Persian Sentiment Analysis

Amirmahdi Aramideh

November 2024

Abstract

This project introduces a custom Transformer-based architecture designed for sentiment analysis on Persian text reviews from the SnappFood dataset. By leveraging a pretrained BERT tokenizer for text encoding and fine-tuning a lightweight Transformer encoder, the model effectively classifies reviews as HAPPY or SAD.. This approach addresses challenges posed by Persian morphology and script, demonstrating the effectiveness of custom architectures for low-resource NLP tasks.

1. Introduction

Sentiment analysis is essential for understanding user opinions in platforms like e-commerce and food delivery. While English dominates NLP research, languages like Persian remain underrepresented due to limited datasets and linguistic complexities such as rich morphology and script variations.

This study focuses on sentiment analysis of Persian text reviews from the SnappFood dataset. We designed a custom Transformer model optimized for sentiment classification while leveraging the pretrained BERT tokenizer for robust tokenization. The approach achieves competitive results with reduced computational overhead compared to full-scale pretrained models.

2. Related Work

Traditional sentiment analysis approaches for Persian relied on classical methods like Naive Bayes and SVMs, enhanced by manually crafted features. The emergence of Transformer-based architectures, such as BERT, ParsBERT, and mBERT, has revolutionized Persian NLP by enabling transfer learning from multilingual corpora.

Our work diverges by creating a custom Transformer model that balances computational efficiency with strong performance, using the pretrained BERT tokenizer to encode text sequences.

3. Methodology

3.1 Dataset

3.1.1 Overview

The **SnappFood Persian Sentiment Analysis Dataset** is a curated collection of customer reviews in Persian, designed to support sentiment analysis tasks. This dataset contains reviews left by customers about their experiences with food delivery services, focusing on sentiments expressed toward food quality, delivery service, and overall satisfaction. The dataset includes three key components:

- **Comment**: The textual content of the review written in Persian, ranging from short phrases to multi-sentence feedback.
- **Label**: A categorical representation of sentiment, expressed as `happy` or `sad`.
- **Label ID**: A numerical representation of the sentiment, where `0` corresponds to `happy` and `1` corresponds to `sad`.

This clear structure enables the dataset to serve as a foundational resource for developing and evaluating sentiment classification models for Persian text.

3.1.2 Characteristics

The dataset presents unique features and challenges:

Languages and Script
The reviews are written in Persian, a morphologically rich language that uses the Arabic script. Persian includes additional characters not present in Arabic and is known for its informal writing styles, especially in user-generated content.

Sentiment Classes
The dataset is divided into two sentiment classes.
Happy reviews typically contain positive language reflecting satisfaction, such as “عالی

بود", while sad reviews often highlight dissatisfaction, such as "غذا سرد بود"

Review Length
The length of the reviews varies significantly. Short reviews often express clear sentiment, while longer reviews may present mixed or nuanced opinions, making them more challenging to classify accurately.

Class Distribution
The balance of the two sentiment classes was assessed during preprocessing. An imbalanced distribution can impact model performance by biasing predictions toward the majority class.

Domain-Specific Vocabulary
The dataset contains many terms related to food delivery services, such as "سفارش" (order) or "پیک" (delivery driver). These terms, combined with informal and colloquial Persian, add complexity to the classification task.

3.1.3 Preprocessing Steps

To prepare the dataset for sentiment analysis, several preprocessing steps were applied to the reviews.

Tokenization and Padding
The comments were tokenized using a pretrained Persian tokenizer capable of handling Persian script and morphology. Each tokenized sequence was truncated or padded to a fixed length, ensuring consistency in input size for the model.

Label Mapping
The `Label` column was mapped to its corresponding numerical `Label ID`, converting the categorical sentiment classes (`happy` and `sad`) into numerical labels (`0` and `1`).

Attention Masking
Attention masks were created to differentiate between real tokens and padding tokens. These masks helped the model focus on meaningful parts of the review during training and ignore padding.

Class Balancing
To address potential class imbalance, techniques such as weighted loss functions or oversampling of the minority class were considered to ensure the model treated both sentiment classes fairly.

3.2 Model

3.2.1 Model Overview

The sentiment analysis model developed for this study is a custom Transformer-based architecture designed to handle the linguistic complexities of Persian text and the nuances of sentiment expressed in customer reviews. Leveraging the foundational components of the Transformer encoder, the model is capable of processing tokenized input sequences, capturing contextual relationships between tokens, and effectively classifying sentiments as either positive or negative. The architecture incorporates domain-specific adaptations, ensuring robust performance in analyzing the SnappFood dataset.

3.2.2 Model Architecture

The model begins by embedding tokenized input sequences into dense vector representations using a trainable embedding layer. This layer is complemented by a positional embedding that helps the model understand the sequential nature of the text. The core of the model is the Transformer encoder, a stack of multiple layers equipped with multi-head self-attention and feedforward networks. These mechanisms enable the model to identify long-range dependencies and intricate contextual relationships between words, which are crucial for understanding sentiment in lengthy and complex reviews.

After processing the input through the Transformer encoder, the model applies mean pooling to aggregate token-level features into a single vector representing the overall review. This pooled representation is then passed through a series of fully connected layers,

where it is refined and projected onto a lower-dimensional space. The final layer outputs probabilities corresponding to the two sentiment classes, `happy` and `sad`. To ensure regularization and improve generalization, the architecture incorporates dropout layers and applies a GELU activation function for non-linearity.

3.2.3 Training Configuration

The model is optimized using the Adam optimizer, which is well-suited for deep learning tasks due to its adaptive learning rate and efficient handling of sparse gradients. Cross-entropy loss is used as the loss function, given its effectiveness in classification tasks involving multiple classes. To stabilize training, gradient clipping is employed, which prevents the gradients from growing excessively large during backpropagation. Regularization techniques, such as dropout, are integrated throughout the model to mitigate overfitting. Additionally, the architecture is implemented to run on GPUs using PyTorch, allowing for faster computation and scalability.

3.2.4 Training and Evaluation

The training process involves feeding tokenized input sequences and their corresponding attention masks into the model, comparing predictions against true sentiment labels, and optimizing the model's weights using backpropagation. Metrics such as accuracy ,precision , recall and F1-score are tracked during training to monitor progress and ensure the model converges effectively. In the evaluation phase, the model processes test data without updating its weights, providing a clear assessment of its performance on unseen data. Metrics like validation loss and accuracy are used to evaluate the model's generalization capability.

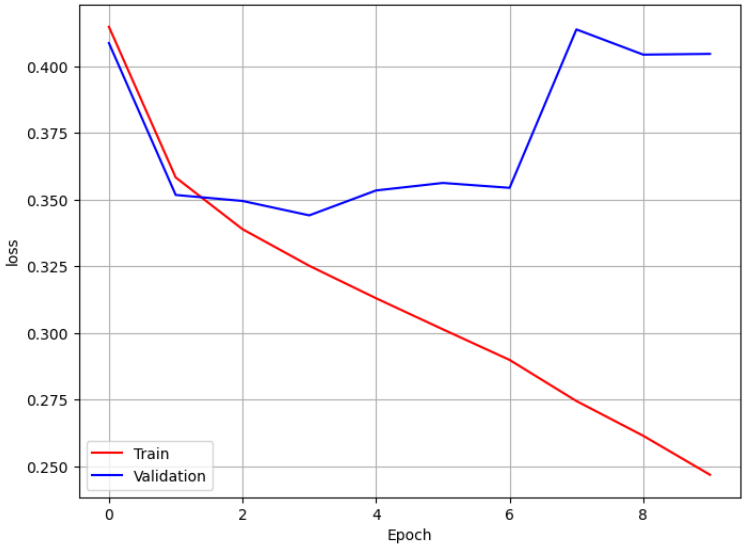
3.3 Training Process

The input text was tokenized using the BERT tokenizer, which efficiently handles Persian-specific linguistic characteristics. Tokenized sequences and attention masks were used as inputs to the model. The model was trained with a categorical cross-entropy loss function and optimized using AdamW. Hyperparameters, such as learning rate, batch size, and dropout rates, were fine-tuned to balance training speed and accuracy.

embedding_dim = 256
hidden_dim = 512
output_dim = 2

4. Results and Analysis

The custom Transformer model achieved the following performance on the SnappFood dataset:



Model	Proposed model	ParsBert
Accuracy	85.30%	-
Precision	85.59%	-
recall	86.75%	-
F1-score	85.27%	88.12%

5. Conclusion

This project proposed a custom Transformer model for sentiment analysis on Persian text reviews. By combining the efficiency of lightweight Transformers with the robustness of pretrained BERT tokenization, the model effectively addressed challenges associated with Persian sentiment analysis. Future work will focus on expanding the dataset, incorporating domain-specific pretraining, and exploring advanced regularization techniques.