**Yapay Sinir Ağları ile Türkçede Konuşma Etiketleme (Pos Tagging) Yapabilmek için Yöntem Geliştirme Örnek Tasarımı**

**Hazırlayan**

Fırat Kaan Bitmez

**Öğrenci Numarası**

23281855

**Dersin Hocası**

Asst.Prof.Dr. İsmail İşeri

**Giriş**

Bu Raporda/Makalede Türkçe metinler üzerinde Part-of-Speech (POS) tagging yapmak için yapay sinir ağı kullanımını ele almaktadır. POS tagging her kelimenin dil bilgisi açısından rolünü belirlemeyi amaçlayan bir işlemdir. Bu çalışmada, Türkçe metinlerden oluşan bir veri kümesi kullanılarak bir yapay sinir ağı örnek model tasarımı ve aşamaları oluşturulmuştur.

**Amacımız**

Bu çalışmanın amacı, Türkçe metinlerde **Yapay sinir ağı** kullanılarak Türkçe Pos Tagging yapabilmek için veri kümesinin nasıl hazırlanmasından başlayarak, özellik vektörünün yapısının nasıl olabileceği konusunda bir örnek tasarım hazırlamak ve tasarım örneğini hazırlarken daha önce yapılan çalışmalardan olan **DEVELOPING METHODS FOR PART OF SPEECH TAGGING IN TURKISH** isimli makaleyi inceleyeceğiz nasıl adımlar izlediklerini çıkarım yapacağız daha sonra ise biz Yapay Sinir Ağı kullanarak bu adımları nasıl yaparız sorusuna cevaplar arayacağız. İnceleyeceğimiz makalenin linkini en altta **kaynaklar** bölümünde bulabilirsiniz.

**DEVELOPING METHODS FOR PART OF SPEECH TAGGING IN TURKISH Makalesinden Kısaca Bahsetmemiz Gerekirse:**

Bu makalede Türkçe konuşma Etiketlemede **SVM**(Destek Vektör Makineleri) Yaklaşımı kullanılarak örnek bir yöntem geliştirilmiş. Tabi daha öncesinde Part-of-Speech, Pos Tagging gibi. kavramlara geniş bir yer verilerek bu kavramlar açıklanmış. Kısaca bahsetmem gerekirse:

Part-of-Speech Tagging =Konuşma sırasında Etiketleme

**Part-of-Speech Tagging**: Bir metindeki sözcüklerin belirli bir şeye karşılık gelecek şekilde etiketlenmesi POS etiketleme morfolojik analizin basitleştirilmiş bir biçimi olarak kabul edilebilir.

**Neden part-of-speech Tagging’e ihtiyacımız var?** : Konuşmanın bir ksımını etiketleme ile adımlardan biridir. Doğal Dil İşlemede önceki aşamalar bir sonraki aşamayı besler bu yüzden ihtiyacımız var.

**Part-of-Speech Tagging Yaklaşımları**

1. Kural Tabanlı Yaklaşım
2. TBL (Dönüşüm Temelli Öğrenme)
3. Markov Modeli Yaklaşımları
4. Maximum Entropi Yaklaşımları
5. Destek Vektör Makineleri (SVM)

**DEVELOPING METHODS FOR PART OF SPEECH TAGGING IN TURKISH** Makalesinde yapılan çalışmada kullanılan **SVM** yaklaşımıdır.

**SVM**, diğer yöntemlere göre iki temel avantaja sahiptir. Yüksek boyutlu uzaylarda işlem yapabilme yeteneğine sahiptir ve bu, genellikle birçok özelliğin kullanılması durumunda yararlıdır. Ayrıca, aşırı uymaya karşı dayanıklıdır, yani genelleme yeteneği daha iyidir. Ancak, diğer yöntemlerin (kural tabanlı, TBL, Markov modeli, maksimum entropi) avantajları ve dezavantajları da incelenebilir. Örneğin, kural tabanlı yaklaşımın dilbilgisi kurallarına dayalı olması, belirli metin türlerinde daha iyi performans gösterebilirken, SVM'in daha genel bir yaklaşım sunması ve veriye dayalı öğrenme yapabilmesi tercih edilebilir.

**Örnek Model Tasarım Aşamaları**

**Veri Kümesi Hazırlığı:**

1. **Veri Toplama**: İlk adım veri toplama, POS etiketli Türkçe cümleler içeren bir veri kümesi oluşturmaktır. Bu veri kümesi, farklı türlerde metinlerden (makaleler, köşe yazıları, kitaplar vb.) toplanabilir.
2. **Veri Temizleme ve Ön İşleme**: Toplanan veri kümesi, gereksiz karakterlerin ve boşlukların temizlenmesi için işlenmelidir. Burada aslında temizle işlemine gürültü, kirlilik gibi farklı isimlerde verilebiliyor. Ardından, cümleler ayrıştırılmalı ve kelimelerine ayrılmalıdır.
3. **POS Etiketleme**: Her kelimeye uygun bir POS etiketi atanmalıdır. Bu etiketler, veri kümesinde bulunan POS etiketleriyle uyumlu olmalıdır.

**Özellik Vektörünün Yapısı:**

1. **Kelime Özellikleri**: Her kelimenin kendisine özgü özellikleri vardır.

Örneğin:

* Kelimenin uzunluğu
* Baş harfinin büyük/küçük olması
* Noktalama işareti içermesi
* Rakam içermesi

1. **Önceki ve Sonraki Kelimelerin Özellikleri**: Bir kelimenin POS etiketi, önceki ve sonraki kelimelerin POS etiketlerine bağlı olabilir.

Örneğin:

* Önceki kelimenin POS etiketi
* Sonraki kelimenin POS etiketi
* Önceki kelimenin son üç karakteri
* Sonraki kelimenin ilk iki karakteri

1. **Sözlük Bazlı Özellikler**: Türkçe dil bilgisi kurallarını temsil eden bir sözlük oluşturulabilir. Bu sözlük, belirli kelimelerin belirli POS etiketleriyle ilişkilendirilmesini içerir.

Örneğin:

* Kelimenin Türkçe sözlükte bulunup bulunmaması
* Kelimenin sözlükteki anlam sayısı
* Kelimenin yaygınlığı

**Yapay Sinir Ağı Tasarımı:**

1. **Giriş Katmanı**: Her kelimenin özelliklerini içeren bir giriş vektörü oluşturulur. Bu vektör, kelimenin özelliklerinin birleştirilmesiyle oluşturulur.
2. **Gizli Katman(lar)**: Yapay sinir ağında bir veya daha fazla gizli katman bulunabilir. Her gizli katmanda, giriş vektöründeki özelliklerin bir ağırlık matrisiyle çarpılması ve bir aktivasyon fonksiyonu (ReLU, sigmoid vb.) ile işlenmesi gerçekleştirilir.
3. **Çıkış Katmanı**: Çıkış katmanı, POS etiketlerinin olasılıklarını içerir. Birden fazla POS etiketi varsa, çoklu sınıflandırma için softmax aktivasyon fonksiyonu kullanılabilir.
4. **Eğitim**: Yapay sinir ağı, eğitim veri kümesi üzerinde eğitilir. Gerçek çıktılarla (POS etiketleri) tahmin edilen çıktılar arasındaki hatayı minimize etmek için bir kayıp fonksiyonu kullanılır (örneğin, çapraz-entropi kaybı).
5. **Değerlendirme**: Eğitim sona erdikten sonra, yapay sinir ağı test veri kümesi üzerinde değerlendirilir. Doğruluk ve diğer metrikler kullanılarak performansı değerlendirilir.

Bu tasarım, Türkçe POS etiketlemesi için bir yapay sinir ağı modeli oluşturmak için temel bir çerçeve sunar. Veri hazırlığı ve özellik vektörü tasarımı, modelin başarı oranını etkileyen önemli adımlardır. Yapay sinir ağının mimarisi ve eğitim süreci, modelin karmaşıklığına ve doğruluğuna bağlı olarak ayarlanabilir.

**Kütüphaneler ve Araçlar**

**Python** programlama dili kullanılarak **yapay sinir ağı örnek tasarım modeli** geliştirilmiştir.

Yapay sinir ağı modeli oluşturmak için **TensorFlow** veya **PyTorch** gibi **derin öğrenme kütüphaneleri** kullanılabilir.

**Veri işleme** ve özellik mühendisliği için **pandas**, **NumPy** ve **scikit-learn** gibi **kütüphaneler** kullanılabilir.

**Metin verilerini işlemek** için **NLTK** (Natural Language Toolkit) gibi doğal dil işleme kütüphaneleri kullanılabilir.

**Örnek bir Python kod parçası yazarsak şu şekilde olabilir:**

import numpy as np

import pandas as pd

import tensorflow as tf

from tensorflow.keras import layers, models

from sklearn.model\_selection import train\_test\_split

from nltk.tokenize import word\_tokenize

from nltk import pos\_tag

# Veri kümesini yükleme ve ön işleme

# Veri kümesinin yüklenmesi, temizlenmesi ve POS etiketlerinin alınması işlemleri burada gerçekleştirilir.

# Özellik vektörlerinin oluşturulması

# Her kelime için özellik vektörleri oluşturulur. Bu özellik vektörleri, kelimenin uzunluğu, büyük/küçük harf olması gibi özellikleri içerir.

# Yapay sinir ağı modelinin tanımlanması

model = models.Sequential([

    layers.Dense(128, activation='relu', input\_shape=(feature\_vector\_size,)),

    layers.Dense(64, activation='relu'),

    layers.Dense(num\_classes, activation='softmax')

])

# Modelin derlenmesi

model.compile(optimizer='adam',

              loss='sparse\_categorical\_crossentropy',

              metrics=['accuracy'])

# Modelin eğitilmesi

history = model.fit(X\_train, y\_train, epochs=10, batch\_size=32, validation\_data=(X\_val, y\_val))

# Modelin değerlendirilmesi

test\_loss, test\_accuracy = model.evaluate(X\_test, y\_test)

**Sonuç**

Bu çalışmada, Türkçe metinler üzerinde doğru POS etiketlerini tahmin etmek için bir yapay sinir ağı model örnek tasarımı geliştirilmiştir. Veri hazırlığı, özellik vektörünün tasarımı ve modelin eğitim süreci gibi aşamalar ele alınmıştır. Geliştirilen modelin, test veri kümesi üzerindeki performansı test adımları yapılmamıştır sadece örnek bir tasarımda neler yapılmalı bunlar ele alınmıştır. Yapay sinir ağı modeli, Türkçe metinlerde POS etiketleme görevini etkili bir şekilde gerçekleştirebilecek güçlü bir araç olarak öne çıkmaktadır.

Bu çalışma, Türkçe doğal dil işleme alanında önemli bir adım olup, gelecekteki çalışmalar için bir temel oluşturmuştur. Modelin daha da iyileştirilmesi ve genişletilmesi, Türkçe metinler üzerindeki doğal dil işleme uygulamalarının kalitesini artıracaktır.

**Öneriler**

1. Modelin Performansını İyileştirmek İçin Daha Fazla Özellik Ekleyin: Özellik vektörü tasarımınızı genişletebilirsiniz. Örneğin, kelimenin kökünü (lemma) ekleyerek veya kelimenin morfolojik özelliklerini içeren özellikler ekleyerek modelin performansını artırabilirsiniz.
2. Daha Fazla Veri Kullanın: Veri setinizi genişletmek, modelin genelleme yeteneğini artırabilir. Farklı kaynaklardan daha fazla POS etiketli Türkçe metinleri toplayabilir ve veri kümenizi çeşitlendirebilirsiniz.
3. Modelin Mimarisini Geliştirin: Modelinizin mimarisini ayarlayarak veya daha karmaşık bir yapıya geçerek performansı artırabilirsiniz. Örneğin, daha fazla gizli katman ekleyerek veya mevcut katmanların genişletilmesiyle modeli daha karmaşık hale getirebilirsiniz.
4. Hiperparametre Ayarlaması Yapın: Eğitim sürecinde kullanılan hiperparametreleri (örneğin, öğrenme oranı, toplu iş boyutu, epoch sayısı) ayarlayarak modelin performansını iyileştirebilirsiniz.
5. Transfer Learning Kullanın: Eğer elinizde yeterince büyük bir veri kümesi yoksa, transfer learning yöntemlerini kullanarak önceden eğitilmiş bir modelden başlayabilirsiniz. Örneğin, Türkçe metinler üzerinde genel doğal dil işleme görevlerinde önceden eğitilmiş bir dil modeli üzerine özelleştirme yapabilirsiniz.
6. Regularizasyon Uygulayın: Aşırı öğrenmeyi önlemek için regularizasyon tekniklerini (örneğin, dropout) kullanabilirsiniz.
7. Farklı Optimizasyon Algoritmalarını Deneyin: Modelinizi eğitmek için farklı optimizasyon algoritmalarını deneyerek daha iyi sonuçlar elde edebilirsiniz.
8. Farklı Aktivasyon Fonksiyonlarını Deneyin: Modelinizde farklı aktivasyon fonksiyonlarını deneyerek performansı artırabilirsiniz.

**Kaynaklar**

DEVELOPING METHODS FOR PART OF SPEECH TAGGING IN TURKISH

by Berna Arslan & Özlem Patan

<https://www.cmpe.boun.edu.tr/~gungort/undergraduateprojects/Developing%20Methods%20for%20Part%20of%20Speech%20Tagging%20in%20Turkish.pdf>