ANKARA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ

ENDÜSTRİYEL UYGULAMALAR İÇİN YENİ BİR DİL MODELİ UYGULAMASI

Göktuğ ÖZENBAŞ

BİLGİSAYAR MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA
2025

ÖZET

Yüksek Lisans Tezi

ENDÜSTRİYEL UYGULAMALAR İÇİN YENİ BİR DİL MODELİ UYGULAMASI

Göktuğ ÖZENBAŞ

Ankara Üniversitesi
Fen Bilimleri Enstitüsü
Bilgisayar Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Mehmet Serdar GÜZEL

Bu tez çalışmasında, Türkiye'deki sanayi kuruluşlarının resmi işlemlerinde kullanmakla yükümlü oldukları PRODTR ürün sınıflandırma sistemine yönelik anlam temelli bir öneri sistemi geliştirilmiştir. Çalışmanın temel amacı, sentetik olarak üretilmiş ürün ve teknik ad verilerinin, doğal dil işleme (NLP) ve gömülü temsillere (embedding) dayalı yöntemlerle doğru PRODTR kodlarıyla eşleştirilmesidir.

Geliştirilen sistem, OpenAI tabanlı embedding modelleri kullanarak metinleri yüksek boyutlu vektörlere dönüştürmekte ve bu temsilleri Elasticsearch altyapısı üzerinde kosinüs benzerliği aracılığıyla karşılaştırmaktadır. Böylece yalnızca yüzeysel kelime eşleşmelerine dayalı değil, bağlamı dikkate alan anlamsal bir sınıflandırma sağlanmıştır.

Üretken yapay zeka ile oluşturulan ve etiketlenmiş 93 sorgudan oluşan test kümesiyle yapılan değerlendirmelerde sistemin %93,5 doğruluk oranına ulaştığı görülmüştür. Ayrıca öneri skorları ile tahmin başarısı arasında istatistiksel olarak anlamlı bir ilişki bulunmuş; yüksek benzerlik skorlarının çoğunlukla doğru sınıflarla örtüştüğü gözlemlenmiştir. Bu durum, sistemin yalnızca otomatik bir eşleştirme aracı değil aynı zamanda güvenilir bir karar destek mekanizması olarak kullanılabileceğini göstermektedir.

Sonuç olarak geliştirilen öneri sistemi; ürün sınıflandırma, envanter yönetimi, kamu raporlama süreçleri ve sektörel danışmanlık uygulamalarında esnek, açıklanabilir ve bağlam duyarlı bir yapay zekâ çözümü olarak öne çıkmaktadır.

Ekim 2025, 43 sayfa

Anahtar Kelimeler: Doğal Dil İşleme, Ürün Sınıflandırması, Gömülü Temsil, Öneri Sistemi

ABSTRACT

Master Thesis

IMPLEMENTATION OF A NEW LANGUAGE MODEL FOR INDUSTRIAL APPLICATIONS

Göktuğ ÖZENBAŞ

Ankara University
Graduate School of Natural and Applied Sciences
Department of Computer Engineering

Supervisor: Prof. Dr. Mehmet Serdar GÜZEL

This thesis introduces a semantic recommendation system designed to support the PRODTR product classification framework, which is mandatory in various industrial and governmental processes in Türkiye. The main objective of the study is to automatically match synthetic product descriptions with the appropriate PRODTR codes using natural language processing (NLP) and embedding-based methods.

The proposed system leverages OpenAI's embedding models to transform textual inputs into high-dimensional vector representations. These vectors are then indexed and compared within an Elasticsearch environment using cosine similarity, enabling context-aware semantic classification beyond traditional keyword matching approaches.

Evaluation was conducted with a labeled dataset of 93 synthetic queries, and the system achieved an accuracy rate of 93.5% when only the top-ranked suggestion was considered. Moreover, a statistically significant correlation was identified between similarity scores and prediction accuracy, indicating that higher scores are strongly associated with correct classifications. This demonstrates that the system functions not only as an automatic matching tool but also as a reliable decision support mechanism.

In conclusion, the developed recommendation system emerges as a robust, explainable, and contextually intelligent digital assistant with potential applications in product classification, inventory management, public reporting, and sectoral consultancy.

October 2025, 43 pages

Keywords: Natural Language Processing, Product Classification, Embedding Representation, Recommendation System

ÖNSÖZ VE TEŞEKKÜR

Yüksek lisans çalışmalarım boyunca akademik gelişimime sunduğu katkıların yanı sıra, profesyonel kariyer yolculuğuma atılmamda da doğrudan rehberlik eden danışman hocam Sayın Prof. Dr. Mehmet Serdar GÜZEL'e en derin saygı ve teşekkürlerimi sunarım. Bilimsel yaklaşımı, yol göstericiliği ve desteği, yalnızca bu tezin değil aynı zamanda mesleki yönelimimin de temel taşlarını oluşturmuştur.

Göktuğ ÖZENBAŞ
Ankara, Ekim 2025

İÇİNDEKİLER

KISALTMALAR DİZİNİ

AI	Artificial Intelligence
API	Application Programming Interface
BERT	Bidirectional Encoder Representations from Transformers
GPT	Generative Pre-trained Transformer
JSON	JavaScript Object Notation
LLM	Large Language Model (Büyük Dil Modeli)
NACE	Nomenclature of Economic Activities (AB Faaliyet Kodları)
NLP	Natural Language Processing (Doğal Dil İşleme)
PRODTR	Türkiye Ürün Sınıflandırma Sistemi

ŞEKİLLER DİZİNİ

ÇİZELGELER DİZİNİ

1. GİRİŞ

1.1 Tezin Amacı ve Kapsamı

Bu yüksek lisans tezinin temel amacı Türkiye'deki sanayi kuruluşlarının çeşitli resmi işlemlerinde kullanmakla yükümlü oldukları PRODTR ürün sınıflandırma sistemine uygun veri girişlerinin, serbest metin girdileri üzerinden otomatik ve anlam temelli biçimde önerilmesini sağlayan bir sistem geliştirmektir. Bu bağlamda, üretim ve beyan süreçlerinde sıkça rastlanan yanlış veya eksik kodlamaların önüne geçilmesi, veri standardizasyonunun sağlanması ve kamusal sistemlerle uyumlu bir altyapı sunulması hedeflenmektedir.

Bu amaç doğrultusunda büyük dil modellerinin (LLM) yeteneklerinden faydalanan embedding tabanlı anlamsal benzerlik ilkesiyle çalışan bir öneri sistemi geliştirilmiştir.
Tez, başlığına uygun biçimde, "Endüstriyel Uygulamalar İçin Yeni Bir Dil Modeli Uygulaması" çerçevesinde hem yapay zekâ uygulamalarına hem de kamu-özel sektör veri entegrasyonuna katkı sağlamayı hedeflemektedir.

1.2 Prolemin Tanımı

PRODTR (Product Classification System for Türkiye), Türkiye Cumhuriyeti devlet kurumları tarafından kullanılan ve ürünlerin istatistiksel, ekonomik ve ticari işlemlerde standart biçimde tanımlanmasına imkân sağlayan bir sınıflandırma sistemidir. Özellikle teşvik başvuruları, sanayi sicil belgeleri, gümrük beyanları ve kamu destek programlarında ürünlerin doğru kodlarla beyan edilmesi gerekmektedir.
Ancak mevcut pratikte firmalar bu sistemlere ürün bilgilerini genellikle serbest metin formatında çoğunlukla teknik olmayan ve standart dışı ifadelerle girmektedir. Bu durum:

- Yanlış sınıflandırmalara,
- İstatistiksel veri bozulmasına,
- Beyan hatalarına ve teşvik reddine,
- Karar destek sistemlerinin yanlış yönlendirilmesine neden olmaktadır.

PRODTR gibi kod sistemlerinin doğru çalışabilmesi için her ürün tanımının sistemin yapısına uygun şekilde sınıflandırılması gerekmektedir. Ancak bu süreç manuel olarak hem zaman alıcıdır hem de insan hatasına fazlasıyla açıktır. Bu yüksek lisans tezinin çıkış noktası da bu sorunu otomatik anlamsal temelli öneri tabanlı bir NLP çözümü ile aşmaktır.

1.3 Tezin Önemi ve Kapsamı

Bu yüksek lisans tezi hem akademik hem de uygulamalı düzeyde önemli katkılar sunmaktadır:

1. Kamu süreçlerinin dijitalleşmesine katkı sağlar: Geliştirilen öneri sistemi, serbest metin verilerini PRODTR sistemine otomatik olarak bağlayarak dijital dönüşüm sürecine katkı sunmaktadır.
2. Yapay zekâ ile veri standardizasyonunu birleştirir: NLP ve embedding teknolojilerini kullanarak geleneksel veri girişlerini anlamsal olarak eşleştirir.
3. Başka sınıflandırma sistemleri için de altyapı sağlar: Sadece PRODTR değil GTIP, NACE, CPA gibi diğer sınıflandırmalara uyarlanabilir.

Bu yönüyle tez kamusal veri kalitesini artıran ve aynı zamanda endüstriyel sistemleri dil modeli teknolojisi ile buluşturan bütüncül bir çözüm önermektedir.

1.4 Yöntem ve Genel Yaklaşım

Bu tezde geliştirilen sistem doğal dil işleme teknikleri kullanarak aşağıdaki adımları izlemektedir:

1. Veri Toplama: PRODTR kodları temel alınarak ChatGPT tabanlı generative AI ile firmaların serbest metin biçiminde girebileceği ürün açıklamaları ve ürün adları simüle edilmiştir
2. Embedding Dönüşümü: OpenAI'nin text-embedding-3-small modeli kullanılarak hem firma açıklamaları hem de PRODTR ürün tanımları yüksek boyutlu vektörlere dönüştürülmüştür.

3. Benzerlik Analizi: Elasticsearch motoru üzerinde kosinüs benzerliği (cosine similarity) yöntemi kullanılarak serbest metin ile en benzer PRODTR tanımları hesaplanmıştır.
4. Öneri Üretimi: Elde edilen benzerlik skorlarına göre kullanıcı girdisine en uygun 10 PRODTR kodu öneri olarak sunulmuştur.

Sistem sözlük eşleşmesi kavramının dışına çıkarak anlamsal düzeyde doğru sınıflandırma yapabilmektedir. Bu yaklaşım verimlilik, doğruluk ve tekrarlanabilirlik açısından avantajlar sağlamaktadır.

2. KURAMSAL TEMELLER VE LİTERATÜR İNCELEMESİ

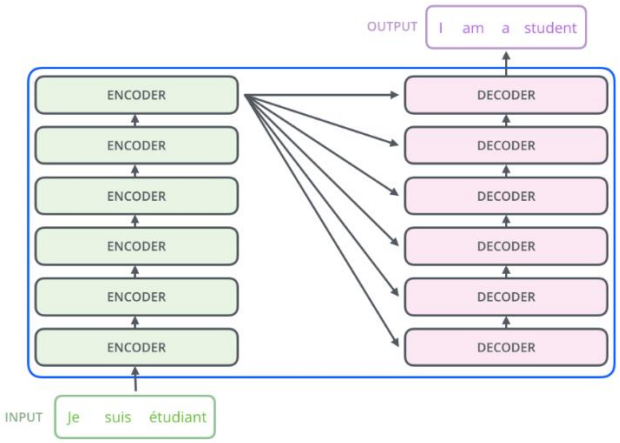
2.1 Doğal Dil İşleme (NLP)

Doğal Dil İşleme (NLP) bilgisayar sistemlerinin doğal dili (örneğin Türkçe) algılayarak anlaması, işlemesi, üretmesi ve yorumlamasını sağlayan yapay zekâ alt alanıdır. Temel hedefi insan dili ile makineler arasında anlamlı ve verimli bir iletişim kurmaktır. Özellikle üretim, kamu hizmetleri, e-ticaret ve yazılım gibi alanlarda otomatik sınıflandırma, etiketleme ve öneri üretme sistemlerinin temelini oluşturur.

Günümüzde büyük veri hacmi çok dilli içerik ve sektörel metinlerin karmaşıklığı göz önüne alındığında, NLP uygulamaları yalnızca istatistiksel yöntemlerle değil daha derin bağlamsal temsillere dayanan modellerle ele alınmaktadır.

2.1.1 NLP'nin tarihçesi ve evrimi

1950'li yıllarda ilk kural tabanlı sistemlerle başlayan NLP, 1990'larda istatistiksel modellere (n-gram, Hidden Markov Model vb.) yönelmiş, 2010 sonrası ise derin öğrenme devrimiyle önemli bir sıçrama yaşamıştır. Özellikle 2017'de önerilen Transformer mimarisi (Vaswani vd. 2017) bağlamsal anlamın modellenmesini mümkün kılarak NLP'yi sembolik analizden anlamsal temsile taşıyan bir değişim başlatmıştır.



Şekil 2.1 Transformer encoder-decoder yığını (Jalammar, 2018)

2.1.2 NLP görevleri ve bileşenleri

Bir NLP sistemi aşağıdaki temel görevleri yerine getirerek metni analiz eder:

Çizelge 2.1 Metinden embeddinge dönüşüm süreci

Görev	Açıklama
Tokenization	Metni kelime veya alt-kelimelere (subword) bölme
Stemming / Lemmatization	Kelimeleri kök veya sözlük formuna indirgeme
Part-of-Speech Tagging	Her kelimenin sözcük türünü (isim, fiil, sıfat vb.) belirleme
Named Entity Recognition (NER)	Özel adları (kişi, kurum, yer vb.) tanıma
Dependency Parsing	Kelimeler arası sentaktik ilişkileri belirleme
Sentiment Analysis	Duygu durumunu pozitif/negatif/nötr olarak analiz etme

Gelişmiş modeller bu görevleri bir arada yürütür. Örneğin bir kullanıcı "boru için kütük çelik" ifadesini girdiğinde sistem bunun metalurjik bir üretim girdisi olduğunu çözümleyerek doğru sınıfa yönlendirme yapabilir.

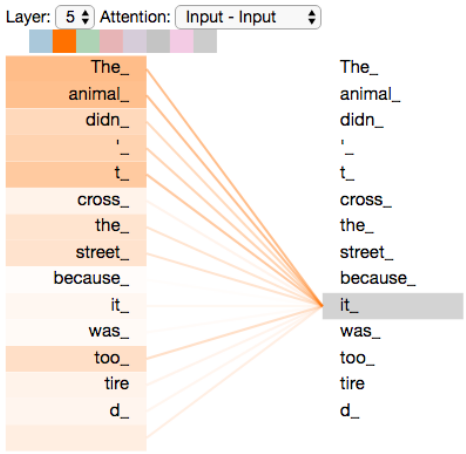
2.1.3 Embedding ve bağlamsal temsil

Embedding sistemleri kelimeleri veya cümleleri çok boyutlu sayısal vektörlere dönüştürerek anlamsal benzerlik ölçümlerini mümkün kılar. Örneğin "dikişsiz boru" ile "alaşımlı boru profili" ifadeleri kelime düzeyinde farklı görünse de embedding düzleminde birbirlerine yakın vektörlere sahiptir.

Şekil 2.2 Anlamsal benzerliğe dayalı embedding temsili (Jalammar, 2018)

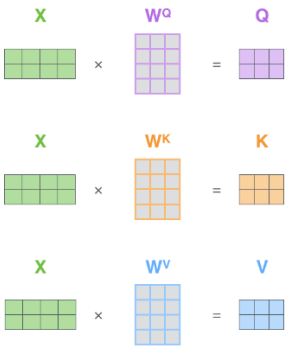
2.1.4 Transformer ve attention mekanizması

Transformer bağlamsal anlamı modelleyen bir yapıdır. Özünde "self-attention" mekanizması sayesinde bir kelimenin tüm diğer kelimelerle olan ilişkisini modelleyebilir. Bu sayede cümledeki anlam sadece kelimelere değil bağlama göre belirlenir (Vaswani vd. 2017).



Şekil 2.3 Self-attention katmanı görselleştirmesi (Jalammar, 2018)

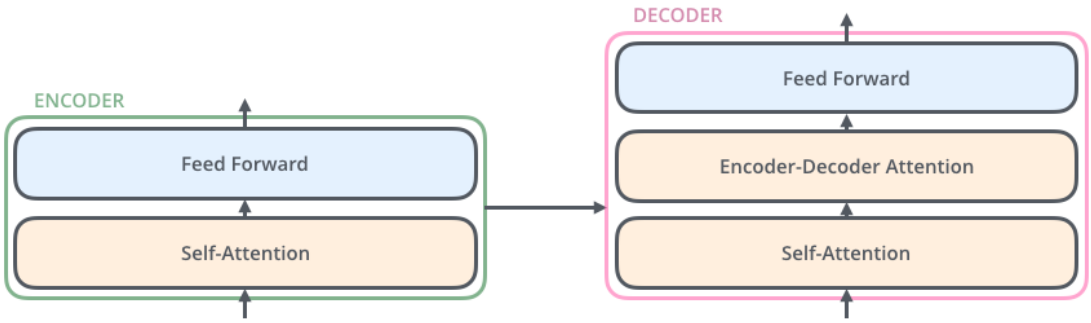
Self-attention mekanizmasında her kelime için üç ayrı vektör üretilir: Query (Q), Key (K) ve Value (V). Query ve Key vektörlerinin noktasal çarpımı, kelimeler arasındaki ilişkiyi (skor) verir. Bu skorlar, Key boyutunun karekökü ile ölçeklendirilir ve softmax fonksiyonuna sokularak olasılıklara dönüştürülür. Sonuçta, Value vektörleri bu olasılıklarla ağırlıklandırılır ve çıktı elde edilir.



Şekil 2.4 Self-attention skor hesaplaması (Jalammar, 2018)

2.1.5 Encoder ve decoder yapıları

Transformer mimarisi iki ana bloktan oluşur: Encoder ve Decoder. Encoder giriş metnini anlamlı bir temsile dönüştürür. Decoder ise bu temsili kullanarak çeviri, özetleme veya sınıflandırma gibi görevler üstlenir.



Şekil 2.5 Transformer encoder decoder yapısı (Jalammar, 2018)

2.2 Anlamsal Temsiller ve Embedding Sistemleri

Doğal dil işleme (NLP) sistemlerinin en temel ihtiyaçlarından biri doğal dildeki kelimeleri veya ifadeleri makine tarafından işlenebilir biçimlere dönüştürmektir. Bu bağlamda geliştirilen embedding sistemleri kelimeleri ve metinleri çok boyutlu sayısal vektörlerle temsil ederek bilgisayarların anlamsal benzerlikleri hesaplamasına olanak tanır. Bu vektör temsilleri bir ifadenin yalnızca dil bilimsel yapısını değil bağlamsal anlamını da yansıtır. Bu nedenle embedding sistemleri öneri motorları, anlamsal arama, belge kümeleme, bilgi çıkarımı ve çeviri gibi birçok NLP uygulamasının temel bileşeni haline gelmiştir (Mikolov vd. 2013, Pennington vd. 2014, Reimers & Gurevych 2019).

2.2.1 Dağıtımsal anlam teorisi ve kavramsal temel

Anlamsal temsillerin arkasındaki temel fikir dağıtımsal anlam teorisidir (distributional anlamsals). Bu teoriye göre “bir kelimenin anlamı çevresindeki diğer kelimelerle birlikte kullanıldığı bağlamda ortaya çıkar” (Firth, 1957). Bu anlayış geleneksel sözlük tabanlı yöntemlerden farklı olarak sözcüklerin kullanılma sıklıkları ve komşuluk ilişkileri

üzerinden anlamın türetilebileceğini savunur. Bu kuramsal çerçeve istatistiksel NLP modellerinin ve özellikle embedding sistemlerinin temelini oluşturur.

2.2.2 Word2Vec: Bağlamsal komşuluğa dayalı vektörler

2013 yılında Mikolov ve arkadaşları tarafından geliştirilen Word2Vec kelimelerin vektörel temsillerini üretmek üzere geliştirilen ilk başarılı yöntemlerden biridir. Bu modelin özünde benzer bağlamlarda kullanılan kelimelerin benzer vektör temsillere sahip olacağı varsayımı yatar (Mikolov vd. 2013). Word2Vec'in iki temel mimarisi bulunmaktadır:

- CBOW (Continuous Bag of Words): Bir bağlam içindeki kelimelerden hedef kelimeyi tahmin eder.
- Skip-Gram: Hedef kelimeden bağlamdaki kelimeleri tahmin eder.

Bu iki yaklaşım da yüksek boyutlu ve seyrek matrisleri düşük boyutlu yoğun vektörlere dönüştürerek anlamsal benzerlik hesaplarını mümkün kılar. Örneğin "çelik" ve "demir" gibi hammadeler sıkça birlikte kullanıldığı için vektör uzayında birbirine yakın yer alırlar.

2.2.3 GloVe: Küresel istatistiklerle derin anlamsal temsiller

Stanford Üniversitesi tarafından geliştirilen GloVe (Global Vectors for Word Representation) modeli kelimeler arası eşleşmeyi yalnızca bağlamsal komşulukla değil aynı zamanda kelime çiftlerinin küresel sıklık dağılımlarıyla kurar (Pennington vd. 2014). Bu yöntem, kelime eşleşmeleri için tüm veri kümesi boyunca elde edilen istatistikleri dikkate alır ve sözcük çiftleri arasındaki ilişkiyi logaritmik olasılık oranları üzerinden optimize eder. Sonuç olarak hem yerel hem de küresel bağlamın birlikte işlenmesine olanak tanır.

2.2.4 Cümle ve paragraf temsilleri: sentence embedding

Kelime düzeyindeki embedding yöntemleri kısa ifadelerde başarılı olsa da daha uzun ve kompleks yapılar olan cümle ve paragrafların temsilinde yetersiz kalabilmektedir. Bu ihtiyaca cevap vermek üzere geliştirilen modeller tüm bir cümleyi ya da metin parçasını vektörleştiren yöntemler sunmuştur. Bu alandaki önemli çalışmalardan biri Sentence-BERT modelidir.

Reimers ve Gurevych (2019) Transformer mimarisi üzerine kurulu olan BERT modelini yeniden yapılandırarak cümle düzeyinde anlamsal yakınlık hesaplamalarına uygun hale getirmiştir. Sentence-BERT iki cümle arasındaki benzerliği kosinüs benzerliği (cosine similarity) gibi ölçütlerle hesaplayabilecek sabit uzunlukta vektörler üretmektedir.

2.2.5 Transformer tabanlı embedding sistemleri

2017 yılında Vaswani ve arkadaşları tarafından geliştirilen Transformer mimarisi yalnızca makine çevirisi değil embedding üretiminde de paradigma değişimine neden olmuştur. Bu mimari self-attention mekanizması sayesinde her kelimenin bağlam içindeki rolünü öğrenebilir ve çok daha isabetli anlamsal temsiller oluşturur (Vaswani vd. 2017).

BERT, GPT, T5 gibi modern modeller bu mimariyi kullanarak dilin bağlam içi anlamını modelleyen yüksek kaliteli embedding vektörleri üretmektedir. Özellikle önceden eğitilmiş dil modelleri kullanılarak, anlamsal temsillerin doğruluğu önemli ölçüde artırılmıştır.

2.2.6 Anlamsal benzerlik ve vektör uzaylarında karşılaştırma

Anlamsal benzerlik iki dilsel ifadenin içerdiği anlamın sayısal olarak ne kadar örtüştüğünü belirlemek amacıyla kullanılan bir kriterdir. Bu benzerlik embedding sistemleri tarafından oluşturulan vektör temsilleri arasında yapılan karşılaştırmalarla

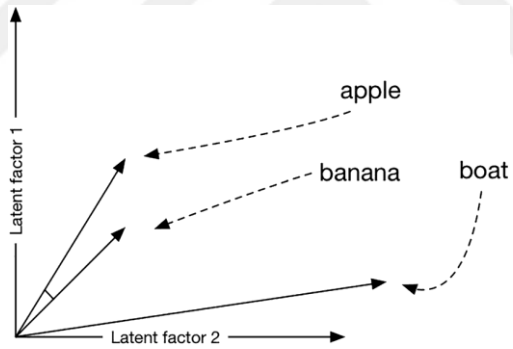
hesaplanır. NLP uygulamalarında kullanılan başlıca benzerlik ve uzaklık ölçüleri şunlardır:

Kosinüs Benzerliği (Cosine Similarity)

Kosinüs benzerliği iki vektörün birbirine yönsel yakınlığını ölçer. Vektörler aynı yönü gösteriyorsa değer 1'e, dik açı oluşturuyorsa 0'a, zıt yönlü ise -1'e yaklaşır. Anlam olarak yakın ama farklı uzunlukta kelime veya cümleler için idealdir çünkü sadece yön önemlidir, büyüklüğün bir önemi yoktur.

$$\text{cosine\_similarity}(\boldsymbol{A}, \boldsymbol{B}) = \frac{\boldsymbol{A} \cdot \boldsymbol{B}}{\|\boldsymbol{A}\| \cdot \|\boldsymbol{B}\|}$$ Denklem 2.1 Kosinüs benzerliği

Kosinüs benzerliği öneri sistemleri arama motorları ve cümle eşleştirme uygulamalarında sıkça tercih edilir. (Reimers & Gurevych, 2019)



Şekil 2.6 Kelimeler arasındaki kosinüs benzerliğinin vektör uzayındaki gösterimi (Wevers vd. 2020)

Öklidyen Mesafe (Euclidean Distance)

Öklidyen mesafe vektörler arasındaki gerçek geometrik uzaklığı ölçer. İki nokta arasındaki doğrusal mesafe gibi de ifade edilebilir. Formülü şu şekildedir:

$$\text{d}(\boldsymbol{A}, \boldsymbol{B}) = \sqrt{\sum_{i=1}^{n} (\boldsymbol{A_i} - \boldsymbol{B_i})^2}$$ Denklem 2.2 Öklidyen Mesafe

Bu yöntem anlamca benzer ama uzunluk farkı gösteren vektörler için yanıltıcı sonuçlar verebilir. Çünkü embedding vektörlerinin normları farklılık gösterebilir.

Noktasal Çarpım (Dot Product)

Dot-product iki vektörün çarpımıdır ve temel olarak kosinüs benzerliğiyle ilişkilidir. Ancak burada vektör büyüklükleri hesaba katılır:

$$\text{dot\_product}(\boldsymbol{A}, \boldsymbol{B}) = \sum_{i=1}^{n} \boldsymbol{A_i} \cdot \boldsymbol{B_i}$$ Denklem 2.3 Noktasal Çarpım

Dot-product büyük değerli vektörleri (örneğin çok sık görülen kelimeler) avantajlı kılabilir. Bu nedenle normalize edilmemiş embedding sistemlerinde yanlılık oluşturabilir. Bu yüzden modern modellerde genellikle normalize edilerek kullanılır (Devlin vd. 2019).

Manhattan Mesafesi (L1 Normu)

Manhattan mesafesi her boyuttaki farkların mutlak değerlerinin toplamını kullanır:

$$d_{L1}(\boldsymbol{A}, \boldsymbol{B}) = \sum_{i=1}^{n} |\boldsymbol{A_i} - \boldsymbol{B_i}|$$ Denklem 2-4 Manhattan mesafesi

Genellikle öklidyen mesafeye göre daha dayanıklı ve istatistiksel dağılımlarda daha az duyarlıdır. Ancak embedding vektörleri için pek tercih edilmez çünkü bağlamsal yakınlık yerine mutlak farklara odaklanır.

NLP alanında anlam temelli karşılaştırmalarda en sık tercih edilen yöntem kosinüs benzerliğidir çünkü vektörlerin büyüklüğünden bağımsız olarak yalnızca yönlerine odaklanır. Bu dilin bağlamsal yönünü daha doğru yansıtır (Mikolov vd. 2013; Reimers & Gurevych, 2019).

2.3 Endüstriyel Uygulamalarda Doğal Dil İşleme ve Büyük Dil Modelleri

Endüstriyel sistemlerde yapılandırılmış verinin yanında giderek artan şekilde yapılandırılmamış metin verileri de oluşmaktadır. Bu veriler; üretim formları, malzeme tanımları, sipariş notları, servis raporları gibi içeriklerdir. Bu tür belgeler klasik otomasyon sistemlerinin işleyemeyeceği biçimde doğal dile dayanmakta ve bu nedenle manuel yorumlama gerektirmektedir.

Geleneksel sınıflandırma sistemleri yapısal veri isterken bu belgelerin çoğu anlamsal düzeyde anlamlandırılmak zorundadır. Bu gereklilik endüstriyel süreçlerin dil modellemesi ile desteklenmesini zorunlu kılmıştır (Raza vd. 2025; Nature).

2.3.1 Endüstride NLP uygulama alanları

BERT (Devlin vd. 2019) ve GPT-3/4 (Brown vd. 2020) gibi LLM'ler bağlam temelli anlamsal temsiller üretme becerileri sayesinde endüstriyel süreçlerde büyük ilgi görmektedir. Bu modellerin sunduğu önemli avantajlar şunlardır:

- Önceden tanımlanmamış ürün ya da terimlerin anlamını bağlamdan çıkarma
- Anlamca yakın ancak yüzeysel olarak farklı ifadeleri eşleştirme
- Zero-shot veya few-shot öğrenme ile önceden etiketlenmemiş sınıflara tahmin üretme

Raza vd. (2025) büyük dil modellerinin sağlık, finans, üretim ve eğitim gibi farklı endüstriyel alanlarda kullanımını inceleyerek; hasta kayıtlarının otomatik analizi, finansal dolandırıcılık tespiti, müşteri hizmetleri otomasyonu, öngörücü bakım ve kalite kontrol gibi uygulamalarda önemli başarılar elde edildiğini raporlamaktadır. Bu çalışmalar LLM tabanlı yaklaşımların yalnızca akademik araştırmalarla sınırlı kalmadığını gerçek dünyadaki iş süreçlerinde de doğruluk, hız ve maliyet açısından dikkate değer iyileştirmeler sağladığını göstermektedir.

2.3.2 Ticari NLP uygulama platformları

Büyük teknoloji şirketleri ve kurumsal yazılım sağlayıcıları, NLP sistemlerini ürünlerine entegre etmektedir. Aşağıda bazı örnekler verilmiştir:

- SAP Leonardo & SAP AI Core: SAP belge sınıflandırma, doğal dil ile ürün sorgulama ve süreç analizi gibi işlevleri NLP ile otomatikleştirmektedir. Amazon Bedrock ile entegre sistemlerde LLM'ler BERT/GPT tabanlı olarak devreye alınmaktadır. (SAP News 2024)

- IBM Watson NLP: Watson platformu müşteri hizmetleri, sigorta değerlendirmesi ve hukuk metin analizi gibi sektörlere özel NLP çözümleri geliştirmiştir. GPT benzeri hibrit modelleri "Watson Assistant" üzerinden sunmaktadır. (IBM Research 2020)
- Amazon Comprehend: AWS bünyesindeki bu hizmet metinlerde konu, duygu, varlık, etiket ve ilişki tespiti yapabilmektedir. Özellikle ürün tanımları yorumlar ve destek talepleri üzerinde anlamsal analiz sağlamaktadır.

2.4 Otomasyon ve Veri Analitiği ile Büyük Dil Modellerinin (LLM) Entegrasyonu

Modern endüstriyel sistemler yalnızca otomasyonla değil aynı zamanda karar desteği ve anlam merkezli işlem katmanlarına ihtiyaç duymaktadır. Özellikle yapılandırılmamış metin verisinin (kullanıcı formları, teknik açıklamalar, iş akışı kayıtları) giderek artması klasik kurallara dayalı otomasyon çözümlerinin yetersiz kalmasına yol açmıştır. Bu noktada büyük dil modelleri (LLM) bağlam anlayışına sahip güçlü temsil yetenekleri sayesinde otomasyonun anlamsal bileşeni haline gelmektedir.

2.4.1 Klasik otomasyon sistemlerinin yetersizliği

Klasik otomasyon sistemleri, önceden tanımlanmış kurallara ve yapılandırılmış girdilere dayanır. Ancak gerçek endüstriyel ortamlarda karşılaşılan metinler genellikle serbest biçimli, eksik, çelişkili ya da bağlamsal olarak yorum gerektiren yapılardadır. Bu durum, özellikle belge işleme, sınıflandırma ve öneri sistemlerinde ciddi hatalara yol açmaktadır.

Örneğin üretim tesislerinde kalite kontrol formlarındaki yorumlar yalnızca belirli anahtar kelimelerle değil bağlam içinde değerlendirildiğinde anlam kazanmaktadır. Aynı şekilde malzeme kodlama sistemleri de doğru sınıflandırma için yüzeysel metin eşleşmesinden daha fazlasını gerektirir.

2.4.2 Büyük dil modelleri ile otomasyonun zenginleştirilmesi

Transformer mimarisiyle geliştirilen LLM'ler klasik otomasyon sistemlerine üç temel alanda zenginlik katmaktadır:

Anlamsal Veri Temsili: LLM'ler metinleri yüzeysel kelime dizilerinden çok boyutlu anlamsal vektörlere dönüştürerek temsil eder. Bu yaklaşım daha önce görülmemiş ifadeleri bile bağlama dayanarak yorumlayabilme imkânı sağlar. (Roussinov, 2023; Raffel vd., 2020)

Karar Destek Sistemleri: LLM'ler yalnızca metni tanımlamakla kalmaz, aynı zamanda bu metne tavsiye üreten yapay uzmanlar gibi davranabilir. Örneğin, "bu tanım şu koda daha uygundur" şeklinde yönlendirme yapabilir. (Guntupalli & Watanabe, 2024)

İnsan Denetimli Öğrenme (Human-in-the-Loop): LLM tabanlı sistemlerde önerilen çıktılar uzman doğrulamasına sunularak model performansı sürekli iyileştirilebilir. Bu yapı, klasik kural tabanlı otomasyon sistemlerinin aksine, esneklik ve sürekli öğrenme imkânı sunar. Reciprocal Human-Machine Learning gibi yaklaşımlar, hibrit insan-AI işbirliği modelleri önermektedir. (Te'eni vd., 2023)

2.4.3 Büyük dil modelleri ile otomasyon sistemlerinin entegrasyonu

Büyük dil modellerinin otomasyon sistemlerine entegrasyonu için tipik bir mimari yapı şu adımları içerir:

1. Veri Toplama: Kullanıcı girişleri, form verileri, işlem metinleri
2. Temizleme ve Önişleme: Gereksiz kelimelerin ayıklanması, metnin normalize edilmesi
3. Embedding Çıkarımı: LLM kullanılarak bağlam temelli vektör oluşturulması
4. Anlamsal Karar: Kod önerisi, kategori eşleşmesi, belge sınıflaması gibi görevlerin yerine getirilmesi
5. İşlem Aktarımı: ERP/MES gibi sistemlere yönlendirme
6. Geri Bildirim: Kullanıcı onayı ile modelin yeniden öğrenmesi

2.4.4 Uygulama alanları ve kurumsal örnekler

SAP & Amazon Bedrock: Figueiredo (2025), SAP Leonardo'nun Amazon Bedrock ile entegre kullanımı sonucu üretim sistemlerinde kod eşlemesi ve sipariş işleme süreçlerinin otomatik hale getirildiğini aktarmaktadır. LLM, bu süreçte kullanıcı girdilerini anlamsal analizden geçirerek SAP sistemine uyumlu sınıflamalar önermektedir.

IBM Watson NLP: Watson NLP, özellikle sözleşme analizi ve müşteri belgelerinin ön sınıflandırması gibi karmaşık görevlerde kullanılmaktadır. 2024 raporunda IBM, Watson Assistant'ın %87 doğrulukla belge sınıfı tespiti yaptığını bildirmiştir.

AutoML + LLM Sistemleri: Bello (2024), AutoML altyapısı üzerinde LLM destekli yönlendirme sistemi kurarak, kullanıcıların hangi makine öğrenimi modelini seçmesi gerektiğini LLM üzerinden belirlemiştir. Sistem, açıklamaları doğal dilde yaparak süreci teknik olmayan personel için erişilebilir hale getirmiştir.

Arıza Teşhis Ağaçları (FMEA): Vidyaratne vd. (2025), üretim hattındaki arızaları çözmek için geliştirilen LLM tabanlı sistemlerin FMEA (Failure Mode and Effects Analysis) ağaçlarını otomatik oluşturabildiğini göstermiştir. Sistem, metinsel arıza kayıtlarından neden-sonuç ilişkilerini anlamsal olarak kurarak mühendis önerileri üretmektedir.

2.5 Literatürdeki Mevcut Modellerin İncelenmesi

Doğal dil işleme (NLP) alanında öneri sistemleri ve metin sınıflandırma modelleri son yıllarda anlamsal gömme (anlamsal embedding) tabanlı temsillere dayalı olarak büyük bir evrim geçirmiştir. Geleneksel olarak kelime sıklığı ve n-gram gibi yüzeysel istatistiksel özelliklere dayanan sistemlerin yerini bağlamsal temsillere odaklanan derin öğrenme tabanlı modeller almıştır. Bu gelişim metin içeriklerinin çok boyutlu ve anlam odaklı bir biçimde temsil edilmesini mümkün kılarak öneri sistemlerinde ve sınıflandırma algoritmalarında doğruluk oranlarını önemli ölçüde artırmıştır.

Anlamsal embedding modelleri yalnızca kelime düzeyinde değil tümce, paragraf ve belge düzeyinde bağlamsal temsiller üretebilme kapasiteleri sayesinde bilgi madenciliği, otomatik etiketleme, bilgi erişimi ve kavramsal ilişki çözümleme gibi alanlarda etkili bir araç hâline gelmiştir. Literatürde bu alanda geliştirilen yöntemlerin büyük bölümü Transformer mimarisi üzerine kurulmuş modellerin çeşitli türevlerini içermektedir. Bu bölümde özellikle anlamsal embedding kullanılarak geliştirilen sınıflandırma sistemleri, akademik öneri sistemleri ve bu sistemlerin ortak mimari yapı taşları detaylı biçimde incelenmektedir.

2.5.1 Anlamsal embedding tabanlı sınıflandırma modelleri

Büyük Metin sınıflandırma doğal dil işlemenin en köklü problemlerinden biridir. Geleneksel yöntemler, bag-of-words ya da TF-IDF gibi kelime sıklığı temelli vektör uzaylarını kullanarak sınıflama işlemlerini yürütmekteydi. Ancak bu yöntemler kelimeler arasındaki anlamsal bağları göz ardı etmeleri nedeniyle bağlamsal anlamı yansıtmakta yetersiz kalmaktadır. Modern sistemlerde bu eksiklik kelime ve cümle temsillerinin embedding vektörleri ile giderilmektedir.

BERT (Devlin vd. 2019) gibi önceden eğitilmiş bağlamsal dil modelleri bir kelimenin anlamını içinde bulunduğu cümle bağlamına göre dinamik olarak yeniden tanımlar. Örneğin, “bank” kelimesi bir dış mobilya anlamında mı yoksa finansal kurum anlamında mı kullanılıyor sorusuna model cümle bağlamını okuyarak cevap verebilir. Bu bağlamsal ayırt edicilik klasik yöntemlerin ötesinde anlamsal ayrımları yakalamayı mümkün kılar.

Bu bağlamsal temsiller genellikle bir sinir ağı sınıflayıcısı ile entegre edilerek metnin sınıfına dair karar verilir. Logistic regression, softmax katmanı gibi tekniklerle birlikte kullanılan bu temsiller özellikle resmi belge sınıflaması, hukuki metin ayrıştırma gibi yüksek hassasiyet gerektiren uygulamalarda başarılı sonuçlar üretmektedir.

Gonzalez-Gomez vd. (2025) çalışmasında embeddingler bir Named Entity Recognition (NER) ve Relation Extraction (RE) işlem hattı ile entegre edilmiştir. Bu yapı özellikle bilgi yapılarının çıkarımı ve gelecekte talep görecek beceri ve yetkinliklerin

sınıflandırılması gibi ileri düzey uygulamalarda kullanılmıştır. Araştırmada geliştirilen akış NER modeli ile "beceri" ve "meslek" gibi KSA (Knowledge, Skills, Abilities) varlıklarını tanımlarken, RE modeli bu varlıklar arasındaki anlamsal ilişkileri yakalayarak dinamik bir sınıflandırma oluşturulmasını sağlamıştır. Modelin NER performansı F1 skoru = 65,38 % ilişki çıkarımı için elde edilen F1 skoru = 82,2 % olarak rapor edilmiştir. Bu sonuçlar embedding temelli yaklaşımların yalnızca sınıflandırma değil aynı zamanda üst düzey anlamsal bilgi çıkarımı ve yapısal analiz görevlerinde de başarılı biçimde kullanılabileceğini göstermesi açısından önemlidir.

2.5.2 Ortak mimariler ve vektör uzayı üzerinde karşılaştırma yaklaşımları

Sınıflandırma ve öneri sistemleri belirli bir mimari şablonu izler: embedding → vektör benzerliği → karar/sıralama. Bu yapı esneklik ve doğruluk arasında optimal bir denge sağlar.

Embedding katmanı kelime veya tümce temelli anlamsal temsiller üretmek üzere önceden eğitilmiş modelleri kullanır. Word2Vec ve GloVe gibi modeller kelime düzeyinde sabit embedding temsilleri sağlarken BERT ve Sentence-BERT gibi modeller bağlamsal temsiller sunar. Özellikle Sentence-BERT metin benzerliğini yüksek doğrulukla ölçebildiği için öneri sistemlerinde yaygın biçimde kullanılmaktadır (Reimers & Gurevych, 2019).

İkinci aşama olan benzerlik hesaplayıcı, embedding vektörleri arasındaki mesafeyi ölçer. En yaygın kullanılan metriklerden biri kosinüs benzerliğidir. Bu metrik vektörlerin yönelimi üzerinden anlamsal yakınlığı hesaplar. 1'e yakın değerler, iki metnin anlam bakımından oldukça benzer olduğunu gösterir.

Son katman ise karar veya sıralama yapısını içerir. Bazı sistemler embedding vektörlerini sınıflayıcı bir modele (ör. Logistic Regression) verirken bazı sistemler top-k en benzer vektörleri seçerek sıralı bir öneri listesi oluşturur. Bu aşamada embedding modelinin domain ile ne derece uyumlu olduğu kritik bir başarı belirleyicidir.

Bu yapı akademik içerik önerisinden ürün tavsiyesine, hukuki belge sınıflamasından medikal teşhis önerilerine kadar pek çok alana uygulanabilir esneklikte tasarlanmıştır. Literatürde bu ortak mimari yapı, birçok modern NLP sisteminin temelini oluşturmaktadır.

3. MATERYAL VE YÖNTEM

3.1 Çalışma Amacı ve Yaklaşımı

Endüstriyel üretim süreçlerinde kullanılan dijital sistemlerin kurumsal düzeyde anlamlı ve standardize edilmiş veriyle beslenmesi, kamu politikaları, ekonomik planlama, mali denetim ve uluslararası karşılaştırmalar açısından hayati öneme sahiptir. Türkiye'de bu gereksinim, işletmelerin faaliyetlerini sınıflandırmak amacıyla kullanılan PRODTR (Ürün Sınıflama) sistemi gibi düzenlemelerle karşılanmaktadır. Bu sistem şirketlerin girdilerini, çıktıları ve üretim faaliyetlerini belirli kodlar ve tanımlar aracılığıyla resmi olarak sınıflandırmalarını zorunlu kılar.

Ancak mevcut durumda söz konusu sınıflandırma sistemlerinin kapsamlı ve teknik yapısı kullanıcılar için oldukça karmaşık bir hale gelmiştir. Özellikle KOBİ düzeyindeki işletmeler bu tür sınıflandırmaları doğru bir şekilde anlamlandırmakta ve uygun kodları seçmekte zorlanmakta bunun sonucu olarak da sistematik hatalar, eksik kodlamalar oluşmaktadır. Bu durum yalnızca bireysel raporlamaları değil daha büyük ölçekte kamu istatistiklerini ekonomik analizleri ve makro planlamaları da etkilemektedir.

Bu yüksek lisans tezinin temel amacı bu karmaşıklığı gidermek ve sınıflandırma süreçlerini anlamsal düzeyde desteklemek üzere doğal dil işleme (NLP) teknikleri ile çalışan bir öneri sistemi geliştirmektir. Geliştirilen sistem işletmelerin serbest biçimde ifade ettikleri üretim tanımlarını analiz ederek bunlara en uygun resmi sınıflandırma kodlarını önerir. Bu öneriler yalnızca kelime düzeyinde benzerlik değil bağlam içi anlam ilişkileri de göz önünde bulundurularak yapılır. Böylece sistem kullanıcıların teknik uzmanlık gerektiren kodlama süreçlerini daha kolay ve daha isabetli yürütmelerine olanak tanır.

Bu yaklaşımın merkezinde güncel büyük dil modeli (LLM) teknolojilerine dayalı olarak çalışan bir embedding mekanizması bulunmaktadır. Kullanıcı tarafından girilen ifadeler bu dil modelleri aracılığıyla anlamsal vektörlere dönüştürülür. Aynı şekilde, resmi

sınıflandırma sistemine ait tanımlar da benzer biçimde vektörleştirilir. Elde edilen bu yüksek boyutlu vektör temsilleri benzerlik ölçütleri aracılığıyla karşılaştırılarak anlamsal olarak en yakın olan resmi tanımlar sistem tarafından öneri olarak sunulur.

Bu yapı yalnızca öneri yapmakla kalmaz aynı zamanda kullanıcıyı anlamaya çalışan ve bağlamın içeriğini çözen bir anlamsal eşleştirme sistemi olarak da işlev görür. Örneğin kullanıcı "yüksek karbonlu çelik kütük" gibi teknik bir ifade yazdığında, sistem bu ifadenin yalnızca yüzeysel anahtar kelimelerini değil içerdiği sektörel bağlamı da hesaba katar. Böylece önerilen kodlar, resmi sınıflandırma sistemi içinde anlamsal açıdan en uygun tanımlara yönlendirilmiş olur.

Geliştirilen sistemin bir diğer önemli özelliği de veri merkezli ve modüler bir mimariye sahip olmasıdır. Sistem yalnızca bir kod tablosuna dayalı eşleştirme yapmaktan öte; önce geniş veri kümelerini analiz eden ardından bu verileri ön işleme adımlarıyla normalize eden ve son olarak bu verilerden öğrenilmiş anlamsal temsillerle öneri yapan çok aşamalı bir yapıya sahiptir. Bu modülerlik sistemin kolayca güncellenebilmesine, farklı sektörlere uyarlanabilmesine ve farklı sınıflandırma sistemlerine (örneğin GTİP, NACE) entegre edilebilmesine olanak tanımaktadır.

Bu tezde ele alınan öneri sistemi sadece teorik bir model olmaktan çok üretime yakın bir prototip olarak da geliştirilmiştir. Sistem hem etkileşimli hem de otomatik kullanım senaryoları için uygundur. Kullanıcı bir web arayüzü veya form sistemi aracılığıyla metinsel tanımını girdiğinde sistem otomatik olarak anlamsal analiz gerçekleştirerek önerileri geri döndürmektedir.

Sistemde embedding üretimi bulut temelli modeller aracılığıyla gerçekleştirilmiştir. Bu sayede sistem karmaşık dil yapılarını anlamlandırabilmekte ve farklı dil düzeylerinden gelen girdilere karşı esnek davranabilmektedir. Dil modeli tarafından üretilen vektör temsiller arama motorları için optimize edilmiş bir altyapı üzerinde saklanmakta ve gerçek zamanlı olarak sorgulanabilmektedir. Bu yapı önerilerin hızlı ve güvenilir şekilde sunulmasını garanti altına almaktadır.

Sistem eğitimli kullanıcılar için olduğu kadar, teknik uzmanlığı sınırlı kullanıcılar için de etkin bir araç sunar. Bu yönüyle hem kamu kurumlarının veri doğrulama süreçlerinde, hem de özel sektör firmalarının raporlama ve başvuru işlemlerinde akıllı yardımcı rolü üstlenebilir. Sistem ayrıca kurumsal bilgi yönetimi politikalarının dijitalleşme süreçlerinde aktif olarak yer alabilecek şekilde tasarlanmıştır.

Sonuç olarak bu çalışmanın temel amacı yalnızca doğru sınıflandırma yapmak değil aynı zamanda kurumsal veri girişlerini dil modelleriyle zenginleştirerek karar destek sistemlerine temel sağlayacak şekilde akıllandırmaktır.

3.2 Veri Kümesi ve Sözlük Yapıları

Bu çalışmada geliştirilen öneri sisteminin temelini, Türkiye'deki üretim ve hizmet faaliyetlerini sistematik biçimde tanımlamak amacıyla oluşturulan PRODTR (Ürün Sınıflama Sistemi) oluşturmaktadır. PRODTR, resmi bir sınıflandırma sistemi olup, her bir ürün veya üretim çıktısı çok seviyeli bir kodlama yapısı ve detaylı tanımlarla ifade edilmektedir.

Referans veri kümesine, md.teyit.org (2020) sitesi üzerinden erişilen "Yıllık Sanayi Ürün İstatistikleri Kod Listesi (PRODTR 2017–2018)" adlı PDF dokümanı esas alınmıştır. Söz konusu PDF, Excel formatına dönüştürülerek yalnızca PRODTR kodları ayıklanmış ve böylece referans veri kümesi oluşturulmuştur.

PRODTR sistemi yalnızca ürünleri ana başlıklar altında toplamakla kalmayıp aynı ürünün varyantlarını ve fiziksel, kimyasal veya işlenmişlik düzeyine göre ayrılmış alt tiplerini de tanımlar. Örneğin “taşkömürü” yalnızca tek bir başlık olarak değil “tuvenan”, “ayıklanmış (parça)”, “yıkanmış (toz)” gibi alt kategorilerle ayrıştırılmış şekilde sistemde yer alır. Aynı şekilde “buhar kömürü” gibi enerji sektöründe kritik öneme sahip ürünler farklı kalori değerleri, kullanım amacı veya fiziksel hâline göre detaylandırılır. Bu düzeyde ayrım sınıflandırma sistemini oldukça detaylı ve teknik hale getirir.

Bu yapı öneri sisteminin başarısı için önemli bir zorluk oluşturur. Çünkü kullanıcı tarafından girilen metinlerin çoğu bu resmi yapıya birebir uymamakta teknik terim eksiklikleri, genel ifadeler veya bağlamsal sapmalar içerebilmektedir. Sistem bu serbest metinleri alarak ilgili PRODTR tanımlarıyla anlamsal eşleşme yapmak bir zorunluluk haline gelmiştir.

Sistemin kullandığı referans veri kümesi PRODTR sisteminde yer alan ürün tanımlarının tümünü kapsamaktadır. Bu tanımlar kod - tanım yapısına sahip olup; her biri, bir ürünün yalnızca ne olduğunu değil aynı zamanda ne için kullanıldığını da açıklar. Tanımlar genellikle şu unsurları içerir:

Kod: Örneğin 05.10.10.30.01, sistemde yer alan ürünün hiyerarşik konumunu gösterir.
Tanım: "Taşkömürü – Tuvenan" gibi başlıklar ve Brüt kalori değeri, kullanım amacı, üretim biçimi gibi detayları içeren tanımları kapsar.

Çizelge 3.1 PRODTR Ürün örnekleri

Kod	Tanım
05.10.10.30.01	Taşkömürü - Tuvenan (Brüt Kalori Değeri > 23,865 kj/kg olan kok üretimine olanak sağlayan maden kömürü)
05.10.10.30.02	Taşkömürü - Ayıklanmış (parça) (Brüt Kalori Değeri > 23,865 kj/kg olan kok üretimine olanak sağlayan maden kömürü)
05.10.10.30.03	Taşkömürü - Yıkanmış (parça) (Brüt Kalori Değeri > 23,865 kj/kg olan kok üretimine olanak sağlayan maden kömürü)
05.10.10.30.04	Taşkömürü - Ayıklanmış (toz) (Brüt Kalori Değeri > 23,865 kj/kg olan kok üretimine olanak sağlayan maden kömürü)
05.10.10.30.06	Taşkömürü - Şlam (Brüt Kalori Değeri > 23,865 kj/kg olan kok üretimine olanak sağlayan maden kömürü)
05.10.10.50.00	Buhar kömürü - (Steam coal) (Brüt Kalori Değeri > 23,865 kj/kg olan, ıslak numunesi külsüz, buhar üretmek ve yer ısıtma a
05.20.10.30.01	Linyit - Tuvenan (Brüt kalori değeri < 23,865 kj/kg olan)

Kullanıcı tarafındaki metinler ise çoğunlukla bu yapının dışındadır. Girdi olarak "linyit kömürü", "yüksek kalori buhar kömürü" gibi ifadelerle karşılaşılabilir. Bu ifadeler, doğrudan kodla eşleşmese de anlam olarak sistemdeki resmi tanımlara yakınlık gösterir. Dolayısıyla sistemin temel işlevi bu serbest ifadeleri embedding vektör temsillerine çevirip, bunları referans kümedeki tanımlarla karşılaştırmaktır.

Bu sistem yalnızca doğrudan eşleşme yapmakla kalmaz aynı zamanda çok benzer tanımları da sıralı olarak kullanıcıya sunar. Örneğin "buhar kömürü" ifadesiyle sistem sorgulandığında, PRODTR'deki 05.10.10.50.00 kodlu "Buhar kömürü – Steam coal" tanımına yönlendirme yapılır. Ancak aynı zamanda kalori değeri benzer başka ürünler de listelenerek alternatif seçenekler sunulur.

Sistem aynı zamanda öneri çeşitliliğini ve esnekliğini artırmak için, resmi sınıflandırmadaki ürün varyasyonlarını dikkate alan bir yapıya sahiptir. "Taşkömürü – Ayıklanmış (parça)" ve "Taşkömürü – Şlam" gibi varyasyonlar tekil bir ana ürün başlığı altında gruplanmaz her biri kendi başına tanımlanır ve sistem tarafından ayrı bir anlamsal nokta olarak değerlendirilir.

3.3 Ön İşleme Adımları

Bir doğal dil işleme (NLP) tabanlı sistemde metinlerin model ile etkileşime geçebilmesi büyük ölçüde önişleme (preprocessing) süreçlerinin etkinliğine bağlıdır. Girdi metinlerinin ham hâlleri, yapısal tutarsızlıklar, noktalama farkları, büyük/küçük harf kullanımı, yazım hataları ve bağlamsal belirsizlikler nedeniyle doğrudan işlenemez hâlde olabilir. Bu nedenle sistemin başarısı yalnızca kullanılan embedding modelinin kapasitesine değil aynı zamanda bu modelle verimli iletişim kurulmasını sağlayan ön hazırlık adımlarına da sıkı sıkıya bağlıdır.

Bu çalışmada, kullanıcı tarafından serbest şekilde girilen üretim veya hizmet tanımları ile Türkiye Cumhuriyeti Sanayi ve Teknoloji Bakanlığı tarafından yayımlanmış PRODTR sistemi üzerindeki resmi tanımlar benzer bir önişleme hattından geçirilerek embedding

işlemlerine hazır hâle getirilmiştir. Böylece her iki veri kümesinin aynı düzlemde karşılaştırılabilir olması sağlanmıştır.

İlk adımda uygulanan işlemler metinsel verilerin sadeleştirilmesini ve model açısından ayırt edici olmayan ögelerden arındırılmasını amaçlamaktadır. Bu kapsamda tüm metinler küçük harfe dönüştürülmüş Türkçe'de yaygın olarak rastlanan büyük/küçük harf farklarından kaynaklı anlamsal sapmaların önüne geçilmiştir. Aynı zamanda fazlalık oluşturan boşluklar silinmiş ve özel karakterler sadeleştirilmiştir.

Yazım birliğinin sağlanması açısından bazı kurallar belirlenmiş ve serbest metinlerin bu kurallara göre normalize edilmesi sağlanmıştır. Örneğin "çelik-döküm", "çelik döküm." veya "ÇELİK DÖKÜM" gibi ifadelerin tümü tekil bir forma dönüştürülmüştür. Bu sayede, yüzeysel olarak farklı ancak anlamsal olarak özdeş içeriklerin embedding sürecinde çakışmaması ve vektör uzayında dağılmaması garanti altına alınmıştır.

Serbest metinlerin daha anlamlı biçimde embedding modeline aktarılabilmesi için özel bir biçimlendirme stratejisi geliştirilmiştir. Girdi metni "ürünadı: {productName} teknik adı: {technicalName}" formatında düzenlenmiştir. Bu yapı sayesinde model her iki kavramsal alanı bağlam içinde birlikte değerlendirebilmekte ve teknik isimlendirmelerle ürün adları arasındaki ilişkiyi daha doğru biçimde kurabilmektedir. Özellikle üretim terminolojisinde sık karşılaşılan eşanlamlı veya yakın anlamlı terimlerin ayrımı bu tür yapılandırılmış formatlama sayesinde sağlanmaktadır.

Sistemde embedding işlemleri OpenAI tarafından geliştirilen text-embedding-3-small modeli kullanılarak gerçekleştirilmiştir. Bu model çok dilli destek sağlayabilen ve anlamsal yakınlık temelli öneri sistemleri için optimize edilmiş bir yapay sinir ağı modelidir. Modelin temel işlevi verilen her bir metni anlam düzeyinde temsil edebilecek yüksek boyutlu bir vektör uzayına dönüştürmektir. Bu işlem sonucunda her bir tanım bağlam ve içerik özelliklerini yansıtan vektör haline gelir.

Önişleme süreci yalnızca temizleme ve yapılandırma ile sınırlı kalmamış aynı zamanda Türkçe diline özgü sorunlara karşı da önlemler alınmıştır. Her ne kadar embedding modeli doğrudan Türkçe eğitimi almış bir yapay zekâ modeli olmasa da bu dildeki örüntüleri tanıma kapasitesine sahiptir. Bu nedenle sistem Türkçe metinleri ham hâlde işleyebilmekte ve bağlamsal ilişkileri çıkarabilmektedir. Buna rağmen, modelin performansını artırmak amacıyla Türkçe'ye özgü dil bilgisel dönüşümler (örneğin "-lik", "-sal", "-cı" eklerinin etkileri) göz önünde bulundurularak ön analizler yapılmıştır.

Token sınırlaması nedeniyle çok uzun açıklamalar, modele tek parça hâlinde gönderilemeyecek düzeyde olsaydı belirli bir kırpma veya bölme stratejisi uygulanabilirdi. Ancak PRODTR sistemi tanımlarının çoğu bu sınırların altında kalacak düzeyde tanımlar içerdiğinden, ek bir bölme adımına ihtiyaç duyulmamıştır.

Embedding çıktıları JSON formatına dönüştürülerek Elasticsearch üzerinde vektörel veri dizini (vector index) şeklinde saklanmaktadır. Elasticsearch'ün dense_vector tipi ve kNN search özelliği, bu tür yüksek boyutlu vektörleri indeksleme ve hızlı karşılaştırma işlemleri için uygundur. Bu yapı sayesinde öneri sistemi çok büyük sayıda referans tanımı barındırsa bile sorgu anında benzerlik derecesi yüksek olan girdileri anlık olarak bulup sıralayabilmektedir. Benzerlik hesaplama süreci Elasticsearch'ün kNN API'siyle gerçekleştirilmekte ve varsayılan olarak kosinüs benzerliği metriğini kullanmaktadır.

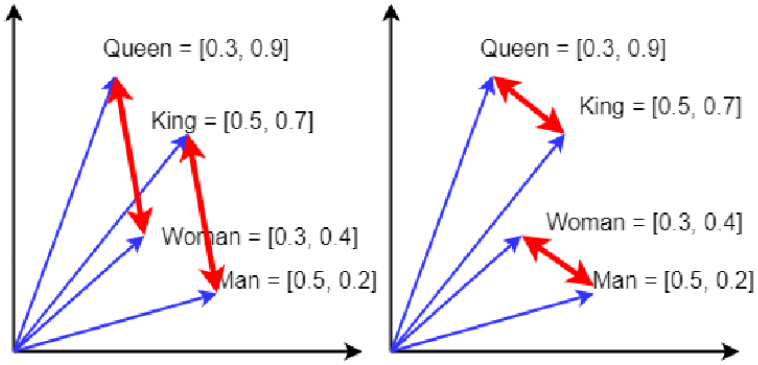
Sistemin son kullanıcıya sunduğu öneriler embedding vektörleri arasındaki kosinüs benzerliği üzerinden sıralanmakta ve her bir öneri bir benzerlik katsayısı ile gösterilmektedir. Bu sayede kullanıcı yalnızca en yakın eşleşmeyi değil anlamca benzer olabilecek alternatifleri de görebilmektedir.

3.4 Benzerlik Hesaplama

Sistemde kullanıcıdan gelen sorgular "ürünadı" ve "teknik adı" bileşenlerinden oluşur. Bu metinler birleştirilerek embedding modeline gönderilir. Modelin çıktısı anlam bütünlüğünü koruyarak çok boyutlu bir vektör uzayındaki karşılığını üretir. Aynı embedding işlemi daha önceden önişlemden geçirilmiş tüm PRODTR tanımlarına da uygulanmıştır. Böylece sistem her iki taraf için ortak bir temsili uzay oluşturmuş olur.

Vektörel uzayda her bir metin çok boyutlu bir noktayı temsil eder. Bu noktalar arasındaki mesafe ya da açı metinler arasındaki anlamsal yakınlığı belirler. Bu çalışmada, vektörler arası benzerlik hesaplamaları için Elasticsearch'ün sağladığı kNN (k-Nearest Neighbors) tabanlı arama altyapısı kullanılmıştır. Elasticsearch her metin embedding'ini dense_vector alanlarında saklamakta ve sorgu anında benzer vektörleri hızlıca bulmaktadır. Benzerlik ölçümünde sistem varsayılan olarak kosinüs benzerliği metriğini kullanır.

Çizelge 3.2 Kral - erkek + kadın ≈ kraliçe örneği. word embeddinglerin vektör uzayındaki temsillerine bir örnek. (Sutor vd. 2019)



Sorgu anında kullanıcıdan alınan metin aynı embedding sürecinden geçirilerek bir sorgu vektörü elde edilir. Bu vektör Elasticsearch üzerinde kayıtlı tüm embedding'lerle karşılaştırılarak en benzer k tanımı döndüren bir arama işlemi başlatır. k değeri 10 olarak seçilmiştir böylece sistem yalnızca en yüksek benzerliği değil alternatif eşleşmeleri de listeleyebilir.

Sistemin bu şekilde çalışması öneri sürecini yalnızca yüzeysel anahtar kelime eşleşmelerine indirgemekten çıkarıp, bağlamsal eşleşmelere dayalı çok boyutlu bir anlam karşılaştırması hâline getirir. Örneğin "yüksek kalorili taşkömürü" ifadesiyle yapılan bir sorguda sistem, yalnızca literal olarak "taşkömürü" geçen girdileri değil "Brüt Kalori Değeri > 23,865 kj/kg olan kok üretimine olanak sağlayan maden kömürü" şeklinde resmi tanımları da doğru biçimde eşleştirebilir.

Çizelge 3.3 Sorgu-tanım eşleşme örneği

Ürün_Adı	Teknik_Ad	Bulunan_Kod	Bulunan_Tanım	Benzerlik	Doğru_Kod	Doğru_Tanım	Eşleşme
Çanlar	Çanlar	25.99.29.82.00	Çanlar, gonglar, vb. (elektrikli olmayan), adi metallerden	0,84315	25.99.29.82.00	Çanlar, gonglar, vb. (elektrikli olmayan), adi metallerden	DOĞRU
Çanlar	Çanlar	25.73.10.10.00	Beller ve kürekler	0,8005			
Çanlar	Çanlar	25.73.10.30.00	Kazmalar, külünkler, çapalar, tırmıklar	0,79706			
Çanlar	Çanlar	20.52.10.20.00	Tutkallar, kazeinden olanlar	0,79487			
Çanlar	Çanlar	10.89.11.00.01	Çorbalar ve bunların müstahzarları	0,79106			
Çanlar	Çanlar	23.14.12.50.00	Ağlar, keçeler, şilteler ve paneller, dokusuz cam elyafından	0,78808			
Çanlar	Çanlar	08.12.12.30.01	Kırmataş	0,78686			
Çanlar	Çanlar	17.22.12.40.00	Vatkalar; vatkadan diğer eşyalar	0,78392			
Çanlar	Çanlar	10.51.40.30.02	Lor ve çökelek	0,78146			
Çanlar	Çanlar	10.84.12.70.02	Çemen	0,78134			
Çanlar	Çanlar	28.30.32.30.00	Tırmıklar (diskarolar hariç)	0,78068			
Çanlar	Çanlar	13.92.22.50.00	Yelkenler	0,77841			
Çanlar	Çanlar	20.41.43.83.00	Metal cilaları	0,77643			
Çanlar	Çanlar	13.93.11.00.03	El halıları, yünden (düğümlü olanlar)	0,77564			
Çanlar	Çanlar	28.30.31.40.00	Pulluklar	0,77547			

Elde edilen eşleşmeler kullanıcıya sıralı olarak sunulmaktadır. Önerilerin her biri ilgili PRODTR tanımının kodu ve ürün tanımı ile kullanıcıya gösterilir. Bu yapı kullanıcıya hem hızlı karar alma imkânı sağlar hem de alternatifleri değerlendirerek kendi tanımını daha netleştirme fırsatı sunar.

Bu mimarinin bir başka avantajı da sistemin sürekli güncellenebilir olmasıdır. Embedding uzayı statik değildir yeni ürün tanımları eklendikçe embedding'leri yeniden hesaplanıp dizine eklenebilir. Aynı şekilde OpenAI embedding modelleri güncellendiğinde sistemin embedding süreci güncellenerek kalite artışı sağlanabilir.

Özetle embedding süreci sistemin en kritik yapılarından biridir ve yalnızca teknik bir dönüşüm değil aynı zamanda anlamlı öneri üretiminin temelini oluşturan bir yöntemdir.

3.5 Sistem Mimarisinin Genel Yapısı

Bu tez kapsamında geliştirilen öneri sistemi modern yapay zekâ ve doğal dil işleme yaklaşımlarını endüstriyel veri sınıflandırma problemlerine çözüm arayan bir mimari üzerine inşa edilmiştir. Sistem temel olarak üç ana bileşenden oluşmaktadır: veri önişleme ve embedding işlemlerini yürüten yapay zekâ modülü, çok boyutlu vektör temsillerini saklayan ve sorgulayan Elasticsearch altyapısı ve kullanıcı etkileşimini sağlayan hafif bir arayüz katmanı.

Sistemin başlangıç noktası kullanıcıdan alınan iki metinsel giriştir: ürün adı ve ürünün teknik/ticari adı. Bu iki girdi birleştirilerek OpenAI'nin sunmuş olduğu text-embedding-3-small modeli aracılığıyla yüksek boyutlu bir embedding vektöre dönüştürülür.

Üretilen embedding daha önceden resmi veri setinden (PRODTR) elde edilmiş tüm tanımların embedding'leri ile Elasticsearch üzerinde karşılaştırılır. Elasticsearch bu amaçla yaklaşık k-en yakın komşu (approximate kNN) algoritması uygular.

Elasticsearch, embedding dizinlerini dense_vector tipinde özel alanlarda saklamakta ve OpenAI vektör boyutuna (1536) uyumlu biçimde yapılandırılmıştır. Sorgulama sürecinde k = 10 ve num_candidates = 200 şeklinde ayarlanan parametrelerle, en yakın 10 eşleşme içerisinden en anlamlı sonuçlar kullanıcının karşısına çıkarılmaktadır.

Backend tarafında geleneksel bir sunucu mantığı yerine doğrudan Elastic sorgulama temelli bir yapı benimsenmiştir. Tüm uygulama Node.js üzerinde geliştirilmiş API bağlantısı veya ayrı bir servis katmanı kullanılmadan embedding işlemi doğrudan istemciden tetiklenmiş ve sonuçlar JSON biçiminde geri döndürülmüştür.

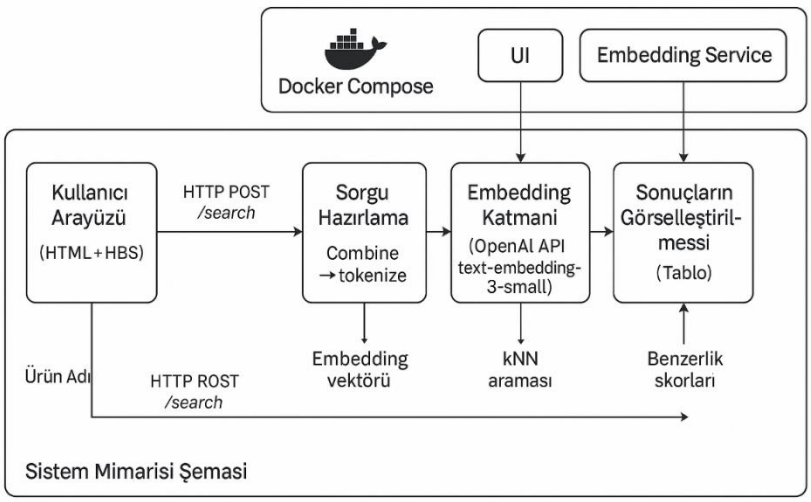
Sistem arama ve öneri sonuçlarını kullanıcıya görsel olarak sunmak amacıyla minimal bir kullanıcı arayüzü (UI) içermektedir. Bu arayüz HTML ve Handlebars.js (HBS) şablon motoru kullanılarak geliştirilmiştir. Kullanıcı ürün adı ve teknik adı için iki ayrı metin

alanını doldurarak sorgu başlatmakta sistem de ilgili benzerlik skorlarını içeren sonuçları tablo halinde listelemektedir.

Tüm bu bileşenler konteyner tabanlı bir yapı içinde çalıştırılmak üzere Docker ile paketlenmiştir. Elasticsearch servisi embedding hesaplama uygulaması ve kullanıcı arayüzü bağımsız ancak birbirine bağlı konteynerler olarak yapılandırılmıştır. Bu yapı sistemin kolayca taşınabilir ve yeniden yapılandırılabilir olmasını sağlamıştır.

Geliştirilen sistemin genel mimari yapısı aşağıdaki gibi özetlenebilir:

- Kullanıcı Arayüzü: HTML + HBS tabanlı form yapısı. Kullanıcı girişlerini alır.
- Sorgu Hazırlama: Ürün Adı ve Ürün Teknik Adı alanları birleşerek embedding oluşturulur.
- Embedding Katmanı: OpenAI API'si üzerinden text-embedding-3-small modeli ile vektör temsili elde edilir.
- Arama Katmanı: Elasticsearch üzerinde embedding dizinine yaklaşık kNN araması uygulanır.
- Sonuçların Görselleştirilmesi: En yüksek benzerlik skorlarına sahip tanımlar tablo halinde kullanıcıya sunulur.



Şekil 3.1 Sistem mimarisi

Bu mimari yapı sayesinde geliştirilen öneri sistemi hem kurumsal süreçlerde insan hatasından kaynaklı kodlama problemlerine çözüm sunmakta hem de doğal dil ile resmi sınıflamalar arasında köprü kurarak otomatikleştirilmiş bir eşleştirme süreci sağlamaktadır.

3.6 Kullanıcı Etkileşimi ve Öneri Süreci

Geliştirilen öneri sistemi kullanıcının serbest metin olarak ifade ettiği ürün bilgilerini resmi sınıflandırma sistemindeki tanımlarla otomatik ve anlamsal olarak eşleştiren bir yapay zekâ modülüyle çalışmaktadır. Bu bağlamda sistem teknik veya sektör bilgisi sınırlı kullanıcıların da doğru kodlara erişmesini kolaylaştırmak amacıyla sadeleştirilmiş bir etkileşim akışı sunar.

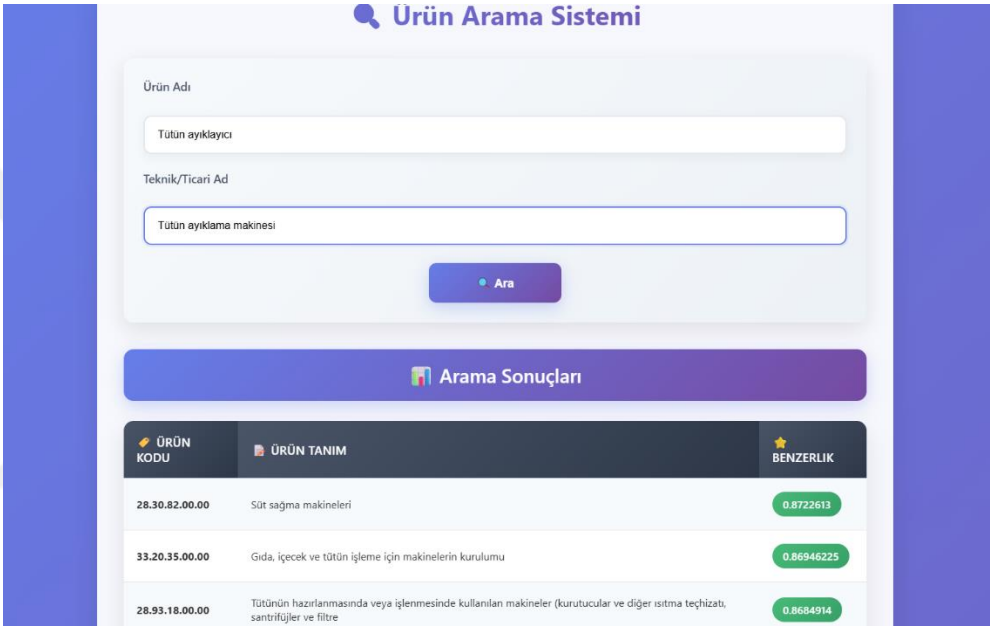
Sistemle etkileşim iki temel veri girişi ile başlar: ürün adı ve ürünün teknik ya da ticari adı. Kullanıcı bu iki alanı doğal Türkçe dilinde dilediği şekilde doldurabilir. Örneğin, "çelik boru" veya "yüksek yoğunluklu polietilen boru" gibi ifadeler doğrudan yazılabilir. Sistem bu serbest metinleri daha önceden embedding'leri alınmış resmi tanımlarla kıyaslamak için anlamsal bir vektör temsiline dönüştürür.

Ardından bu vektördaha önceden Elasticsearch dizinine alınmış resmi PRODTR tanımlarıyla karşılaştırılır. Kullanıcıya bu süreç sonunda en benzer 10 sonuç içinden en anlamlı olanlar skor sırasına göre listelenmiş şekilde sunulur. Her bir sonuçta resmi tanım kod ve eşleşme oranı (benzerlik skoru) açıkça belirtilir. Kullanıcı bu öneriler arasından seçim yaparak kendi ürününü en doğru biçimde sınıflandırabilir.

Sistem özellikle üretim tanımları gibi çok çeşitli ve karmaşık terimlerin kullanıldığı alanlarda insan hatası riskini azaltmayı ve doğru eşleştirme sürecini otomatikleştirmeyi hedeflemektedir. Girdi çeşitliliğine rağmen doğru tanımlara ulaşabilmesi sistemin anlamsal tabanlı öneri mimarisinden kaynaklanmaktadır. Örneğin, kullanıcı "tuvenan taşkömürü" yazdığında, sistem bu ifadenin "Brüt Kalori Değeri > 23,865 kj/kg olan kok

üretimine olanak sağlayan taşkömürü” tanımıyla olan yakınlığını tanıyarak ilgili resmi kodu öne çıkarmaktadır.

Sistem arayüzü basit bir HTML formu aracılığıyla bu süreci kullanıcı dostu hâle getirmektedir. Kullanıcıya yalnızca gerekli alanları doldurmak ve “Ara” butonuna basmak kalmaktadır. Sonuçlar ise tablo biçiminde açık, sade ve kıyaslanabilir bir yapıda listelenmektedir.



Şekil 3.2 Sistem arayüzü

Sonuç olarak sistemin kullanıcı etkileşim süreci arkasında yer alan embedding ve anlamsal kıyaslama mimarisine rağmen oldukça sade kullanıcı deneyimini amaçlamaktadır. Bu yapı yapay zekâ destekli öneri sistemlerinin endüstriyel sınıflandırma süreçlerinde nasıl etkili ve erişilebilir bir çözüm sunabileceğine dair başarılı bir örnek teşkil etmektedir.

4. DENEYSEL UYGULAMA VE SONUÇLAR

Embedding tabanlı öneri sisteminin performansı, PRODTR kodlarından türetilmiş ve ChatGPT tabanlı generative AI ile simüle edilmiş etiketli bir test kümesi üzerinde ölçülmüştür. Değerlendirme sürecinde, sayısal doğruluk oranlarının yanı sıra örnek senaryolar aracılığıyla bağlamsal başarı da incelenmiştir.

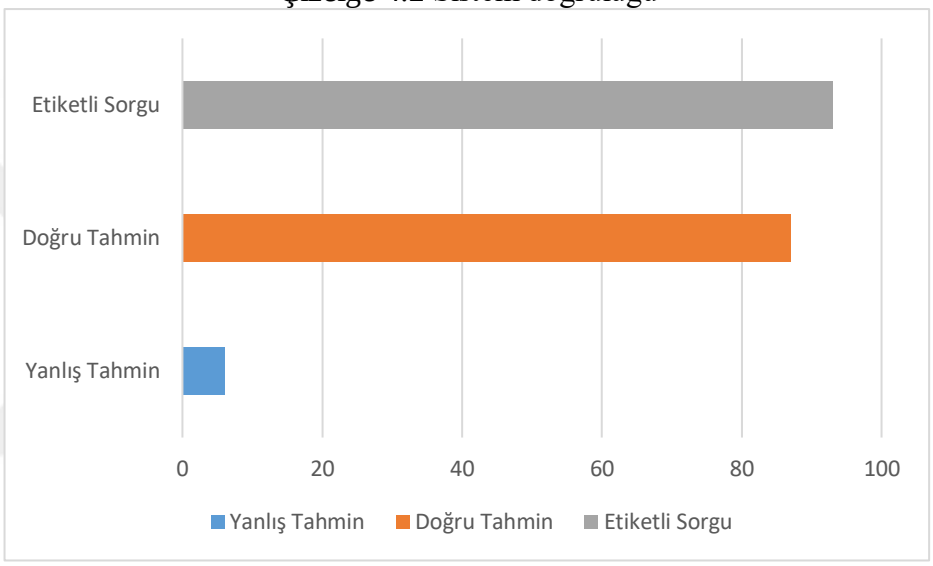
Çizelge 4.1 Üretilen veri örneği

Kod	**Ürün Tanımı**	**Ürün Adı**	**Ürün Ticari/Teknik Adı**
25.99.29.82.00	Çanlar, gonglar, vb. (elektrikli olmayan), adi metallerden	Çanlar	Çanlar
10.12.10.10.00	Tavuk etleri, taze veya soğutulmuş, bütün halde	Tavuk etleri	Yüksek Kaliteli Tavuk etleri
20.16.30.60.00	Floropolimerler	Floropolimerler	Floropolimerler
23.69.19.80.00	Çimento, beton veya suni taşlardan eşyalar, inşaat dışı kullanımlar için (vazolar, saksılar, mimari süsler veya bahçe süsleri	Çimento	Yüksek Kaliteli Çimento
21.10.20.20.00	Glutamik asit ve tuzları	Glutamik asit ve tuzları	Endüstriyel Glutamik asit ve tuzları
20.42.19.45.01	Tıraş öncesi, tıraş sırasında veya tıraştan sonra kullanılan müstahzarlar; losyonlar	Tıraş öncesi	Endüstriyel Tıraş öncesi
25.94.12.10.00	Yaylı rondelalar ve diğer sıkıştırma rondelaları, demir veya çelikten	Yaylı rondelalar ve diğer sıkıştırma rondelaları	Standart Yaylı rondelalar ve diğer sıkıştırma rondelaları
19.20.28.70.02	Fuel - oil no.6	Fuel - oil no.6	

4.1 Genel Başarı Oranı

Modelin doğruluğu sistemden dönen ilk 15 tahmin içerisinde etiketli verinin doğru kabul ettiği PRODTR kodunu içerip içermemesine göre hesaplanmıştır. Değerlendirme setinde yer alan 93 etiketli sorgu üzerinden yapılan analizde sistemin 87 sorguda doğru, 6 sorguda yanlış öneri sunduğu görülmüştür. Bu sonuçlara göre modelin genel doğruluk (accuracy) oranı %93,5 olarak hesaplanmıştır.

Çizelge 4.2 Sistem doğruluğu



4.2 Bulguların Genel Değerlendirmesi

Modelin önerilerinin doğruluk oranı benzerlik skorlarıyla birlikte analiz edildiğinde sistemin skor seviyeleriyle tahmin güvenilirliği arasında anlamlı bir ilişki kurabildiği görülmektedir. Doğru tahminlerin büyük çoğunluğu yüksek benzerlik skorlarında toplanırken yanlış tahminlerin daha çok düşük skor aralıklarında kümelendiği tespit edilmiştir.

Ayrıca sistemin bazı durumlarda düşük benzerlik skoruna rağmen doğru sınıflandırma yapabilmesi modelin yalnızca yüzeysel kelime eşleşmesine dayalı kalmadığını;

bağlamsal anlamı da başarıyla çözebildiğini göstermektedir. Bu bulgu sistemin semantik düzeyde sağlam bir kavrayış sergilediğinin göstergesidir.

Bu sonuçlar öneri sisteminin yalnızca nicel doğruluk oranlarıyla değil aynı zamanda sunduğu önerilerin güvenilirliği ve bağlamı çözümleme kabiliyetiyle de değerlendirilmesi gerektiğini ortaya koymaktadır.

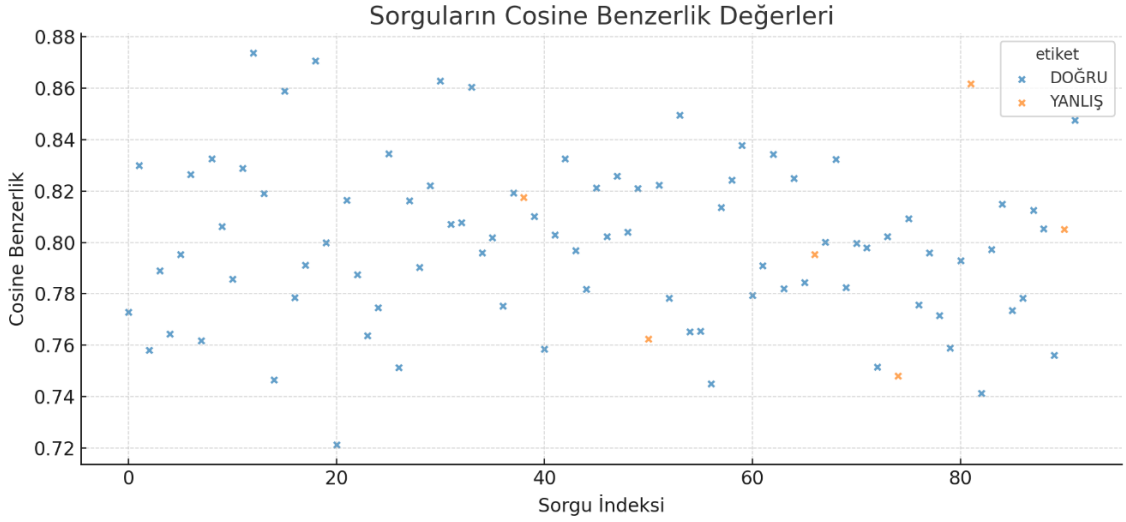
5. TARTIŞMA VE SONUÇLAR

Bu bölümde geliştirilen embedding tabanlı öneri sisteminin sunduğu performans hem nicel başarı ölçütleri hem de modelin içsel davranışlarını anlamaya dönük grafiksel analizlerle değerlendirilmiştir. %93,5 doğruluk oranı sistemin ilk önerisi baz alınarak yapılan değerlendirmede oldukça tatmin edici bir sonuç olmakla birlikte önerilerin skor bazlı güvenilirliği, bağlamsal çözümleme gücü, kullanıcıya sunduğu açıklanabilirlik seviyesi ve potansiyel uygulama alanları gibi başka boyutlar da sistem başarımının bütünsel değerlendirmesi için göz önünde bulundurulmuştur.

Geleneksel eşleştirme sistemlerinden farklı olarak bu sistem kelime seviyesinde bire bir eşleşmeler yerine kelimelerin taşıdığı bağlamı vektör uzayında modelleyerek öneride bulunmaktadır. Bu yaklaşım anlam eşleştirmesi gerektiren teknik sınıflandırma problemlerinde daha sofistike ve dayanıklı çözümler sunma potansiyeline sahiptir.

Öneri sistemlerinin değerlendirilmesinde yalnızca doğruluk oranı değil önerilere eşlik eden güvenilirlik skorlarının da karar destek bağlamında anlamlı olup olmadıkları önemlidir. Bu nedenle, sistemin cosine benzerlik skorlarının tahmin doğruluğu ile ilişkisi üç ayrı grafik ile analiz edilmiştir.

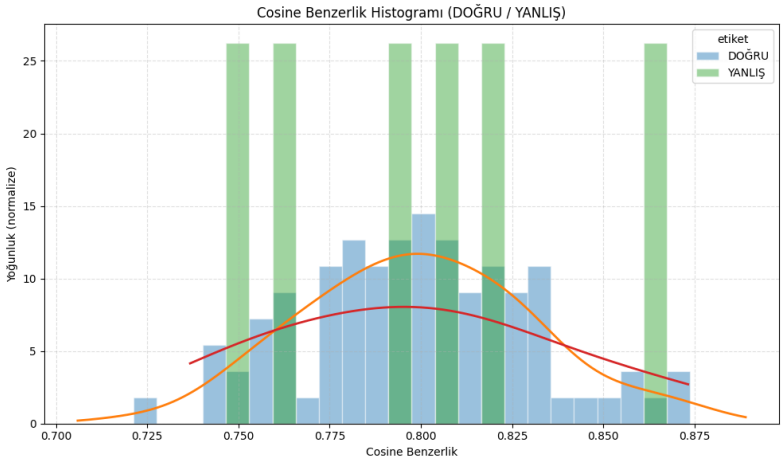
Çizelge 5.1 Kosinüs benzerlik değerlerinin sorgu bazında dağılımı



Bu grafik her bir sorgu için sistemin sunduğu ilk 15 önerinin doğru ya da yanlış olup olmadığını göstermektedir. Görselleştirmeden görüldüğü üzere:

- Doğru etiketli sonuçlar (mavi işaretler) çoğunlukla 0.78–0.86 aralığında yoğunlaşmıştır. Bu durum, sistemin büyük oranda tutarlı ve güvenilir benzerlik skorları ürettiğini göstermektedir.
- Yanlış etiketli sonuçlar (turuncu işaretler) ise daha seyrek ve düzensiz bir dağılım göstermektedir. Bu dağılım, hatalı tahminlerin belirli bir skora sıkışmadığını, farklı benzerlik seviyelerinde ortaya çıkabildiğini ortaya koymaktadır.

Çizelge 5.2 Doğru/yanlış öneriler için kosinüs benzerlik histogramı



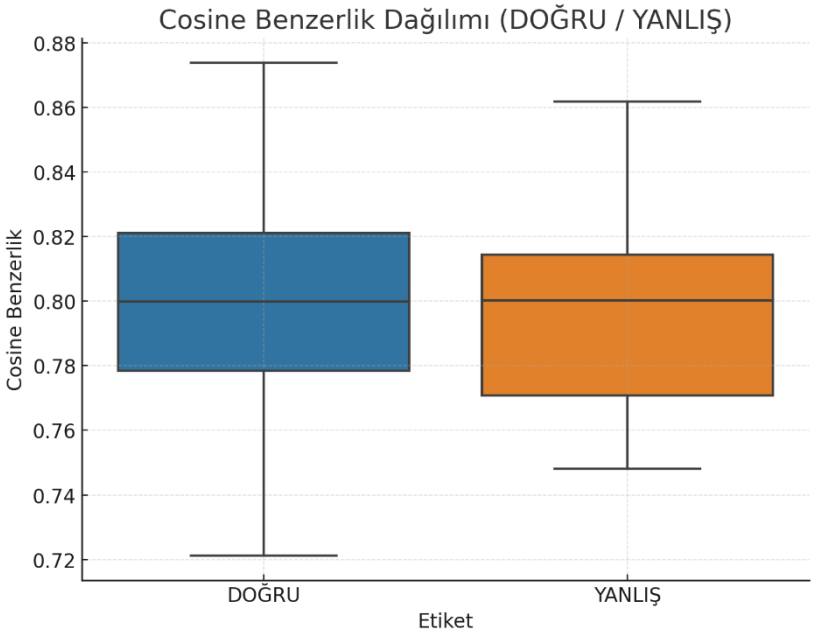
Bu histogram, sistemin ürettiği doğru (mavi) ve yanlış (yeşil) eşleşmelerin cosine benzerlik skorlarına göre nasıl dağıldığını göstermektedir.

- Doğru eşleşmeler çoğunlukla 0.78–0.83 aralığında yoğunlaşmıştır. Bu durum, sistemin yüksek benzerlik skorlarında daha güvenilir çalıştığını ortaya koymaktadır.
- Yanlış eşleşmeler ise daha çok 0.75–0.80 aralığında dağılmış olup, belirgin bir yoğunlaşmadan ziyade düzensiz bir yayılım göstermektedir. Bu da düşük skorlarda sistemin hata yapma ihtimalinin arttığını göstermektedir.

- Çizilen yoğunluk eğrileri, doğru ve yanlış örneklerin dağılım farklılıklarını görselleştirerek hangi skor aralıklarının güvenilirlik için kritik olduğunu ortaya koymaktadır.

Bu bulgu sistem için eşik değerlerin (örneğin 0.78 veya 0.80) belirlenmesinin kullanıcıya güvenilirlik bandı sunmada stratejik bir avantaj sağlayabileceğini göstermektedir. Böylece kullanıcı, yalnızca sistemin önerisini değil, aynı zamanda bu önerinin güven düzeyini de değerlendirebilir.

Çizelge 5.3 Kosinüs benzerlik dağılımı: doğru vs yanlış



Bu kutu grafiği, doğru ve yanlış önerilerin cosine benzerlik skorlarının istatistiksel dağılımlarını karşılaştırmaktadır.

- Doğru tahminlerin medyan değeri 0.80 civarında olup, alt ve üst çeyrek değerleri daha dar bir aralıkta toplanmıştır. Bu durum, doğru eşleşmelerin daha tutarlı ve öngörülebilir bir skor bandında gerçekleştiğini göstermektedir.
- Yanlış tahminlerde ise medyan değer yine 0.80 civarındadır; ancak dağılım daha geniştir ve alt sınır değerleri 0.74'lere kadar inmektedir. Bu, yanlış eşleşmelerin

belirli bir aralıkta yoğunlaşmadığını, daha düzensiz ve belirsiz bir yapı sergilediğini göstermektedir.

- Her iki dağılımın da üst sınırlarının birbirine yakın olması, yüksek skorların her zaman mutlak doğruluk garantisi vermediğini; ancak düşük skorların sistemin güvenilirliğini önemli ölçüde zayıflattığını ortaya koymaktadır.

Bu bulgu, sistemin sadece "doğru/yanlış" ayrımı üzerinden değil, aynı zamanda benzerlik skorunun istatistiksel güvenilirliği üzerinden de değerlendirilebileceğini göstermektedir.

Modelin en belirgin güçlü yönlerinden biri dilsel çeşitlilik içeren, eksik tanımlı veya teknik deyim içeren girdileri bağlamsal olarak çözümleyebilmesidir. Üretim sektörüne ait serbest metin verilerinde sıkça rastlanan çok anlamlı kelimeler, sıralama farklılıkları, sektör jargonu ve eksik bağlam gibi durumlar, klasik eşleştirme sistemlerinde ciddi başarısızlıklara neden olabilirken, bu model bu tür durumları başarılı şekilde ele alabilmiştir.

Örneğin "çelik boru imalatı" ile "boru üretimi - çelik dış kaplama" gibi anlamsal olarak yakın ama farklı formüle edilmiş girdilerde model aynı hedef sınıfı önerebilmiştir. Bu yetkinlik, modelin sadece kelime düzeyinde değil bağlam düzeyinde kavrama gerçekleştirdiğini göstermektedir. Böylece sistem, klasik kelime eşleştirmeye dayalı kurallı sistemlerin ötesine geçerek anlamsal eşleştirme gücünü kullanmaktadır.

Sistem performansı genel olarak yüksek olmakla birlikte aşağıdaki sınırlılıklar göze çarpmaktadır:

Skor Dağılımı: 0.74–0.78 aralığında doğruluk oranı %90'ın üzerinde gerçekleşmiştir. Yanlış sınıflamalar belirgin bir skor bandına yoğunlaşmak yerine dağınık biçimde ortaya çıkmıştır. Bu durum, modelin güvenilirlik eğrisinin genel olarak dengeli olduğunu göstermektedir.

Sınıf Yakınlığına Duyarlılık: Birbirine anlamsal olarak çok yakın ancak sistemde farklı kodlara karşılık gelen sınıflar (örneğin metal işleme ile çelik döküm) arasında ayrım yapmakta zorluk yaşanabilmektedir.

Buna karşılık geliştirilebilecek noktalar şunlardır:

Etkin Öğrenme: Kullanıcı geri bildirimlerini kullanan sistemler, modelin zamanla kendi doğruluğunu artırmasını sağlar.

Negatif Örnek Üretimi: Sınıflar arası ayrım gücünü artırmak için bilinçli olarak benzer görünümlü ama farklı sınıf etiketli örnekler modele eklenebilir.

Çok Dillilik: Model, aynı yapı çok dilli embedding modellerine uygulandığında farklı ülkelerdeki kullanıcılar için de kullanılabilir hâle gelir.

Açıklama Üretici Modül: Girdinin hangi ifadelere göre sınıflandığını açıklayan bir modül, önerilerin şeffaflığını artırabilir.

Sistem kamu kurumlarında sektör kodu atama, ürün belgelendirme ya da KOBİ danışmanlıklarında kullanılabileceği gibi, özel sektörde tedarikçi analizi, e-fatura sınıflandırması ve otomatik raporlama süreçlerine de kolayca entegre edilebilir.

Bu çalışmada, serbest metinli kullanıcı girdilerinin resmi ürün sınıflarına doğru şekilde eşlenebilmesi amacıyla geliştirilen embedding temelli öneri sistemi, yüksek doğruluk oranı (%93,5) ile anlam odaklı eşleştirme gücünü ortaya koymuştur.

Yapılan grafiksel analizler bu skorların istatistiksel olarak tahmin kalitesiyle uyumlu olduğunu göstermiş sistemin kullanıcıya sadece öneri değil aynı zamanda karar destek sinyali sunduğu anlaşılmıştır.

Sonuç olarak geliştirilen sistem, sektörel sınıflandırma görevlerinde yalnızca teknik bir yapay zekâ bileşeni değil karar destek, iş zekâsı ve dijital dönüşüm araçlarının bir parçası olmaya adaydır.

KAYNAKLAR

Bello, A. 2024. Embedding-based product classification in manufacturing industries. Expert Systems with Applications, 232, 120948.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. 2020. Language models are few-shot learners. Advances in Neural Information Processing Systems, 33, 1877–1901.

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. Proceedings of NAACL-HLT, 4171–4186.

Figueiredo, M. 2025. Generative AI with SAP and Amazon Bedrock: Utilizing GenAI with SAP and AWS Business Use Cases. Apress/Springer.

Firth, J. R. 1957. Papers in Linguistics 1934–1951. Oxford University Press.

Gonzalez-Gomez, L. J., Hernandez-Munoz, S. M., Borja, A., Arana-Salas, F. A., Azofeifa, J. D., Noguez, J., & Caratozzolo, P. 2025. Dynamic taxonomy generation for future skills identification using a named entity recognition and relation extraction pipeline. Frontiers in Artificial Intelligence, 8, Article 1579998.

Gurevych, I. 2019. Neural learning for natural language processing. Technical Report, UKP Lab.

IBM Research. 2020. Advancing natural language processing. IBM Research Blog.

Jalammar, J. 2018. The Illustrated Transformer. Retrieved from https://jalammar.github.io/illustrated-transformer/

Manivelan, M. 2024. Text vectorization models: Performance analysis in domain-specific categorization. Journal of Artificial Intelligence Research, 81(1), 45–63.

md.teyit.org. 2020. Yıllık Sanayi Ürün İstatistikleri Kod Listesi (PRODTR 2017–2018). Retrieved from https://md.teyit.org/file/yillik-sanayi-urun-istatistikleri-kod-listesi.pdf

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. 2013. Distributed representations of words and phrases and their compositionality. Advances in Neural Information Processing Systems, 26.

Pennington, J., Socher, R., & Manning, C. D. 2014. GloVe: Global vectors for word representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1532–1543.

Peyton, K., Unnikrishnan, S., & Mulligan, B. 2025. A review of university chatbots for student support: FAQs and beyond. Discov Educ, 4, 21.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of Machine Learning Research.

Raza, M., Jahangir, Z., Riaz, M. B., et al. 2025. Industrial applications of large language models. Scientific Reports, 15, 13755.

Reimers, N., & Gurevych, I. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, 3982–3992. Association for Computational Linguistics.

Roussinov, D. 2023. Fine-tuning language models to recognize anlamsal relations. International Journal of Lexical Computing.

SAP News. 2024. Generative AI hub in SAP AI Core integrates with foundation models in Amazon Bedrock. SAP News.

Sutor, P., Aloimonos, Y., Fermüller, C., & Summers Stay, D. 2019. Metaconcepts: Isolating context in word embeddings.

Te'eni, D., Yahav, I., Zagalsky, A., Schwartz, D. G., & Silverman, G. 2023. Reciprocal human-machine learning: A theory and an instantiation for message classification. Management Science.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. 2017. Attention is all you need. Advances in Neural Information Processing Systems, 30.

Vidyaratne, L., Nallaperuma, D., & Samarakoon, M. 2025. Evaluation of anlamsal search techniques in low-resource taxonomies. Information Processing & Management, 62(1), 102854.

Wevers, M., & Koolen, M. 2020. Digital begriffsgeschichte: Tracing anlamsal change using word embeddings. Historical Methods: A Journal of Quantitative and Interdisciplinary History, 5.