Team34-->Varchar(4)

Dialogue Act Classification

using DistilBERT

built from scratch

SE22UARI165; SE22UARI014;
SE22UARI142; SE22UARI141

Abstract Introduction

This project is training a DistilBERT model for sequence classification; for a dialogue-related task.

This project focuses on Dialogue Act Classification using DistilBERT, a transformer-based model, trained on the DailyDialog dataset. The code incorporates preprocessing, tokenization, and a robust model training pipeline using Hugging Face's Trainer API. By leveraging PyTorch for custom datasets and integrating early stopping, the model achieves efficient and accurate predictions of dialogue acts. The journey also included exploring Hierarchical Multi-Task Learning (H-MTL), emotion prediction, and extensive hyperparameter tuning.

Prior related work:

- Tried implementing Hierarchical Multi-Task Learning (H-MTL):
  - Explored integrating dialogue act and emotion classification in a hierarchical structure.
  - Introduced coarse-grained (acts) and fine-grained (emotions) classification challenges.
- Emotion Prediction:
  - Used the DailyDialog dataset with emotion labels.
  - Experimented with joint prediction of acts and emotions.
- Hyperparameter Tuning:
  - Focused on optimizing batch size, learning rates, and training epochs.
- Failures & Iterations:
  - Initial experiments with H-MTL suffered from imbalanced task contributions.
  - Pivoted towards a simpler yet effective single-task model for dialogue act classification.

DATASET

The DailyDialog dataset serves as the foundation, with train(70%), validation(15%), and test(15%) splits handled independently to prevent data leakage.

To summarise; the dataset has 3 columns: dialogue(string), act(categorical), emotion(categorical)

Categorical Representation:

- **dialog**: a list of string features.

- **act:** a list of classification labels, with possible values including __dummy__ (0), inform (1), question (2), directive (3) and commissive (4).

- **emotion:** a list of classification labels, with possible values including no emotion (0), anger (1), disgust (2), fear (3), happiness (4), sadness (5) and surprise (6).

- ***Dataset Preprocessing:***

1. Text Cleaning: Lowercased and stripped unnecessary spaces.
2. Fixing Malformed Labels:
    - Addressed issues with the act column containing malformed lists.
    - Used Python's ast.literal_eval to parse string representations of lists.
    - Resolved ambiguous classifications by selecting the most frequent label.

Tokenization

- DistilBERT Tokenizer:
    - Tokenizes dialogue text with max_length=128 and ensures padding and truncation.
    - Produces input_ids and attention_mask tensors for model input.
- This step standardizes input format for both training and inference.

Methodology/ Model

Base Model: DistilBERT.

- Modified classification head for 5 dialogue act labels.

Data Pipeline:

- **Preprocessing:** Cleaned and tokenized the text using DistilBERT Tokenizer.
- Custom PyTorch Dataset for input tensors (input_ids, attention_mask) and target labels.

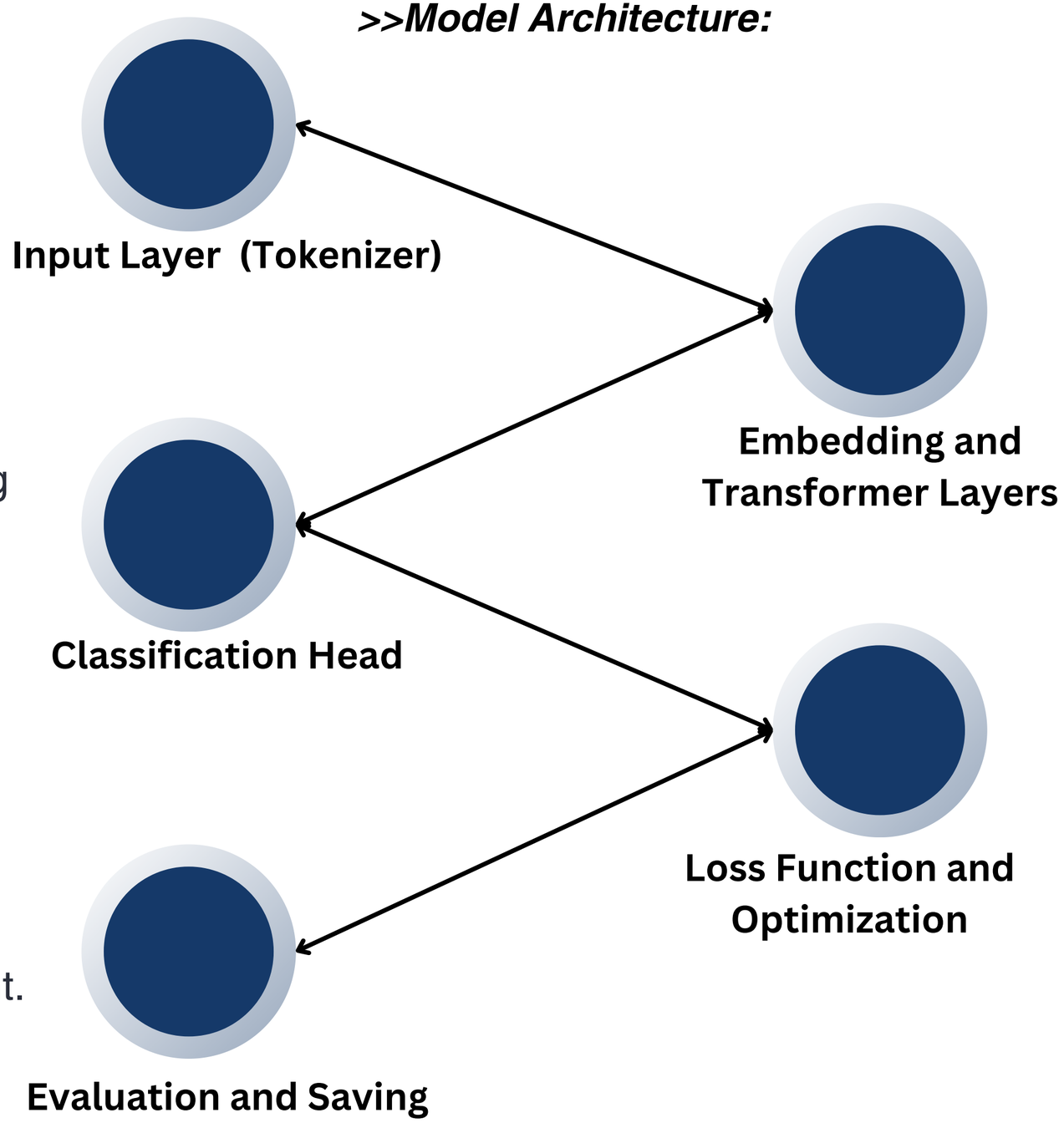
Training Pipeline:

- Utilized Hugging Face's Trainer API.
- Configured TrainingArguments:
  - Epochs: 10
  - Batch size: 16
  - Early stopping after 5 epochs without improvement.

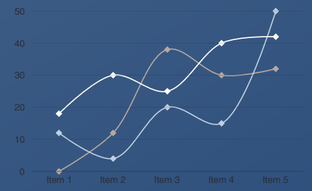
Evaluation:

- Monitored validation loss and accuracy metrics.

>>Model Architecture:



Experiments



Baseline Model : Dialogue act classification using DistilBERT.

Hyperparameter Tuning

- Experiments:
  - Learning Rates: 2e-5, 5e-5 (optimal: 2e-5).
  - Batch Sizes: 16, 32 (optimal: 16).
  - Epochs: 5, 10, 15 (early stopping at ~10).

Multi-Task Learning

- Tasks:
  - Added emotion prediction alongside dialogue acts.
  - Explored coarse-to-fine hierarchical classification.
- Challenges:
  - Emotion prediction diluted the main task.
  - Label restructuring for coarse-to-fine classification added complexity.
- Decision: Returned to single-task focus for better performance.

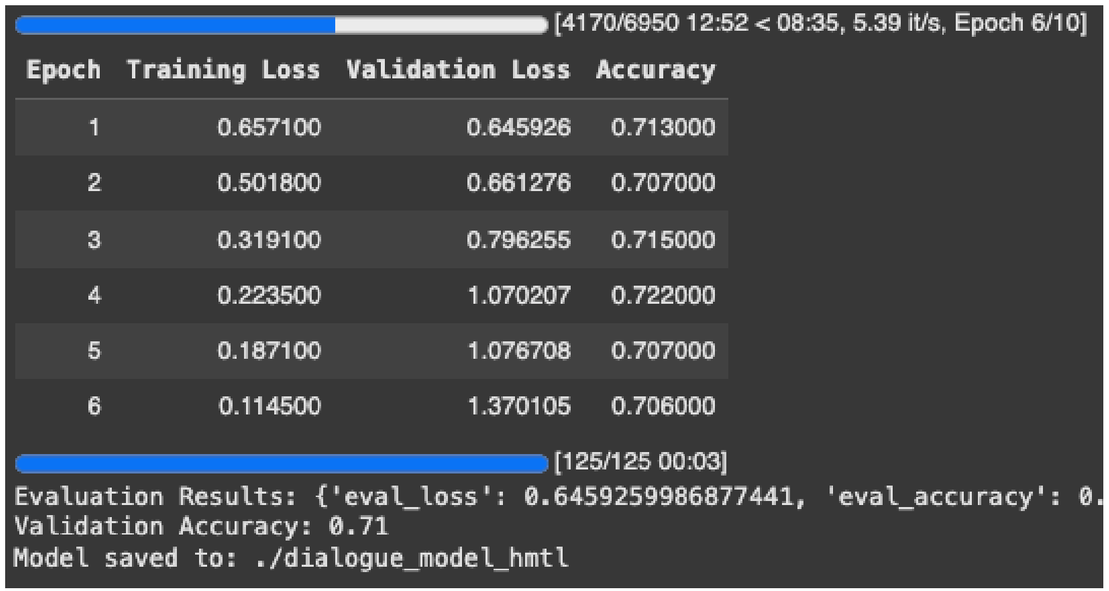
Evaluation Metrics

- Tracked:
  - Validation Accuracy: Key for model performance insights.
  - Validation Loss: Guided early stopping and model selection.

Tokenization Experiment

- Tried: AutoTokenizer for flexibility and compatibility. But, DistilBertTokenizer showed slight performance improvements due to task-specific optimizations.

RESULTS



Performance Metrics

- Training and Validation Accuracy:
  - Epoch 1: Training Loss: 0.6571 | Validation Loss: 0.6459 | Accuracy: 71.3%
  - Epoch 4: Training Loss: 0.2235 | Validation Loss: 1.0702 | Accuracy: 72.2%
  - Epoch 6: Training Loss: 0.1145 | Validation Loss: 1.3701 | Accuracy: 70.6%
- Best Validation Accuracy: 72.2% at Epoch 4.

Observations

- Training loss decreased steadily, showcasing effective learning.
- Validation loss increased after Epoch 4, indicating potential overfitting.
- Best results achieved early in training; early stopping proved effective.

Evaluation Results

- Final Validation Accuracy: 71.3%

Output

- Model and tokenizer saved for deployment at: ./dialogue_model_hmtl.

Analysis & Conclusions

Analysis

- Training Trends: Training loss consistently decreased across epochs, reflecting successful learning.
- Validation Trends: Validation loss started increasing after Epoch 4, signaling overfitting despite continued training.
- Best Results: Achieved 72.2% validation accuracy at Epoch 4, highlighting optimal early stopping point.
- Component Initialization: Some classifier weights were randomly initialized, which may have affected convergence.

Conclusions

1. Model Performance:
    - Achieved a reasonable 71.3% final validation accuracy.
    - Shows potential but could benefit from further tuning or architecture enhancements.
2. Challenges Identified:
    - Overfitting in later epochs, suggesting a need for better regularization or adaptive stopping strategies.
    - Training on a larger dataset or applying data augmentation could improve generalization.
3. Future Directions:
    - Experiment with advanced models like BERT or RoBERTa for better semantic representation.
    - Explore additional techniques, such as multi-task learning or hierarchical classification refinements, to enhance performance.

Output Of the CLI;

```
Interactive Dialogue Act Classifier
Type 'exit' to quit.
Enter a dialogue: i am going to hyd today
Predicted Dialogue Act: inform
Enter a dialogue: can you do this by tmro evening or night
Predicted Dialogue Act: directive
Enter a dialogue: are you awake?
Predicted Dialogue Act: question
Enter a dialogue: i am eating lunch
Predicted Dialogue Act: inform
Enter a dialogue: how are you?
Predicted Dialogue Act: question
Enter a dialogue: can you paint it red
Predicted Dialogue Act: directive
Enter a dialogue: exit
Exiting. Goodbye!
```

Thank You!

Team-34

VARCHAR(4)