**Proje Adı:** Türkçe Metinlerde Duygu Analizi ve Konu Çıkarımı

**Amaç:** Türkçe metinleri işleyerek duygu analizi yapmak ve metinlerdeki ana konuları çıkarmak

**Örnek Veri seti:** <https://huggingface.co/datasets/winvoker/turkish-sentiment-analysis-dataset>

**Örnek veri setinin tanımı ve durumu:**

Türkçe metinlerin sınıflandırılması için oluşturulmuş bir veri setidir. Bu veri seti genellikle doğal dil işleme (NLP) ve duygu analizi projelerinde kullanılır. Veri seti, içerdiği metinleri sınıflandırmak amacıyla olumlu, olumsuz, nötr etiketlerle belirtir. Duygu analizi modellerinin eğitiminde ve değerlendirilmesinde kullanılabilir. Örneğin, duygu analizi modellerini eğitmek veya doğrulamak için bu veri setini kullanabilirsiniz. Bu veri seti ham metin verilerinden oluşturulmuştur. Üzerinde herhangi bir işlem uygulanmamıştır. Veriler kullanılmadan önce aşağıda 2. adımda belirtilen ön işleme adımlarından geçirilmesi gerekmektedir. Veri setinin içeriğine ait örnek aşağıda verilmiştir.

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Adımlar:**

1. **Veri Toplama:** Türkçe içerikli metin verilerini toplayın. Bu veriler, sosyal medya paylaşımları, müşteri yorumları veya haber metinleri olabilir.
   1. **Veri toplama araçları:** Web Scraping Araçları (BeautifulSoup, Scrapy), Sosyal Medya API'leri (Twitter, Facebook), Anketler ve Formlar
   2. **Veri birleştirme yöntemleri:** Birden fazla kaynaktan gelen metin verilerini birleştirmek için çeşitli teknikler kullanılabilir. Örneğin, metinlerin benzerliklerini belirleyerek veya metinler arasındaki ilişkileri analiz ederek verileri birleştirebilirsiniz.
2. **Veri Ön İşleme**
   1. **Veri ön işleme yöntemleri:**
      1. **Metin Normalizasyonu:** Metin verilerini normalleştirerek, büyük-küçük harf dönüşümü gibi işlemleri yaparak veriyi daha tutarlı hale getirebilirsiniz.
      2. **Gereksiz Karakterlerin Kaldırılması:** Metin verilerinden gereksiz karakterlerin (örneğin, noktalama işaretleri, özel karakterler) kaldırılması, veriyi daha temiz hale getirmektedir.
      3. **Tokenizasyon:** Metin verilerini kelimelere veya cümlelere ayırarak tokenlere dönüştürme işlemidir. Bu işlem, metin verilerini daha küçük parçalara böler ve analiz için daha uygun hale getirir.
      4. **Stop Words (Durma Kelimeleri) Kaldırma:** Dilde sıkça kullanılan ancak analiz için anlamsız olan kelimeleri (durma kelimeleri) kaldırarak veriyi temizler.
      5. **Stemming ve Lemmatization:** Stemming, kelimelerin köklerine indirgeme işlemidir. Lemmatization ise kelimeleri sözlükteki kök hallerine dönüştürme işlemidir. Bu yöntemler, kelime çeşitliliğini azaltarak veriyi daha anlaşılır hale getirir.
      6. **N-gram Oluşturma:** Metin verilerindeki kelime gruplarını oluşturarak, kelime dizilerinin yanı sıra kelime sıralarını da göz önünde bulunduran analizler yapmayı sağlar.
   2. **Veri ön işleme araçları:**

**2.2.1. NLTK (Natural Language Toolkit):** Python'da doğal dil işleme için kullanılan popüler bir kütüphanedir. NLTK, metin verilerini tokenleştirmek, durma kelimelerini kaldırmak, stemming ve lemmatization gibi ön işleme işlemlerini yapmak için kullanılabilir.

**2.2.3. TextBlob:** Basit ve kullanımı kolay bir Python kütüphanesidir. Metin verilerini temizlemek, cümleleri ayrıştırmak, durma kelimelerini kaldırmak gibi temel ön işleme işlemlerini yapmak için kullanılabilir.

1. **Özellik Çıkarma:**
   1. **Özellik çıkarma yöntemleri:**
      1. **TF-IDF (Term Frequency-Inverse Document Frequency):** Bir belgedeki terimin ne kadar önemli olduğunu belirlemek için kullanılan istatistiksel bir ölçüdür. Terimin bir belgedeki sıklığı, o terimin **o belgedeki önemini** belirlerken, terimin bütün belgelerdeki sıklığı o terimin **genel önemini** belirler.
      2. **Word Embeddings (Kelime Gömme):** **Kelimeleri** sayısal vektörlerle temsil eden tekniklerdir. Örneğin, Word2Vec, GloVe gibi algoritmalarla kelime vektörleri oluşturulabilir. Bu vektörler, kelimenin anlamını yakalamak için kullanılabilir.
      3. **Bag of Words (BoW):** Belirli bir metinde geçen **kelimelerin sayısını** vektör olarak temsil eden basit bir yöntemdir. Kelimelerin sırası veya ilişkisi göz ardı edilir, sadece kelimenin varlığı önemlidir.
      4. **Word Frequencies (Kelime Sıklıkları):** Bir belgedeki her kelimenin sıklığını içeren bir vektör, belgenin içeriği hakkında genel bilgi sağlayabilir.
   2. **Özellik çıkarma araçları:** Metin verilerini sayısal vektörlerle temsil etme işlemidir. Bu işlem, metin verilerini makine öğrenimi algoritmalarıyla kullanabilmek için gereklidir.
      1. **Gensim:** Metin verilerini vektörlere dönüştürmek ve temsil etmek için kullanılan bir Python kütüphanesidir. Özellikle kelime vektörleri oluşturma ve kelime gömme modelleri üzerinde çalışmak için kullanışlıdır.
      2. **Scikit-learn:** Genel amaçlı bir makine öğrenimi kütüphanesidir, ancak metin verilerini ön işleme için bazı araçlar sağlar. Özellikle metin verilerini vektörlere dönüştürmek için kullanılabilir.
2. **Metin Sınıflandırma Algoritması Tasarımı:**
   1. **Metin sınıflandırma için yaygın kullanılan algortimalar (teknikler, algoritmalar):**

**4.1.1 SVM (Support Vector Machine):** SVM, metin sınıflandırmasında da sıkça kullanılan bir sınıflandırma algoritmasıdır. Özellikle çoklu sınıflandırma problemlerinde etkilidir.

**4.1.2 KNN (k-Nearest Neighbors):** KNN, basit ve veriye dayalı bir sınıflandırma algoritmasıdır. Metin verileri üzerinde de kullanılabilir, ancak büyük veri kümeleri için maliyetli olabilir.

**4.1.4 Evrişimli Sinir Ağları (Convolutional Neural Networks):** CNN'ler, özellikle metin sınıflandırmasında kelime gömme (word embedding) teknikleriyle birleştirilerek kullanıldığında etkilidir. Özellikle metin verilerindeki lokal özellikleri (kelime gruplamaları) tanımak için kullanılmaktadır.

**4.1.5 Transformer Models:** Transformer modelleri, özellikle BERT (Bidirectional Encoder Representations from Transformers) ve GPT (Generative Pre-trained Transformer) gibi modeller, metin sınıflandırmasında büyük başarı elde etmiştir. Bu modeller, büyük metin veri kütleleri üzerinde önceden eğitilir ve daha sonra ince ayar yaparak belirli metin sınıflandırma görevlerinde kullanılabilir.

* 1. **Metin sınıflandırma için kullanılan araçlar (teknikler, algoritmalar):**
     1. **TensorFlow ve Keras**: TensorFlow ve Keras, derin öğrenme modelleri oluşturmak ve eğitmek için kullanılan araçlardır. Özellikle TensorFlow, metin sınıflandırması için kullanılan çeşitli derin öğrenme modelleriyle birlikte gelir.
     2. **PyTorch:** PyTorch, derin öğrenme modelleri oluşturmak ve eğitmek için kullanılan bir araçtır. Metin sınıflandırması için çeşitli önceden eğitilmiş modeller ve örnekler bulunmaktadır.
     3. **scikit-learn:** scikit-learn, makine öğrenimi için genel amaçlı bir Python kütüphanesidir. Metin sınıflandırması için kullanılabilecek çeşitli sınıflandırıcılar ve öznitelik çıkarma araçları içerir.

1. **Duygu Analizi:** Türkçe metinlerdeki duyguları (olumlu, olumsuz, nötr) belirlemek için duygu analizi yapılabilir. Bu adımda, kelime bazlı veya cümle bazlı duygu analizi yöntemleri kullanılabilir.
2. **Görselleştirme:**
   1. **Görselleştirme yöntemleri:**
      1. **Confusion Matrix (Karmaşıklık Matrisi):** Karmaşıklık matrisi, sınıflandırma sonuçlarını görselleştirmek için kullanılan temel bir araçtır. Gerçek ve tahmin edilen sınıfların karşılaştırılmasını sağlar.
      2. **Precision-Recall Curve (Hassasiyet-Duyarlılık Eğrisi):** Hassasiyet ve duyarlılık metriklerini görselleştirmek için kullanılır. Farklı sınıflandırma eşikleri altında hassasiyet ve duyarlılık değerlerini gösterir.
      3. **ROC Curve (Receiver Operating Characteristic Eğrisi):** ROC eğrisi, duyarlılık ve özgüllük arasındaki ilişkiyi görselleştirir. Sınıflandırma modelinin performansını değerlendirmek için kullanılır.
      4. **Class Distribution Visualization (Sınıf Dağılımı Görselleştirme):** Sınıf dağılımını göstermek için çubuk grafikleri veya pasta grafikleri gibi basit grafikler kullanılabilir. Bu, sınıflar arasındaki dengesizliği görselleştirmek için faydalı olabilir.
      5. **Word Clouds (Kelime Bulutları):** Metin verilerinde en sık geçen kelimeleri görselleştirmek için kelime bulutları kullanılabilir. Farklı sınıflara ait metinlerdeki belirgin kelimeleri göstermek için faydalı olabilir.
   2. **Görselleştirme araçları:**
      1. **Matplotlib:** Matplotlib, Python dilinde güçlü bir görselleştirme kütüphanesidir. Confusion matrix, precision-recall curve, ROC curve gibi grafikleri oluşturmak için sıklıkla kullanılır.
      2. **Seaborn**: Seaborn, Matplotlib'e dayalı daha yüksek seviyede bir görselleştirme kütüphanesidir. Matplotlib'in daha karmaşık grafikler oluşturmak için bazı zorluklarını giderir.
3. **Sonuçlar ve Öneriler:** Analiz sonuçlarından çıkarılan bilgileri değerlendirin ve işletmeler veya araştırmacılar için öneriler sunun.
4. **Proje Çıktısı:** Bu proje, Türkçe metinleri analiz ederek duygu ve konu çıkarımları yaparak, Türkçe içerikli verilerden anlamlı bilgiler elde etmelerine yardımcı olabilir.

**Örnek referans makale:**

1. A Comprehensive Analysis of Static Word Embeddings forTurkish (Sarıtaş ve diğ. 2023)