maltepe university
istanbul
www.maltepe.edu.tr

Faculty of Engineering and Natural Sciences

Turkish Joke Generator

Prepared by

Furkan Aksoy ~ 210706029

Emre Sarı ~ 220706304

Mehmet Güzel ~ 210706030

Tamay Yazgan ~ 210706022

Ömer Faruk Özer ~ 210706028

Yaren Yıldız ~ 200706040

Table Of Contents

Turkish Language Models for Joke Generation and Platform Analysis

1. Introduction

In recent years, AI-powered text generation has made significant progress, particularly in domains that require cultural and linguistic creativity. This report examines how Turkish language models—primarily available on Hugging Face and through OpenAI's offerings—can be leveraged for generating humorous content, specifically Turkish jokes (fıkra). The analysis includes an evaluation of notable models, their use cases, and a comparison of platforms. This document is adapted for our Turkish Joke Generator project, which integrates a fine-tuned GPT-2 model to create culturally relevant humorous content.

2. Hugging Face Platform and Turkish Language Models

2.1 Overview

Hugging Face is a leading platform for open-source Natural Language Processing (NLP) models. It hosts a variety of models that are specifically trained for the Turkish language. Many of these models are compatible with the 'transformers' library in Python, making them easily integrable into various applications, including joke generation.

2.2 Notable Models

- **dbmdz/bert-base-turkish-cased:**
  A BERT model trained specifically for Turkish, useful for language understanding tasks.
- **savasy/bert-base-turkish-sentiment-cased:**
  Designed for sentiment analysis in Turkish; while not directly a

generative model, its underlying architecture can inform humor and tone.

- **xlm-roberta-base:**
  A multilingual model that includes Turkish; often used for tasks requiring a broader language understanding.
- **cagri/zemberek-turkish-morphology:**
  This model focuses on morphological analysis in Turkish, essential for processing the linguistic nuances found in humor and wordplay.

2.3 Use Cases for Turkish Joke Generation

The models hosted on Hugging Face can be applied to various NLP tasks, such as:

- **Text Classification:** Differentiating joke types or categorizing humor based on sentiment.
- **Named Entity Recognition (NER):** Identifying cultural references like "Nasreddin Hoca" or "Temel."
- **Morphological Analysis:** Essential for understanding the structure of Turkish jokes.
- **Text Generation:** Particularly with GPT-2-based models, which are well-suited for creative content generation like jokes.

2.4 Example: Turkish Sentiment Analysis

Below is an example of using a Hugging Face model for sentiment analysis in Turkish, which can be adapted for pre-processing humor-related content:

```
from transformers import pipeline

classifier = pipeline("sentiment-analysis", model="savasy/bert-base-turkish-sentiment-cased")
```

```
result = classifier("Bu oyun harika olmuş!")
print(result)
```

3. OpenAI Platform and Turkish Language Support

3.1 General Information

OpenAI's GPT-4 and GPT-3.5 models are trained on extensive multilingual datasets, which include Turkish. Although these models are not exclusively trained for Turkish, they demonstrate strong performance in creative text generation tasks, including humor. Their capability to maintain coherence in long-form content makes them a strong candidate for generating culturally nuanced jokes.

3.2 Advantages

- **Conversational Fluency:** These models understand and generate conversational Turkish well.
- **Humor Generation:** They are effective at producing humorous content with unexpected twists.
- **Coherent Long-form Content:** They maintain context and coherence over extended outputs, important for multi-sentence joke narratives.

3.3 Example: Turkish Humor Text Generation

Here is an example using OpenAI's API to generate a Turkish anecdote in the style of Nasreddin Hoca:

```
import openai

openai.api_key = "API_KEY"
response = openai.ChatCompletion.create(
```

```
    model="gpt-4",
    messages=[
        {"role": "user", "content": "Write me a short, funny Turkish anecdote in the style of Nasreddin Hoca."}
    ]
)
print(response['choices'][0]['message']['content'])
```

4. Key Considerations in Turkish Joke Generation

When generating Turkish jokes, several linguistic and cultural aspects need to be considered:

- **Linguistic Structure:** Turkish humor often relies on wordplay, idioms, and puns. The model must capture these nuances to produce authentic jokes.
- **Cultural Context:** Jokes frequently reference folklore and cultural figures such as Nasreddin Hoca and Temel. Incorporating these references is essential for maintaining cultural relevance.
- **Tone and Style:** Jokes should be concise, fluent, and contain a twist or punchline that surprises the audience.
- **Dataset Quality:** To develop a robust model, a curated dataset of Turkish jokes is crucial. This dataset should be diverse enough to cover various styles and topics in Turkish humor.

5. Hugging Face vs. OpenAI: Platform Comparison

The following table compares the two platforms based on several criteria relevant to our project:

Criteria	Hugging Face	OpenAI
Open-source	Yes	No

Criteria	Hugging Face	OpenAI
Turkish-specific models	Yes	Limited (multilingual models include Turkish)
Humor Generation	Available (mainly through GPT-2 based models)	Moderate to Strong (with GPT-4/GPT-3.5)
Ease of Use	Moderate (requires integration via transformers)	Easy (API-based, plug-and-play)
Offline Usage	Yes	No

6. Conclusion & Sample Joke

AI-powered Turkish joke generation holds significant promise for both educational and entertainment industries. By leveraging the strengths of Hugging Face's open-source models and OpenAI's advanced text generation capabilities, projects like the Turkish Joke Generator can offer innovative solutions for applications in gaming, mobile apps, virtual assistants, and more.

Example Joke (Generated with GPT-4):

Temel wakes up early, looks in the mirror, and says: "Temel, this village can't handle so much handsomeness. Better be late to the coffee house!"

7. References

- **Hugging Face Documentation:**
  Hugging Face Transformers
  Provides guidelines and best practices for using open-source NLP models.
- **OpenAI Documentation:**
  OpenAI API Documentation
  Essential for understanding how to implement GPT-3.5 and GPT-4 for text generation tasks.

- **Schwaber, K., & Sutherland, J. (2020).** *The Scrum Guide.*
  Offers insights into agile development methodologies which inform our project management.
- **Schwalbe, K. (2015).** *Information Technology Project Management.*
  Cengage Learning.
  Provides foundational concepts in project planning, risk management, and resource allocation.
- **Flask Official Documentation:**
  Flask
  Reference for developing robust web applications using Flask.
- **Google Colab Documentation:**
  Google Colab
  Supports model training and collaborative coding practices.