CSE556: Natural Language Processing
Assignment-3

Himanshu Raj (2022216) | Ishita (2022224) | Ritika Thakur (2022408)

April 5, 2025

1 Task 1

1.1 Preprocessing Steps

Preprocessing is done by the tokenizer when it tokenizes the input for the model. We are removing the speaker names present in the dataset, for example,"First Citizen :" or"All :" as it doesn't contribute to the meaning of a sentence, and we don't want to generate this in our intended task. We are converting all the text to lowercase and removing extra whitespace. We are also removing punctuations - [: , . ? ! ;] - as they don't contribute to the meaning of the sentence except for apostrophes, for example, "We know't , we know't .", here 't means it.

1.2 Model Architecture and Hyperparameters

The model is a causal (autoregressive) Transformer-based Language Model and follows the decoder-only paradigm, where causal masking is used to prevent attention to future tokens. It learns to predict the next token in a sequence based on previous ones.

The **MultiHeadAttention** block has linear projection layers for query, key and value. It splits Q, K and V across multiple heads, performs self-attention across heads, merges them back and passes through a final projection layer.
The **TransformerBlock** has a multi-head self-attention block and a feed-forward neural network of two linear layers with GELU activation over the first layer. It has two Add & Norm layers for each multi-head attention block and feed-forward network. It implements skip connections as well.
The **TransformerLM** uses an embedding layer to generate token embeddings of dimension 'd_model'. It adds positional encoding to each embedding. It has a stack of 6 transformer blocks and then a final linear layer to project hidden states to vocabulary logits.

The model parameters are MAX_LEN = 256, d_model = 512, num_heads = 8, dropout_probability = 0.1, d_query = d_key = d_value = 64 (d_model/num_heads).
The hyperparameters (chosen empirically after trying various configurations) are learning_rate = 1e-4 and num_epochs = 20.

1.3 Training and Validation Loss

Figure 7 shows the plot for training and validation loss over the epochs.

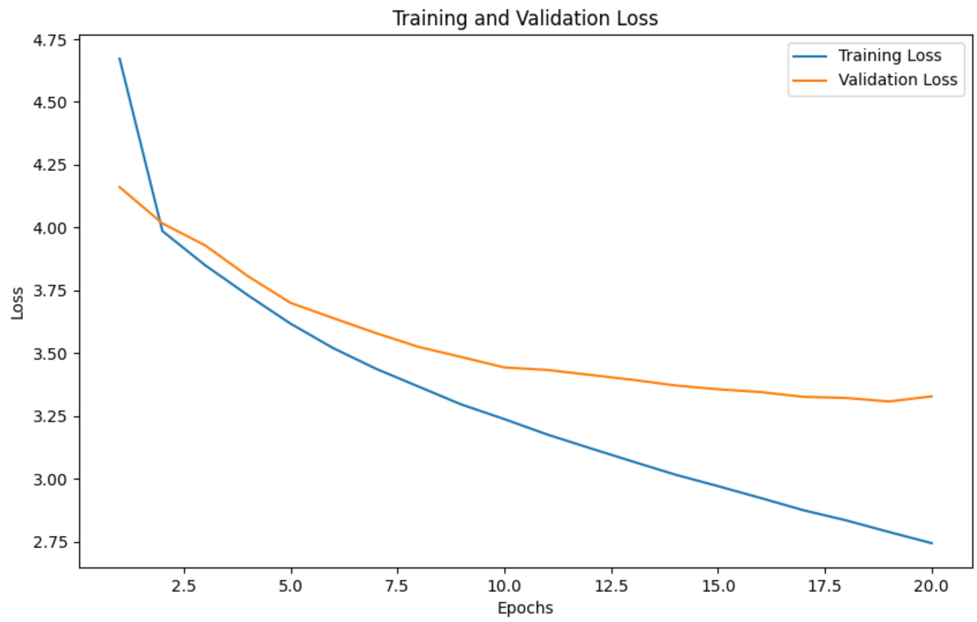


Figure 1: Loss plot for TransfomerLM (task1)

2 Task 2

2.1 Preprocessing Steps

The following steps were applied:

- **Lowercase Conversion:** All input text is converted to lowercase to eliminate case-based discrepancies.
- **Contraction Expansion:** Contractions (e.g., "can't") are expanded to their full forms (e.g., "cannot") using the `contractions` library.
- **Abbreviation Expansion:** Common abbreviations (e.g., "gov.") are replaced with their full forms (e.g., "governor") using a predefined dictionary.
- **URL Removal:** URLs are removed using regular expressions to avoid non-informative tokens.
- **Special Character Removal:** Unnecessary special characters are removed while preserving essential punctuation for readability.
- **Whitespace Normalization:** Extra whitespace is removed to standardize the input text.

2.2 Model Architecture and Hyperparameters

The primary model used in this task is BART.

Model Architecture

- **BART:** An encoder-decoder architecture that efficiently handles tasks like summarization and, in this case, claim normalization.
- **Tokenizer:** The model utilizes the associated BART tokenizer which handles tokenization, padding, and truncation.

Hyperparameters

Key hyperparameters include:

- Learning Rate: 3×10^{-5}
- Weight Decay: 0.01
- Warmup Steps: 500
- Maximum Input Length: 512 tokens
- Maximum Target Length: 128 tokens
- Batch Size: 8 during training (with a larger batch size for inference when memory permits)
- Number of Training Epochs: 3
- Beam Search: Beam size of 4 is used during generation to improve prediction quality.

2.3 Training and Validation Loss Plots

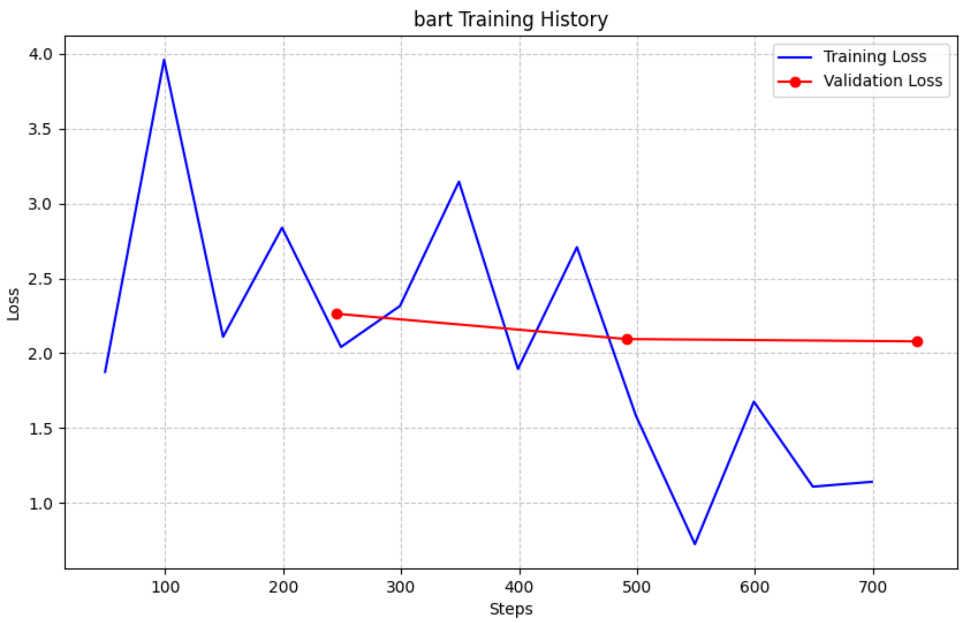


Figure 2: Loss plot for Bart

Test metric	DataLoader 0
val_bertscore	0.6797538399969635
val_bleu4	0.20771007239818573
val_loss	2.1323001384735107
val_rouge1	0.39769190549850464
val_rouge2	0.2669219970703125
val_rouge_l	0.36265984177589417

Figure 3: Bart Metric

Figure 4: Loss plot for T5

Test metric	DataLoader 0
val_bertscore	0.6522139310836792
val_bleu4	0.19351518154144287
val_loss	2.404681682586670
val_rouge1	0.3612382411956787
val_rouge2	0.23798666894435883
val_rouge_l	0.32908788323402405

Figure 5: T5 Metric

2.4 Evaluation Metrics on the Test Set

```
Sample predictions:
Example 1:
  Post: i declare covid 19 over world health organisation boss says coronavirus is no longer a global emerge...
  True claim: the world health organisation has declared the covid 19 pandemic over.
  Predicted: the coronavirus is no longer a global emergency.

Example 2:
  Post: ajike media update, donald trump love for biafra...
  True claim: false donald trump did not call kenya a very corrupt country
  Predicted: ajike media update, donald trump love for biafra

Example 3:
  Post: nobody making under 400,000 will have their taxes raised. period, says joebiden ....
  True claim: biden s tax rate on a family making 75,000 dollars would go from 12 to 25 .
  Predicted: under 400,000 people will have their taxes raised. period, says joe biden.

Example 4:
  Post: winner of 1.28 billion lottery gets 433.7 million after tax. congratulations to the irs on winning t...
  True claim: irs would collect 846 million from the winner of a 1.28 billion lottery
  Predicted: winner of 1.28 billion lottery gets 433.7 million after tax

Example 5:
  Post: it is horrible, look at the fire of today s blast. peshawarblast peshawar peshawarunderattack peshaw...
  True claim: it is horrible, look at the fire of today s blast. peshawarblast peshawar peshawarunderattack peshawarattack
  Predicted: it is horrible, look at the fire of today s blast.


Evaluation metrics on sample predictions:
  ROUGE-L: 0.3380
  BLEU-4: 0.2393
  BERTScore: 0.6626
```

Figure 6: Predictions for Samples in Test Dataset using BART

```
Sample predictions:
Example 1:
  Post: i declare covid 19 over world health organisation boss says coronavirus is no longer a global emerge...
  True claim: the world health organisation has declared the covid 19 pandemic over.
  Predicted: covid 19 over world health organisation boss says coronavirus is no longer a global emergency.

Example 2:
  Post: ajike media update, donald trump love for biafra...
  True claim: false donald trump did not call kenya a very corrupt country
  Predicted: donald trump love for biafra

Example 3:
  Post: nobody making under 400,000 will have their taxes raised. period, says joebiden ....
  True claim: biden s tax rate on a family making 75,000 dollars would go from 12 to 25 .
  Predicted: nobody making under 400,000 will have their taxes raised.

Example 4:
  Post: winner of 1.28 billion lottery gets 433.7 million after tax. congratulations to the irs on winning t...
  True claim: irs would collect 846 million from the winner of a 1.28 billion lottery
  Predicted: winner of 1.28 billion lottery gets 433.7 million after tax.

Example 5:
  Post: it is horrible, look at the fire of today s blast. peshawarblast peshawar peshawarunderattack peshaw...
  True claim: it is horrible, look at the fire of today s blast. peshawarblast peshawar peshawarunderattack peshawarattack
  Predicted: peshawar peshawar attack peshawarattack peshawarattack peshawarattack


Evaluation metrics on sample predictions:
  ROUGE-L: 0.2562
  BLEU-4: 0.0000
  BERTScore: 0.6686
```

Figure 7: Predictions for Samples in Test Dataset using T5

2.5 Comparative Analysis of Model Performance

Two models were considered during experimentation:

- **BART:** Achieved strong performance with consistent loss reduction and high evaluation scores on the test set. Its encoder-decoder architecture proved effective for the normalization task.
- **T5:** Preliminary comparisons suggest that while T5 is efficient, the BART model yielded better results under the specific experimental conditions.

The comparison is illustrated in the metrics bar chart (`plots/model_comparison.png`), which visually contrasts the performance based on ROUGE-L, BLEU-4, and BERTScore.

2.6 Discussion on Resource Constraints and Model Selection

Resource constraints played a significant role in the model selection and training process:

- **Hardware Availability:** The experiments were optimized to utilize available GPU resources. When GPUs were not available, the code automatically fell back to CPU execution.
- **Memory Management:** Batch sizes and maximum sequence lengths were carefully chosen to avoid memory overload, ensuring that the training process was both stable and efficient.
- **Computational Efficiency:** BART was preferred due to its balance between performance and computational demands. Although T5 is a robust alternative, its larger model size and higher computational requirements made it less favorable given the available resources.
- **Scalability:** The training and inference pipelines were designed to process data in batches. This approach not only enhanced scalability but also minimized the risk of exhausting computational resources during experimentation.

3 Task 3

3.1 Preprocessing

- **Dataset Loading:** The dataset is loaded from a TSV file containing columns for `pid`, `text`, `explanation`, and `target_of_sarcasm`. Additional image descriptions and detected objects are loaded from pickle files.
- **Text Preprocessing:** For each sample, the text components (`sarcasm_target`, `text`, image description, and detected objects) are concatenated into a single input string.
- **Image Preprocessing:** Images are loaded from a specified directory (with `.jpg` extension) and transformed using resizing to 224×224, conversion to tensor, and normalization (mean = [0.5, 0.5, 0.5] and std = [0.5, 0.5, 0.5]).
- **Tokenization:** The concatenated text inputs and target explanations are tokenized using the `BartTokenizer`, with padding and truncation applied (maximum length of 256 for inputs and 64 for targets).
- **Custom Collation:** A custom collate function is used to batch the tokenized inputs, targets, and images.

3.2 Model Architecture & Hyperparameters

- **Model Components:**
  - **Text Encoder-Decoder:** `BartForConditionalGeneration` (using the `facebook/bart-base` checkpoint) is used for generating the explanation text.
  - **Image Encoder:** `ViTModel` (using the `google/vit-base-patch16-224` checkpoint) extracts image features.
  - **Fusion:** A linear fusion layer projects the ViT global image feature to the dimensionality of BART's hidden states. The projected image feature is added (broadcast) to each token embedding from BART's encoder.
- **Hyperparameters:**
  - **Input and Target Tokenization:** Maximum token length of 256 for input text and 64 for target explanation.
  - **Training:**
    - Optimizer: AdamW with a learning rate of 5×10^{-5}.
    - Batch size: 8.
    - Number of epochs: 3.
  - **Generation Settings (Validation):**
    - Decoding: Beam search with sampling.
    - Number of beams: 4.
    - Maximum generation length: 64 tokens.
    - Temperature: 1.5.
    - Top-K: 50.
    - Repetition penalty: 2.0.
    - No-repeat n-gram size: 3.
    - Early stopping enabled.
  - **Post-processing:** A custom function is applied to remove repeated sentences and collapse repeated tokens.

3.3 Evaluation

Training Loss and Validation Loss:

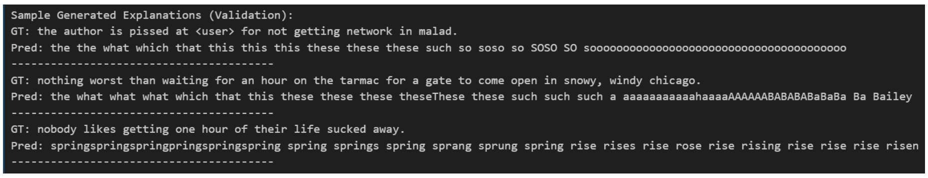
Epoch	**Training Loss**	**Validation Loss**
Epoch 1	3.1209	0.1868
Epoch 2	0.1065	0.0101
Epoch 3	0.0321	0.0029

Evaluation Metrics on Validation Set:

Metric	**Epoch 1**	**Epoch 2**	**Epoch 3**
ROUGE-1	0.0665	0.0379	0.0375
ROUGE-2	0.0003	0.0000	0.0005
ROUGE-L	0.0627	0.0369	0.0359
BLEU	0.0078	0.0046	0.0046
METEOR	0.0411	0.0223	0.0229
BERTScore_F1	0.7759	0.7685	0.7611

Sample Generated Explanations (Validation)

- **GT:** *the author is pissed at <user>for not getting network in malad.*
  Pred: *the the what which that this this this these such so soso so SOSO SO soooooooooooooooooooooooooooooooooooo*
- **GT:** *nothing worst than waiting for an hour on the tarmac for a gate to come open in snowy, windy chicago.*
  Pred: *the what what what which that this these these these These these such such such a aaaaaaaaaaahaaaaAAAAAABABABABaBaBa Ba Bailey*
- **GT:** *nobody likes getting one hour of their life sucked away.*
  Pred: *springspringspringpringspringspring spring springs spring sprang sprung spring rise rises rise rose rise rising rise rise rise risen rise up above below above over above above above lower bottom*



```
Sample Generated Explanations (Validation):
GT: the author is pissed at <user> for not getting network in malad.
Pred: the the what which that this this this these these these such so soso so SOSO SO soooooooooooooooooooooooooooooooooooooo
----------------------------------------
GT: nothing worst than waiting for an hour on the tarmac for a gate to come open in snowy, windy chicago.
Pred: the what what what which that this these these these theseThese these such such such a aaaaaaaaaaahaaaaAAAAAABABABABaBaBa Ba Bailey
----------------------------------------
GT: nobody likes getting one hour of their life sucked away.
Pred: springspringspringpringspringspring spring springs spring sprang sprung spring rise rises rise rose rise rising rise rise rise risen
----------------------------------------
```

4 Individual Contribution

- Himanshu Raj: Task 1 implementation and report
- Ishita: Task 2 implementation and report
- Ritika Thakur: Task 3 implementation and report

5 References

- Target-Augmented Shared Fusion-based Multimodal Sarcasm Explanation Generation