Prakash Saput is a celebrated Nepali singer and social influencer with over **2.51 million YouTube subscribers**. His work blends music with activism, tackling important social issues in Nepal. Recently, he faced online controversy after expressing support for PM KP Oli, prompting polarized public responses.

This project aims to **analyze audience sentiment** on YouTube comments from his videos using **Natural Language Processing (NLP)** techniques.

Here is what he says about himself:

“*I'm Prakash Saput, and my life is a story of dreams, music, and a deep commitment to social change. Growing up in Nepal, I found solace in music, beginning as a radio jockey and later releasing songs that addressed critical issues. I've transformed from a village dreamer into a singer, composer, model, actor, and advocate for change. Nepal is my muse, and I'm dedicated to using my voice and art for its positive transformation.”*

Dataset

* Total comments: 244722
* Columns: video\_id, author, comments
* Language Mix:
  + Nepali (Devnagari script)
  + English
  + Romanized Nepali (Nepali written in latin script)
  + Unknown/Symbolic comments

Models and Methods

1. **TextBlob**

* Pros: Fast, intitutive for English sentiment analysis
* Features: Sentiment analysis, tokenization, POS tagging, spelling correction
* Limitations: Fails on Nepali and Romanized Nepali
  + Returns 0 polarity/subjectivity for positive comments like
    - Dherai ramro song chha ❤❤
    - Music compose मा Jhuma limbu jiu koi name...

from textblob import TextBlob

text = TextBlob("Songs are really nice heart touching plz we need other songs as well plzzz as soon as possible 🙏🙏")

print(text.sentiment)

Sentiment(polarity=0.24375000000000002, subjectivity=0.71875)

1. Multilingual BERT via. Hugging Face Transformers

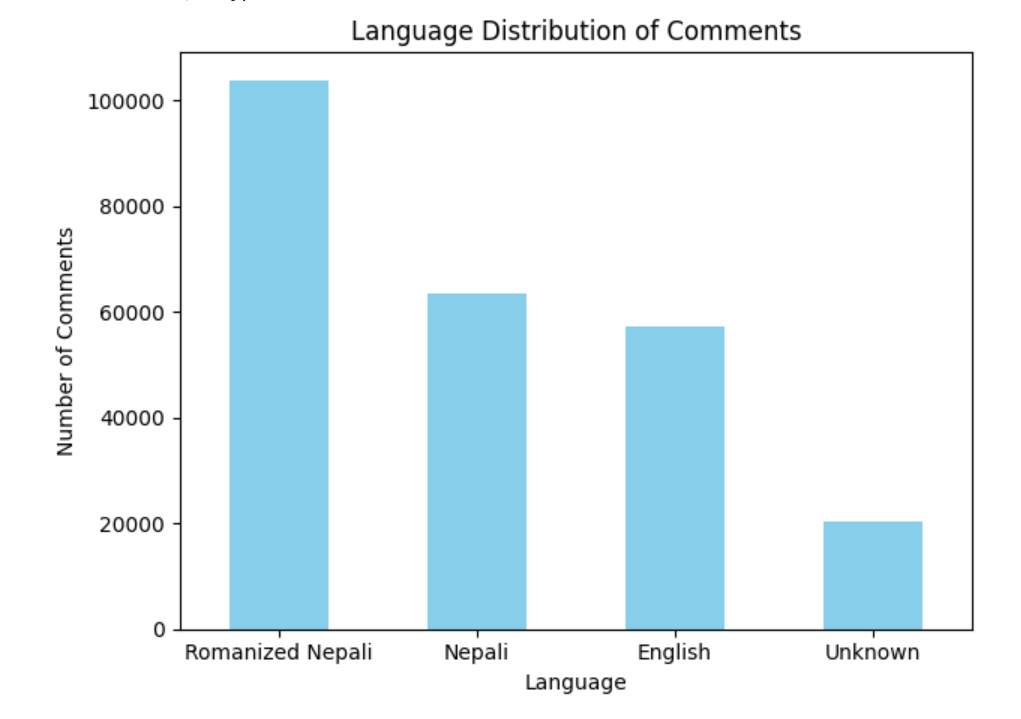
* Model: nlptown/bert-base-multilingual-uncased-sentiment
* Supports: Nepali (Devanagari), English, many European and South Asian languages
* Output: Star rating (1 to 5), with confidence score
* from transformers import pipeline
* sentiment\_pipeline = pipeline("sentiment-analysis", model="nlptown/bert-base-multilingual-uncased-sentiment")
* results = sentiment\_pipeline(["मलाई यो गीत धेरै राम्रो लाग्यो।", "I hated this movie."])
* print(results)
* [{'label': '5 stars', 'score': 0.5550222396850586}, {'label': '1 star', 'score': 0.7954590916633606}]
  + Input: "Dherai ramro song chha ❤❤" → Output: {'label': '1 star', 'score': 0.28}
  + This is a known **failure case**, as the model is **not trained on Romanized Nepali**

1. Next Step

* Dropped romanized and unknown comments and retained only English + Nepali comments for further sentiment analysis
* Final retained comments: 120, 539
* BERT accepts maximum 512 tokens, some comments exceeds this so truncated long comments to the first 512 characters before passing to the model.

1. Key takeways

* TextBlob works well for **English**, but **not usable for Nepali** or Romanized Nepali.
* Multilingual BERT supports **Nepali**, but fails on **Romanized Nepali**.
* Filtering language + truncating text is essential for clean results.
* Over 100K valid comments were processed for sentiment with confidence.



A graph with blue bars

AI-generated content may be incorrect.

Figure: Sentiment distribution on filtered comments, after removing romanized comments and made longer comments shorter that are of same length as for BERT input 512