**Character Trait Model Evaluation**

This project implements a character trait recommendation system that identifies similar characters traits across different movies based on their dialogue patterns. In this evaluation, we compared pairwise similarities between character embeddings and assessed how well the model groups similar characters together. The system uses natural language processing techniques to analyze character dialogue and generate personalized recommendations.

**Core Model Components**

* **Zero-Shot Classification Framework**
  + Model Name: RoBERTa-Large-MNLI Model
  + RoBERTa model fine-tuned on the Multi-Genre Natural Language Inference (MNLI) dataset.
  + The model is implemented using the HuggingFace zero-shot-classification pipeline, which streamlines the process of applying the model to new text without requiring task-specific training data.
* **Text Preprocessing Pipeline**
  + Lowercase Conversion: Normalizes all text to lowercase.
  + Whitespace Normalization: Replaces multiple spaces with single spaces.
  + Text Cleaning: Removes extraneous characters and standardizes format.
  + Truncation: The model's maximum token limit is 512 tokens.
* **Trait Hypothesis Formation**
  + Candidate Labels: Defined set of personality traits that represent the possible characteristics we want to detect in each character.
  + Hypothesis Generation: The pipeline automatically constructs a natural language inference hypothesis. The model then evaluates whether the dialogue entails, contradicts, or is neutral toward this hypothesis.
  + Multi-Label Classification: Since characters can exhibit multiple traits simultaneously, we treat this as a multi-label classification problem. The model assigns a probability score to each trait independently.
* **Similarity Computation**
  + Trait Vector Representation: Each character is represented as a vector of trait probabilities or binary trait indicators.
  + Method: Character similarity is computed using cosine similarity between trait vectors.
  + Threshold Optimization: Evaluate model performance across different threshold values (0.3 to 0.7) to find the optimal balance between precision and recall.
  + Cross-Movie Filter: Excludes recommendations from the same movie.

**Performance Metrics**

For the best performing model (base accuracy 0.9), here are the per-trait Precision, Recall, and F1 scores.

|  |  |  |  |
| --- | --- | --- | --- |
| **Trait** | **Precision** | **Recall** | **F1** |
| friendly | 1 | 0.9333 | 0.9657 |
| cowardly | 0.9167 | 1 | 0.9568 |
| compassionate | 1 | 0.9 | 0.9477 |
| hostile | 1 | 0.875 | 0.9336 |
| courageous | 0.9091 | 0.9091 | 0.9094 |
| naive | 0.8462 | 0.9167 | 0.8803 |
| funny | 0.9 | 0.8182 | 0.8574 |
| ruthless | 0.8 | 0.8889 | 0.8424 |
| extroverted | 0.75 | 0.9231 | 0.8279 |
| introverted | 0.75 | 0.9 | 0.8185 |
| intelligent | 0.8125 | 0.8125 | 0.8128 |
| serious | 0.5714 | 0.8889 | 0.696 |

Visualization of F1 Scores over various Accuracy levels.

A group of graphs with different colored lines

AI-generated content may be incorrect.

**Analysis and Insights**

The evaluation indicates that the character trait similarity model is effective at grouping similar characters based on dialogue embeddings. The trade-offs between precision and recall can be managed by adjusting the similarity threshold, with a threshold near 0.55 providing a good balance. Additionally, visualization confirms that the embeddings preserve semantic relationships between characters.

The analysis shows:

* Higher base accuracy leads to better F1 scores across all thresholds
* The optimal threshold varies depending on whether we use strict or lenient evaluation
* Some traits like "friendly" and "cowardly" are easier to detect than others
* Including secondary traits (lenient evaluation) generally improves overall performance

**Error Analysis**

Based on the testing results, here are some key points from the error analysis:

* Threshold Sensitivity: The evaluation indicates that performance can vary significantly with the chosen confidence threshold. For example, at lower thresholds, while recall tends to be higher (capturing more traits), precision suffers; whereas higher thresholds yield better precision but may miss some valid trait signals. This suggests that the zero-shot predictions are somewhat sensitive to thresholding, and finding the optimal value is crucial.
* Strict vs. Lenient Evaluation Differences: The strict evaluation (considering only primary traits) generally yields lower recall and F1 scores compared to the lenient evaluation (which includes secondary traits). This discrepancy shows that the classifier sometimes under-predicts traits that are moderately present. It indicates that the confidence scores might not be sharply discriminative, especially for traits with ambiguous presence in the dialogue.
* Trait-Specific Variability: Some traits, like "friendly" or "hostile," have consistently higher accuracy and F1 scores, whereas others such as "naive" or "serious" show more variability, suggesting that the model may be more adept at detecting clear expressions of certain personality styles than subtler traits. This variability implies that certain trait hypotheses might need refinement.

**Recommendations for Implementation**

* Optimize Thresholding
  + The model's performance is sensitive to prediction thresholds
  + Consider adapting the thresholds based on real-world feedback or treatment groups to better balance precision and recall
* Leverage Detailed Trait Profiles
  + Use the output trait vectors as a nuanced user profile
  + Capture both primary and secondary trait signals
* Combine Multiple Similarity Cues
  + Consider integrating additional features such as movie genre, context of dialogue, or even temporal dynamics where possible