)

- Veritabanından makale PDF'ini al
- PDF'i tarayıcıda göster
- Bulunamazsa hata mesajı döndür

• Hakem Atama (/assign_hakem)

- Gelen JSON verisini işle
- Makale ve hakem kimliklerini al
- Veritabanında hakem atamasını yap
- Makale durumunu güncelle
- Sonucu JSON olarak döndür

C. Ana İş Akışları

• Makale Yükleme Akışı

- 1) Yazar bilgilerini ve PDF'i gönderir
- 2) Sistem makale başlığını ve konusunu belirler
- 3) Benzersiz takip numarası oluşturur
- 4) Veritabanına kaydeder
- 5) Takip numarasını yazara verir

- 6) Yazar takip numarasını takip sayfasına girer
- 7) Yazar numarayla istediği zaman kullanır

- **Anonimleştirme Modülü**

- 1) Makaledeki yazar isimleri ve kurum bilgilerini tespit eder
- 2) Tespit edilen bilgileri anonimleştirir
- 3) Anonimleştirilmiş PDF oluşturur

- **Hakem Atama Modülü**

- 1) Makale konusuna uygun hakemleri listeler
- 2) Editör tarafından hakem ataması yapılır
- 3) Atama sonucu veritabanına kaydedilir

- **Değerlendirme Modülü**

- 1) Hakem anonimleştirilmiş makaleyi görür
- 2) Değerlendirme yazısını sisteme girer
- 3) Değerlendirme makale PDF'sine eklenir

- **Takip Modülü**

- 1) Yazarlar makale durumlarını takip edebilir
- 2) Editörler bütün süreci izleyebilir

D. Temel Fonksiyonlar

- Veritabanı işlemleri (CRUD)
- PDF işleme ve değerlendirme ekleme
- Makale analizi (başlık ve konu çıkarma)
- Durum yönetimi (makale yaşam döngüsü)
- Hata yönetimi ve kullanıcı geri bildirimi

E. Hata Durumları

- Geçersiz PDF dosyası
- Eksik form bilgisi
- Makale bulunamadı
- Veritabanı hatası
- Yetkisiz erişim denemesi

F. Algoritmalar

1) Veritabanı Yönetim Algoritmaları

- `createTables()`: Tüm veritabanı tablolarını oluşturur
- `veriOlustur()`: Örnek başlık ve hakem verilerini oluşturur
- CRUD operasyonları: Makale, hakem ve değerlendirme kayıtlarını yönetir

2) Makale Konu Belirleme Algoritması

- Makale metninden anahtar kelimeler çıkarılır

Fig. 5. Değerlendirme Sayfası:

- Anahtar kelimeler önceden tanımlanmış konularla karşılaştırılır
- En yüksek benzerlik skoruna sahip konu seçilir
- `extract_title_and_keywords_from_byt` PDF'den başlık ve anahtar kelimeleri çıkarır
- `detailed_extract_keywords()`: Metin içeriğinden RAKE-benzeri yöntemle anahtar kelimeler çıkarır

3) Anonimleştirme Algoritması

- PDF'deki metin blokları analiz edilir
- Kişi isimleri tespit edilir
- Tespit edilen isimler siyah şeritlerle kapatılır
- `find_exclusion_zones()`: Akademik makalelerde anonimleştirilmeyecek bölgeleri belirler
- `redact_text()`: Yazar adı, e-posta ve kurum bilgilerini siyah şeritlerle kapatır
- `blur_faces()`: PDF'deki yüz görüntülerini bulanıklaştırır
- `anonymize_pdf()`: Tüm anonimleştirme adımlarını koordine eden ana fonksiyondur

4) Hakem Eşleştirme Algoritması

- Makale konusuna göre uygun hakemler sorgulanır
- Hakemlerin uzmanlık alanlarıyla makale konusu eşleştirilir

5) Mesajlaşma Algoritması

- Makale konusunda yazar ve editörler arası mesajlaşmayı sağlar

İsim

▼ Tablolar (6)

- > article_reviewers
- > articles
- > reviewers
- > reviews
- > sqlite_sequence
- > users

Fig. 6. Tablolar:

TABLE I
VARLIK-ÖZELLİK TABLOSU

Varlık	Özellikler
Makale	id, title, author, institution, email, abstract, keywords, status, submission_date
Hakem	email, name, assigned_articles
Değerlendirme	reviewer_name, decision, comments, review_date
Admin	email, name, role
Durum	status_name (Onay Bekliyor, Onaylandı, Değerlendiriliyor, vb.)

VI. SONUÇ

Geliştirilen akademik makale yönetim sistemi, yazarlar, editörler ve hakemler için kullanışlı bir platform sunmaktadır. Sistemin temel özellikleri şunlardır:

- Kullanıcı dostu arayüz ile kolay makale yükleme
- Otomatik anonimleştirme ile adil değerlendirme süreci
- Konu bazlı hakem atama ile etkin süreç yönetimi
- Şeffaf makale takip sistemi
- Güvenli veri saklama ve işleme

Proje, akademik yayın süreçlerini dijitalleştirerek daha verimli hale getirmeyi başarmıştır. Özellikle anonimleştirme ve hakem atama süreçlerindeki otomasyon, sistemin en güçlü yönlerini oluşturmaktadır.

VII. KAYNAKÇA

- Flask <https://flask.palletsprojects.com>
- PyMuPDF <https://pymupdf.readthedocs.io/>
- spaCy <https://spacy.io/>
- SQLite <https://www.sqlite.org/>

VIII. KATKILAR

Bu projede Anıl Engin KERETLİ database ve raporlama ile ilgilenmiş, Bilge ÇEŞME ise web sitesinin frontend ve bacekendini yapmıştır.