CBU5201_miniproject

December 24, 2024

1 HOLMES: HOlistic Lightweight Language Model for Emergent Story Authentication

2 1 Author

Student Name: Junming Lin

Student ID: 221166954

3 2 Problem formulation

This project aims to address the challenging problem of deception detection in narrated stories. Specifically, I will develop a **LLM** system that can analyze audio recordings of 3-5 minutes in duration and predict whether the story being told is true or deceptive. This problem falls within the broader domain of natural language processing (NLP), with potential applications in areas such as journalism, content moderation, and security, which is super relevant today with all the misinformation floating around.

3.1 2.1 Definition of the Problem

The task is a binary classification problem in deception detection through audio storytelling. Given an audio recording of a narrated story (3-5 minutes in duration), we aim to predict whether the story being told is truthful or deceptive. Formally:

Let $\mathcal{X}$ be the input space of audio recordings, and $\mathcal{Y} = \{\text{True Story}, \text{Deceptive Story}\}$ be the output space.

Given the dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ where $N = 100$, $x_i \in \mathcal{X}$ represents the i-th audio recording, and $y_i \in \mathcal{Y}$ denotes its corresponding label.

Our objective is to learn a classification function $f : \mathcal{X} \to \mathcal{Y}$ that captures the underlying structural patterns distinguishing true stories from deceptive ones in the general population. Rather than simply minimizing empirical risk on our limited training data, we aim to minimize the expected risk over the true population distribution $P(X, Y)$:

$$\min_f \mathbb{E}_{(x,y)\sim P(X,Y)}[\mathcal{L}(f(x), y)]$$

where $\mathcal{L}$ is an appropriate loss function.

3.2 2.2 Challenges and Difficulties

This deception detection task presents several significant challenges:

3.2.1 2.2.1 Limited Dataset Size

With only 100 labeled stories available, the dataset size poses significant constraints on model training. This limitation could lead to:

$$P(\hat{\theta}) \neq P(\theta_{\text{true}})$$

where $\hat{\theta}$ represents our learned model parameters and θ_{true} represents the true underlying patterns. To address this:

Use other corpus and dataset for pretraining and fine-tuning, in order to enhancing general understanding ability of LLM

3.2.2 2.2.2 Long-Sequence Understanding

Processing 3-5 minute stories poses challenges for traditional sequence models. RNNs and LSTMs suffer from vanishing gradients and information loss over long sequences:

$$h_t = \text{LSTM}(x_t, h_{t-1}) \quad \text{where} \quad \|\frac{\partial h_t}{\partial h_k}\| \to 0 \quad \text{as} \quad |t-k| \to \infty$$

To address this, we employ Transformer architectures with self-attention:

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

This allows direct modeling of long-range dependencies.

3.2.3 2.2.3 Multimodal Feature Integration

The task requires Multimodal Alignment of: - Semantic content (what is said) - Prosodic features (how it's said) - Acoustic markers of deception …

This necessitates the model architecture or system that can effectively combine:

$$f(x) = g(f_{\text{semantic}}(x), f_{\text{acoustic}}(x), f_{\text{prosodic}}(x))$$

where $g(\cdot)$ learns to weight and integrate these different modalities for the final classification decision.

4 3 Methodology

In this project, I tackled the fascinating challenge by developing a machine learning system to detect deception in narrated stories using a combination of advanced architecture. Specifically, i proposed **HOLMES**, a novel LLM system designed to detect deception in narrated stories. This lightweight system, with only **26M** parameters' LLM backbone(HOLMES-26M) **trained from scratch**. HOLMES leverages the power of Language Models and speech recognition technology to determine whether a 3-5 minute audio recording contains a true or fabricated story.

4.1 3.1 HOLMES System Components

HOLMES is a two-stage system for deception detection in narrated stories, consisting of:

Whisper: OpenAI's state-of-the-art ASR model, to transcribe audio stories into text format. - Input: 3-5 minute audio recordings from MLEnd Deception Dataset - Output: Text transcripts for each audio recording - Advantage: Highly accurate speech-to-text conversion while preserving linguistic nuances

HOLMES-26M: our specialized 26M parameter language model. - Architecture: Decoder-only transformer (detailed in Section 3.2) - Input: Transcribed text from Whisper - Output: Binary classification with probability score - Key Features: - Lightweight design (26M parameters) - Trained from scratch for deception detection - Optimized for story comprehension and truthfulness assessment

HOLMES-26M, which is based on the transformer architecture. The model is designed for text generation and follows a decoder-only structure similar to GPT-3. Section 3.3 will detail the architecture of **HOLMES-26M**, including specific parameters, nerual network choices.

4.2 3.2 Methodology Overview

The core of our methodology involves a two-stage process:

1. **Speech-to-Text Transcription:** Utilizing OpenAI's Whisper, a state-of-the-art automatic speech recognition (ASR) model, we first transcribe the audio recordings of the stories into text format. This process ensures high accuracy in converting spoken language into written text, capturing the nuances of the narratives.
2. **Truthfulness Classification with Fine-tuned HOLMES-26M:** The transcribed text is then fed into our specialized language model, HOLMES-26M. This model, with its decoder-only transformer architecture, has been fine-tuned specifically for the task of deception detection. The fine-tuning process involves training HOLMES-26M on a set of MLEnd dataset, enabling it to learn patterns and linguistic cues indicative of deception. The model outputs a probability score representing its confidence in the story's truthfulness.

In essence, the methodology leverages the strengths of both Whisper and HOLMES-26M to create a pipeline that first converts audio to text and then analyzes the text to determine the likelihood of deception. Further details about data preprocessing, model training&validation, and evaluation metrics will be detailed in the following sections.

4.3 3.3 HOLMES-26M's Architecture

The LLM architecture and training code are adapted from: https://github.com/jingyaogong/minimind

HOLMES-26M is based on the transformer architecture, it consists of the following key components:

1. **Token Embedding Layer(Tokenizer):** Maps input tokens to dense vector representations.
2. **Transformer Layers(Transformer architerture):** A stack of transformer blocks, each containing a Multi-Head Self-Attention (MHSA) mechanism and a FeedForward Network (FFN).
3. **Normalization Layer:** Applies RMSNorm for data normalization within each transformer block.
4. **Output Layer:** A linear layer that projects the final hidden states to the vocabulary space, followed by a Softmax function to produce probability distributions over the vocabulary.

The model adopts a pretrain approach similar to GPT-3, applying RMSNorm to the input of each sub-layer within the transformer blocks. It utilizes the SwiGLU activation function instead of ReLU for improved performance and incorporates Rotary Positional Embeddings (RoPE) instead of absolute positional embeddings.

4.3.1 3.3.1 Token Embedding Layer(Tokenizer)

The token embedding layer converts input tokens into continuous vector representations.

Parameters:

- **Vocabulary Size (vocab_size):** 6400
- **Embedding Dimension (dim):** 512

Given an input token t, the embedding layer retrieves the corresponding embedding vector e_t from the embedding matrix $E \in \mathbb{R}^{vocab\_size \times dim}$.

$$e_t = E[t]$$

The embedding matrix E is a learnable parameter of the model, and it shares weights with the output layer's weight matrix.

4.3.2 3.3.2 Transformer Layers

The core of **HOLMES-26M** is a stack of $n\_layers = 8$ transformer layers. Each layer consists of an MHSA sub-layer and an FFN sub-layer.

3.3.2.1 Multi-Head Self-Attention (MHSA) The MHSA mechanism allows the model to attend to different parts of the input sequence and capture contextual relationships.

Parameters:

- **Number of Heads (n_heads):** 16
- **Number of Key-Value Heads (n_kv_heads):** 8 (default, can be equal to n_heads)
- **Head Dimension (head_dim):** $dim/n\_heads = 512/16 = 32$

- **Dropout Rate (dropout):** 0.0
- **Flash Attention (flash_attn):** True (Utilize efficient attention implementation if available)

1. **Linear Projections:** The input $x \in \mathbb{R}^{seq\_len \times dim}$ is projected into query Q, key K, and value V matrices using learnable weight matrices $W_q \in \mathbb{R}^{dim \times n\_heads \cdot head\_dim}$, $W_k \in \mathbb{R}^{dim \times n\_kv\_heads \cdot head\_dim}$, and $W_v \in \mathbb{R}^{dim \times n\_kv\_heads \cdot head\_dim}$.

$$Q = xW_q$$
$$K = xW_k$$
$$V = xW_v$$

2. **Rotary Positional Embeddings (RoPE):** RoPE is applied to the query and key matrices to incorporate positional information. This is achieved by element-wise multiplication of the query and key with complex exponentials that represent rotations.

$$q'_m = f_q(x, m) = q_m e^{im\theta}$$
$$k'_n = f_k(x, n) = k_n e^{in\theta}$$

where m, n are sequence position indices, and $\theta_i = 10000^{-2(i-1)/d}$, $i \in [1, 2, ..., d/2]$, and d is the dimension.

3. **Attention Scores:** The attention scores are computed as the dot product of the rotated query and key matrices, scaled by the inverse square root of the head dimension.

$$scores = \frac{QK^T}{\sqrt{head\_dim}}$$

4. **Masking:** A causal mask is applied to the attention scores to prevent the model from attending to future tokens.

$$masked\_scores = scores + M$$

where $M_{ij} = -\infty$ if $i > j$ and 0 otherwise.

5. **Softmax:** The masked scores are passed through a Softmax function to obtain attention weights.

$$attention\_weights = softmax(masked\_scores)$$

6. **Weighted Value:** The attention weights are multiplied by the value matrix to obtain the weighted value.

$$output = attention\_weights V$$

7. **Output Projection:** The concatenated outputs from all heads are projected back to the model dimension using a learnable weight matrix $W_o \in \mathbb{R}^{n\_heads \cdot head\_dim \times dim}$.

$$MHSA(x) = outputW_o$$

8. **KV Cache:** For efficient inference, the key and value matrices are cached to avoid redundant computations.

3.3.2.2 FeedForward Network (FFN) The FFN provides non-linearity and further transforms the output of the MHSA sub-layer.

Parameters:

- **Hidden Dimension (hidden_dim):** Calculated as $int(2 * (4 * dim)/3)$, rounded up to the nearest multiple of `multiple_of` (default 64). For example, with `dim=512`, `hidden_dim` will be 1408.
- **Dropout Rate (dropout):** 0.0

The FFN consists of two linear layers with a SwiGLU activation function in between.

$$FFN(x) = (W_2(SwiGLU(W_1x) \odot W_3x))$$

where $W_1 \in \mathbb{R}^{dim \times hidden\_dim}$, $W_2 \in \mathbb{R}^{hidden\_dim \times dim}$, $W_3 \in \mathbb{R}^{dim \times hidden\_dim}$ are learnable weight matrices, and $\odot$ denotes element-wise multiplication. The SwiGLU activation is defined as:

$$SwiGLU(x) = x \cdot sigmoid(\beta x)$$

where β is a learnable parameter, it is set to 1 here.

4.3.3 3.3.3 Normalization Layer

HOLMES-26M uses RMSNorm for pre-normalization within each transformer block.

Parameters:

- **Dimension (dim):** 512
- **Epsilon (eps):** 1e-5

$$RMSNorm(x) = \frac{x}{\sqrt{\frac{1}{dim}\sum_{i=1}^{dim} x_i^2 + eps}} \cdot \gamma$$

where $\gamma \in \mathbb{R}^{dim}$ is a learnable gain parameter.

4.3.4 3.3.4 Output Layer

The output layer projects the final hidden states to the vocabulary space.

Parameters:

- **Output Dimension:** Equal to the vocabulary size (6400)

Mathematical Formulation:

$$logits = hW_{out}$$

where $h \in \mathbb{R}^{seq\_len \times dim}$ is the output of the last transformer layer, and $W_{out} \in \mathbb{R}^{dim \times vocab\_size}$ is a learnable weight matrix. A Softmax function is then applied to the logits to obtain probability distributions over the vocabulary.

$$probabilities = softmax(logits)$$

The weight matrix W_{out} is shared with the embedding matrix E.

4.4 3.4 Training Task

The LLM architecture and training code are adapted from: https://github.com/jingyaogong/minimind

The training of **HOLMES-26M** involves three distinct phases: **pretraining**, **full supervised fine-tuning (SFT)**, and **cross-validated SFT**. Each phase utilizes a specific dataset and objective, contributing to the model's overall ability to detect deception in narrated stories.

4.4.1 3.4.1 Pretraining Phase

Objective: To train the model to predict the next token in a sequence, enabling it to learn general language understanding and generation capabilities.

Dataset: Seq-Monkey, a comprehensive and high-quality Chinese corpus comprising approximately 10 billion tokens. This dataset is sourced from diverse domains, including web pages, encyclopedias, blogs, open-source code, and books. It has undergone rigorous cleaning and deduplication to ensure data quality and representativeness.

Link: https://github.com/mobvoi/seq-monkey-data

Training Procedure:

1. **Data Preparation:** The Seq-Monkey dataset is preprocessed into sequences of fixed length (`max_seq_len = 2048` tokens), which is defined in `LMConfig.py`. Each sequence is tokenized using the **HOLMES-26M** tokenizer.
2. **Model Input:** For each sequence, the model takes the first `n-1` tokens as input (denoted as `X`) and the last `n-1` tokens as the target output (denoted as `Y`).
3. **Loss Function:** The model is trained to minimize the cross-entropy loss between the predicted token probabilities and the actual next token. The loss is calculated as follows:

$$\text{Loss} = -\frac{1}{\sum_{i=1}^{N} m_i} \sum_{i=1}^{N} m_i \cdot \log(p(y_i|x_{1:i}))$$

where:

- N is the sequence length.
- $x_{1:i}$ represents the input tokens from position 1 to i.

- y_i is the target token at position i.
- $p(y_i|x_{1:i})$ is the predicted probability of the target token given the input tokens.
- m_i is the mask value at position i (1 if the token is not padding, 0 otherwise).

4. **Optimization:**
   - The model is trained using the **Adam optimizer** with a learning rate of `2e-4`.
   - **Gradient accumulation** is employed with `8` accumulation steps to simulate larger batch sizes.
   - **Gradient clipping** is applied with a threshold of `1.0` to prevent exploding gradients.
   - Learning rate is held constant during pretraining.
   - **Mixed-precision training** (using `bfloat16`) is used to accelerate training and reduce memory usage.
   - A **LossTracker** class is used to track and visualize the loss curve during training.
5. **Evaluation:** The model's performance is periodically evaluated by monitoring the loss on the training set.
6. **Implementation Details:**
   - The pretraining process is implemented in `1-pretrain.py`.
   - The model is trained for 1 epochs.
7. **Implementation:** the whole training process can be found at https://wandb.ai/mjuicem3-beijing-university-of-posts-and-telecommunications/HOLMES-Pretrain?nw=nwusermjuicem3

[ ]:
```
!torchrun --nproc_per_node 2 1-pretrain.py
```

```
Total LLM parameters: 26.878M
Epoch:[0/1](0/41914) loss:8.861 lr:0.0002000 epoch_Time:1089.0min:
Epoch:[0/1](100/41914) loss:7.439 lr:0.0002000 epoch_Time:108.0min:
Epoch:[0/1](200/41914) loss:6.955 lr:0.0002000 epoch_Time:103.0min:
Epoch:[0/1](300/41914) loss:6.456 lr:0.0002000 epoch_Time:101.0min:
Epoch:[0/1](400/41914) loss:6.025 lr:0.0002000 epoch_Time:100.0min:
Epoch:[0/1](500/41914) loss:5.740 lr:0.0001999 epoch_Time:99.0min:
Epoch:[0/1](600/41914) loss:5.521 lr:0.0001999 epoch_Time:99.0min:
Epoch:[0/1](700/41914) loss:5.315 lr:0.0001999 epoch_Time:99.0min:
Epoch:[0/1](800/41914) loss:5.132 lr:0.0001998 epoch_Time:98.0min:
Epoch:[0/1](900/41914) loss:5.025 lr:0.0001998 epoch_Time:97.0min:
Epoch:[0/1](1000/41914) loss:5.023 lr:0.0001997 epoch_Time:97.0min:
Epoch:[0/1](1100/41914) loss:4.833 lr:0.0001997 epoch_Time:97.0min:
Epoch:[0/1](1200/41914) loss:4.667 lr:0.0001996 epoch_Time:97.0min:
Epoch:[0/1](1300/41914) loss:4.797 lr:0.0001996 epoch_Time:96.0min:
Epoch:[0/1](1400/41914) loss:4.543 lr:0.0001995 epoch_Time:96.0min:
Epoch:[0/1](1500/41914) loss:4.419 lr:0.0001994 epoch_Time:96.0min:
Epoch:[0/1](1600/41914) loss:4.417 lr:0.0001994 epoch_Time:96.0min:
Epoch:[0/1](1700/41914) loss:4.365 lr:0.0001993 epoch_Time:95.0min:
Epoch:[0/1](1800/41914) loss:4.402 lr:0.0001992 epoch_Time:95.0min:
Epoch:[0/1](1900/41914) loss:4.274 lr:0.0001991 epoch_Time:95.0min:
Epoch:[0/1](2000/41914) loss:4.013 lr:0.0001990 epoch_Time:95.0min:
```

```
Epoch:[0/1](2100/41914) loss:4.222 lr:0.0001989 epoch_Time:95.0min:
Epoch:[0/1](2200/41914) loss:4.072 lr:0.0001988 epoch_Time:94.0min:
Epoch:[0/1](2300/41914) loss:4.046 lr:0.0001987 epoch_Time:94.0min:
Epoch:[0/1](2400/41914) loss:4.138 lr:0.0001985 epoch_Time:94.0min:
Epoch:[0/1](2500/41914) loss:3.937 lr:0.0001984 epoch_Time:94.0min:
Epoch:[0/1](2600/41914) loss:3.905 lr:0.0001983 epoch_Time:93.0min:
Epoch:[0/1](2700/41914) loss:3.907 lr:0.0001982 epoch_Time:93.0min:
Epoch:[0/1](2800/41914) loss:3.949 lr:0.0001980 epoch_Time:93.0min:
Epoch:[0/1](2900/41914) loss:3.950 lr:0.0001979 epoch_Time:93.0min:
Epoch:[0/1](3000/41914) loss:3.906 lr:0.0001977 epoch_Time:92.0min:
Epoch:[0/1](3100/41914) loss:3.929 lr:0.0001976 epoch_Time:92.0min:
Epoch:[0/1](3200/41914) loss:3.815 lr:0.0001974 epoch_Time:92.0min:
Epoch:[0/1](3300/41914) loss:3.909 lr:0.0001973 epoch_Time:92.0min:
Epoch:[0/1](3400/41914) loss:3.739 lr:0.0001971 epoch_Time:91.0min:
Epoch:[0/1](3500/41914) loss:3.827 lr:0.0001969 epoch_Time:91.0min:
Epoch:[0/1](3600/41914) loss:3.724 lr:0.0001967 epoch_Time:91.0min:
Epoch:[0/1](3700/41914) loss:3.664 lr:0.0001966 epoch_Time:91.0min:
Epoch:[0/1](3800/41914) loss:3.675 lr:0.0001964 epoch_Time:91.0min:
Epoch:[0/1](3900/41914) loss:3.689 lr:0.0001962 epoch_Time:90.0min:
Epoch:[0/1](4000/41914) loss:3.659 lr:0.0001960 epoch_Time:90.0min:
Epoch:[0/1](4100/41914) loss:3.616 lr:0.0001958 epoch_Time:90.0min:
Epoch:[0/1](4200/41914) loss:3.670 lr:0.0001956 epoch_Time:90.0min:
Epoch:[0/1](4300/41914) loss:3.648 lr:0.0001954 epoch_Time:89.0min:
Epoch:[0/1](4400/41914) loss:3.712 lr:0.0001951 epoch_Time:89.0min:
Epoch:[0/1](4500/41914) loss:3.567 lr:0.0001949 epoch_Time:89.0min:
Epoch:[0/1](4600/41914) loss:3.681 lr:0.0001947 epoch_Time:89.0min:
Epoch:[0/1](4700/41914) loss:3.651 lr:0.0001945 epoch_Time:88.0min:
Epoch:[0/1](4800/41914) loss:3.589 lr:0.0001942 epoch_Time:88.0min:
Epoch:[0/1](4900/41914) loss:3.471 lr:0.0001940 epoch_Time:88.0min:
Epoch:[0/1](5000/41914) loss:3.468 lr:0.0001938 epoch_Time:88.0min:
Epoch:[0/1](5100/41914) loss:3.490 lr:0.0001935 epoch_Time:87.0min:
Epoch:[0/1](5200/41914) loss:3.643 lr:0.0001933 epoch_Time:87.0min:
Epoch:[0/1](5300/41914) loss:3.375 lr:0.0001930 epoch_Time:87.0min:
Epoch:[0/1](5400/41914) loss:3.441 lr:0.0001927 epoch_Time:87.0min:
Epoch:[0/1](5500/41914) loss:3.430 lr:0.0001925 epoch_Time:86.0min:
Epoch:[0/1](5600/41914) loss:3.322 lr:0.0001922 epoch_Time:86.0min:
Epoch:[0/1](5700/41914) loss:3.401 lr:0.0001919 epoch_Time:86.0min:
Epoch:[0/1](5800/41914) loss:3.452 lr:0.0001916 epoch_Time:86.0min:
Epoch:[0/1](5900/41914) loss:3.404 lr:0.0001913 epoch_Time:86.0min:
Epoch:[0/1](6000/41914) loss:3.433 lr:0.0001911 epoch_Time:85.0min:
Epoch:[0/1](6100/41914) loss:3.297 lr:0.0001908 epoch_Time:85.0min:
Epoch:[0/1](6200/41914) loss:3.427 lr:0.0001905 epoch_Time:85.0min:
Epoch:[0/1](6300/41914) loss:3.241 lr:0.0001902 epoch_Time:85.0min:
Epoch:[0/1](6400/41914) loss:3.551 lr:0.0001898 epoch_Time:84.0min:
Epoch:[0/1](6500/41914) loss:3.484 lr:0.0001895 epoch_Time:84.0min:
Epoch:[0/1](6600/41914) loss:3.283 lr:0.0001892 epoch_Time:84.0min:
Epoch:[0/1](6700/41914) loss:3.292 lr:0.0001889 epoch_Time:84.0min:
Epoch:[0/1](6800/41914) loss:3.283 lr:0.0001886 epoch_Time:83.0min:
```

```
Epoch:[0/1](6900/41914) loss:3.165 lr:0.0001882 epoch_Time:83.0min:
Epoch:[0/1](7000/41914) loss:3.259 lr:0.0001879 epoch_Time:83.0min:
Epoch:[0/1](7100/41914) loss:3.312 lr:0.0001876 epoch_Time:83.0min:
Epoch:[0/1](7200/41914) loss:3.388 lr:0.0001872 epoch_Time:82.0min:
Epoch:[0/1](7300/41914) loss:3.293 lr:0.0001869 epoch_Time:82.0min:
Epoch:[0/1](7400/41914) loss:3.139 lr:0.0001865 epoch_Time:82.0min:
Epoch:[0/1](7500/41914) loss:3.373 lr:0.0001862 epoch_Time:82.0min:
Epoch:[0/1](7600/41914) loss:3.127 lr:0.0001858 epoch_Time:82.0min:
Epoch:[0/1](7700/41914) loss:3.313 lr:0.0001854 epoch_Time:81.0min:
Epoch:[0/1](7800/41914) loss:3.177 lr:0.0001851 epoch_Time:81.0min:
Epoch:[0/1](7900/41914) loss:3.392 lr:0.0001847 epoch_Time:81.0min:
Epoch:[0/1](8000/41914) loss:3.191 lr:0.0001843 epoch_Time:81.0min:
Epoch:[0/1](8100/41914) loss:3.230 lr:0.0001839 epoch_Time:80.0min:
Epoch:[0/1](8200/41914) loss:3.272 lr:0.0001835 epoch_Time:80.0min:
Epoch:[0/1](8300/41914) loss:3.206 lr:0.0001831 epoch_Time:80.0min:
Epoch:[0/1](8400/41914) loss:3.092 lr:0.0001827 epoch_Time:80.0min:
Epoch:[0/1](8500/41914) loss:3.263 lr:0.0001823 epoch_Time:79.0min:
Epoch:[0/1](8600/41914) loss:3.150 lr:0.0001819 epoch_Time:79.0min:
Epoch:[0/1](8700/41914) loss:3.382 lr:0.0001815 epoch_Time:79.0min:
Epoch:[0/1](8800/41914) loss:3.150 lr:0.0001811 epoch_Time:79.0min:
Epoch:[0/1](8900/41914) loss:3.061 lr:0.0001807 epoch_Time:78.0min:
Epoch:[0/1](9000/41914) loss:3.232 lr:0.0001803 epoch_Time:78.0min:
Epoch:[0/1](9100/41914) loss:3.053 lr:0.0001799 epoch_Time:78.0min:
Epoch:[0/1](9200/41914) loss:3.094 lr:0.0001794 epoch_Time:78.0min:
Epoch:[0/1](9300/41914) loss:3.257 lr:0.0001790 epoch_Time:78.0min:
Epoch:[0/1](9400/41914) loss:3.117 lr:0.0001786 epoch_Time:77.0min:
Epoch:[0/1](9500/41914) loss:3.139 lr:0.0001781 epoch_Time:77.0min:
Epoch:[0/1](9600/41914) loss:3.034 lr:0.0001777 epoch_Time:77.0min:
Epoch:[0/1](9700/41914) loss:3.187 lr:0.0001772 epoch_Time:77.0min:
Epoch:[0/1](9800/41914) loss:2.957 lr:0.0001768 epoch_Time:76.0min:
Epoch:[0/1](9900/41914) loss:3.081 lr:0.0001763 epoch_Time:76.0min:
Epoch:[0/1](10000/41914) loss:3.048 lr:0.0001759 epoch_Time:76.0min:
Epoch:[0/1](10100/41914) loss:3.090 lr:0.0001754 epoch_Time:76.0min:
Epoch:[0/1](10200/41914) loss:3.137 lr:0.0001750 epoch_Time:75.0min:
Epoch:[0/1](10300/41914) loss:3.164 lr:0.0001745 epoch_Time:75.0min:
Epoch:[0/1](10400/41914) loss:2.931 lr:0.0001740 epoch_Time:75.0min:
Epoch:[0/1](10500/41914) loss:3.082 lr:0.0001735 epoch_Time:75.0min:
Epoch:[0/1](10600/41914) loss:3.057 lr:0.0001731 epoch_Time:74.0min:
Epoch:[0/1](10700/41914) loss:3.002 lr:0.0001726 epoch_Time:74.0min:
Epoch:[0/1](10800/41914) loss:3.066 lr:0.0001721 epoch_Time:74.0min:
Epoch:[0/1](10900/41914) loss:2.947 lr:0.0001716 epoch_Time:74.0min:
Epoch:[0/1](11000/41914) loss:3.050 lr:0.0001711 epoch_Time:74.0min:
Epoch:[0/1](11100/41914) loss:3.046 lr:0.0001706 epoch_Time:73.0min:
Epoch:[0/1](11200/41914) loss:3.143 lr:0.0001701 epoch_Time:73.0min:
Epoch:[0/1](11300/41914) loss:3.079 lr:0.0001696 epoch_Time:73.0min:
Epoch:[0/1](11400/41914) loss:3.040 lr:0.0001691 epoch_Time:73.0min:
Epoch:[0/1](11500/41914) loss:3.032 lr:0.0001686 epoch_Time:72.0min:
Epoch:[0/1](11600/41914) loss:3.067 lr:0.0001681 epoch_Time:72.0min:
```

```
Epoch:[0/1](11700/41914) loss:3.090 lr:0.0001676 epoch_Time:72.0min:
Epoch:[0/1](11800/41914) loss:3.059 lr:0.0001670 epoch_Time:72.0min:
Epoch:[0/1](11900/41914) loss:2.914 lr:0.0001665 epoch_Time:71.0min:
Epoch:[0/1](12000/41914) loss:3.071 lr:0.0001660 epoch_Time:71.0min:
Epoch:[0/1](12100/41914) loss:3.042 lr:0.0001655 epoch_Time:71.0min:
Epoch:[0/1](12200/41914) loss:2.969 lr:0.0001649 epoch_Time:71.0min:
Epoch:[0/1](12300/41914) loss:2.988 lr:0.0001644 epoch_Time:70.0min:
Epoch:[0/1](12400/41914) loss:3.092 lr:0.0001638 epoch_Time:70.0min:
Epoch:[0/1](12500/41914) loss:2.949 lr:0.0001633 epoch_Time:70.0min:
Epoch:[0/1](12600/41914) loss:3.024 lr:0.0001628 epoch_Time:70.0min:
Epoch:[0/1](12700/41914) loss:3.134 lr:0.0001622 epoch_Time:69.0min:
Epoch:[0/1](12800/41914) loss:3.069 lr:0.0001617 epoch_Time:69.0min:
Epoch:[0/1](12900/41914) loss:2.972 lr:0.0001611 epoch_Time:69.0min:
Epoch:[0/1](13000/41914) loss:2.869 lr:0.0001606 epoch_Time:69.0min:
Epoch:[0/1](13100/41914) loss:3.013 lr:0.0001600 epoch_Time:69.0min:
Epoch:[0/1](13200/41914) loss:2.937 lr:0.0001594 epoch_Time:68.0min:
Epoch:[0/1](13300/41914) loss:3.007 lr:0.0001589 epoch_Time:68.0min:
Epoch:[0/1](13400/41914) loss:3.037 lr:0.0001583 epoch_Time:68.0min:
Epoch:[0/1](13500/41914) loss:3.032 lr:0.0001577 epoch_Time:68.0min:
Epoch:[0/1](13600/41914) loss:2.926 lr:0.0001572 epoch_Time:67.0min:
Epoch:[0/1](13700/41914) loss:2.938 lr:0.0001566 epoch_Time:67.0min:
Epoch:[0/1](13800/41914) loss:3.034 lr:0.0001560 epoch_Time:67.0min:
Epoch:[0/1](13900/41914) loss:2.901 lr:0.0001554 epoch_Time:67.0min:
Epoch:[0/1](14000/41914) loss:3.042 lr:0.0001548 epoch_Time:66.0min:
Epoch:[0/1](14100/41914) loss:2.984 lr:0.0001542 epoch_Time:66.0min:
Epoch:[0/1](14200/41914) loss:2.989 lr:0.0001537 epoch_Time:66.0min:
Epoch:[0/1](14300/41914) loss:2.839 lr:0.0001531 epoch_Time:66.0min:
Epoch:[0/1](14400/41914) loss:2.872 lr:0.0001525 epoch_Time:65.0min:
Epoch:[0/1](14500/41914) loss:2.988 lr:0.0001519 epoch_Time:64.0min:
Epoch:[0/1](14600/41914) loss:2.939 lr:0.0001513 epoch_Time:64.0min:
Epoch:[0/1](14700/41914) loss:2.959 lr:0.0001507 epoch_Time:64.0min:
Epoch:[0/1](14800/41914) loss:2.942 lr:0.0001501 epoch_Time:64.0min:
Epoch:[0/1](14900/41914) loss:3.048 lr:0.0001495 epoch_Time:63.0min:
Epoch:[0/1](15000/41914) loss:2.942 lr:0.0001489 epoch_Time:63.0min:
Epoch:[0/1](15100/41914) loss:2.918 lr:0.0001483 epoch_Time:63.0min:
Epoch:[0/1](15200/41914) loss:3.010 lr:0.0001476 epoch_Time:63.0min:
Epoch:[0/1](15300/41914) loss:2.790 lr:0.0001470 epoch_Time:62.0min:
Epoch:[0/1](15400/41914) loss:2.772 lr:0.0001464 epoch_Time:62.0min:
Epoch:[0/1](15500/41914) loss:2.868 lr:0.0001458 epoch_Time:62.0min:
Epoch:[0/1](15600/41914) loss:3.024 lr:0.0001452 epoch_Time:62.0min:
Epoch:[0/1](15700/41914) loss:2.844 lr:0.0001446 epoch_Time:61.0min:
Epoch:[0/1](15800/41914) loss:2.983 lr:0.0001439 epoch_Time:61.0min:
Epoch:[0/1](15900/41914) loss:2.956 lr:0.0001433 epoch_Time:61.0min:
Epoch:[0/1](16000/41914) loss:3.019 lr:0.0001427 epoch_Time:61.0min:
Epoch:[0/1](16100/41914) loss:3.023 lr:0.0001420 epoch_Time:60.0min:
Epoch:[0/1](16200/41914) loss:2.935 lr:0.0001414 epoch_Time:60.0min:
Epoch:[0/1](16300/41914) loss:2.780 lr:0.0001408 epoch_Time:60.0min:
Epoch:[0/1](16400/41914) loss:2.962 lr:0.0001401 epoch_Time:60.0min:
```

Epoch:[0/1](16500/41914) loss:2.995 lr:0.0001395 epoch_Time:60.0min:
Epoch:[0/1](16600/41914) loss:2.879 lr:0.0001389 epoch_Time:59.0min:
Epoch:[0/1](16700/41914) loss:3.004 lr:0.0001382 epoch_Time:59.0min:
Epoch:[0/1](16800/41914) loss:2.939 lr:0.0001376 epoch_Time:59.0min:
Epoch:[0/1](16900/41914) loss:3.000 lr:0.0001369 epoch_Time:59.0min:
Epoch:[0/1](17000/41914) loss:2.860 lr:0.0001363 epoch_Time:58.0min:
Epoch:[0/1](17100/41914) loss:3.005 lr:0.0001357 epoch_Time:58.0min:
Epoch:[0/1](17200/41914) loss:2.962 lr:0.0001350 epoch_Time:58.0min:
Epoch:[0/1](17300/41914) loss:3.002 lr:0.0001344 epoch_Time:58.0min:
Epoch:[0/1](17400/41914) loss:2.833 lr:0.0001337 epoch_Time:57.0min:
Epoch:[0/1](17500/41914) loss:2.925 lr:0.0001331 epoch_Time:57.0min:
Epoch:[0/1](17600/41914) loss:2.877 lr:0.0001324 epoch_Time:57.0min:
Epoch:[0/1](17700/41914) loss:2.837 lr:0.0001318 epoch_Time:57.0min:
Epoch:[0/1](17800/41914) loss:2.947 lr:0.0001311 epoch_Time:56.0min:
Epoch:[0/1](17900/41914) loss:2.987 lr:0.0001304 epoch_Time:56.0min:
Epoch:[0/1](18000/41914) loss:2.940 lr:0.0001298 epoch_Time:56.0min:
Epoch:[0/1](18100/41914) loss:2.884 lr:0.0001291 epoch_Time:56.0min:
Epoch:[0/1](18200/41914) loss:2.955 lr:0.0001285 epoch_Time:56.0min:
Epoch:[0/1](18300/41914) loss:2.780 lr:0.0001278 epoch_Time:55.0min:
Epoch:[0/1](18400/41914) loss:2.948 lr:0.0001271 epoch_Time:55.0min:
Epoch:[0/1](18500/41914) loss:2.783 lr:0.0001265 epoch_Time:55.0min:
Epoch:[0/1](18600/41914) loss:2.776 lr:0.0001258 epoch_Time:55.0min:
Epoch:[0/1](18700/41914) loss:2.859 lr:0.0001252 epoch_Time:54.0min:
Epoch:[0/1](18800/41914) loss:2.935 lr:0.0001245 epoch_Time:54.0min:
Epoch:[0/1](18900/41914) loss:2.721 lr:0.0001238 epoch_Time:54.0min:
Epoch:[0/1](19000/41914) loss:3.000 lr:0.0001232 epoch_Time:54.0min:
Epoch:[0/1](19100/41914) loss:2.879 lr:0.0001225 epoch_Time:53.0min:
Epoch:[0/1](19200/41914) loss:2.729 lr:0.0001218 epoch_Time:53.0min:
Epoch:[0/1](19300/41914) loss:2.899 lr:0.0001211 epoch_Time:53.0min:
Epoch:[0/1](19400/41914) loss:2.840 lr:0.0001205 epoch_Time:53.0min:
Epoch:[0/1](19500/41914) loss:2.894 lr:0.0001198 epoch_Time:52.0min:
Epoch:[0/1](19600/41914) loss:3.015 lr:0.0001191 epoch_Time:52.0min:
Epoch:[0/1](19700/41914) loss:2.812 lr:0.0001185 epoch_Time:52.0min:
Epoch:[0/1](19800/41914) loss:2.939 lr:0.0001178 epoch_Time:52.0min:
Epoch:[0/1](19900/41914) loss:2.749 lr:0.0001171 epoch_Time:52.0min:
Epoch:[0/1](20000/41914) loss:2.991 lr:0.0001165 epoch_Time:51.0min:
Epoch:[0/1](20100/41914) loss:2.955 lr:0.0001158 epoch_Time:51.0min:
Epoch:[0/1](20200/41914) loss:2.820 lr:0.0001151 epoch_Time:51.0min:
Epoch:[0/1](20300/41914) loss:2.898 lr:0.0001144 epoch_Time:51.0min:
Epoch:[0/1](20400/41914) loss:2.977 lr:0.0001138 epoch_Time:50.0min:
Epoch:[0/1](20500/41914) loss:2.792 lr:0.0001131 epoch_Time:50.0min:
Epoch:[0/1](20600/41914) loss:2.843 lr:0.0001124 epoch_Time:50.0min:
Epoch:[0/1](20700/41914) loss:2.900 lr:0.0001117 epoch_Time:50.0min:
Epoch:[0/1](20800/41914) loss:2.931 lr:0.0001111 epoch_Time:49.0min:
Epoch:[0/1](20900/41914) loss:2.772 lr:0.0001104 epoch_Time:49.0min:
Epoch:[0/1](21000/41914) loss:2.998 lr:0.0001097 epoch_Time:49.0min:
Epoch:[0/1](21100/41914) loss:2.732 lr:0.0001090 epoch_Time:49.0min:
Epoch:[0/1](21200/41914) loss:2.973 lr:0.0001084 epoch_Time:48.0min:

Epoch:[0/1](21300/41914) loss:2.623 lr:0.0001077 epoch_Time:48.0min:
Epoch:[0/1](21400/41914) loss:2.877 lr:0.0001070 epoch_Time:48.0min:
Epoch:[0/1](21500/41914) loss:2.825 lr:0.0001063 epoch_Time:48.0min:
Epoch:[0/1](21600/41914) loss:2.831 lr:0.0001057 epoch_Time:48.0min:
Epoch:[0/1](21700/41914) loss:2.898 lr:0.0001050 epoch_Time:47.0min:
Epoch:[0/1](21800/41914) loss:2.859 lr:0.0001043 epoch_Time:47.0min:
Epoch:[0/1](21900/41914) loss:2.706 lr:0.0001036 epoch_Time:47.0min:
Epoch:[0/1](22000/41914) loss:2.856 lr:0.0001030 epoch_Time:47.0min:
Epoch:[0/1](22100/41914) loss:2.877 lr:0.0001023 epoch_Time:46.0min:
Epoch:[0/1](22200/41914) loss:2.830 lr:0.0001016 epoch_Time:46.0min:
Epoch:[0/1](22300/41914) loss:2.603 lr:0.0001010 epoch_Time:46.0min:
Epoch:[0/1](22400/41914) loss:2.913 lr:0.0001003 epoch_Time:46.0min:
Epoch:[0/1](22500/41914) loss:2.907 lr:0.0000996 epoch_Time:45.0min:
Epoch:[0/1](22600/41914) loss:2.630 lr:0.0000989 epoch_Time:45.0min:
Epoch:[0/1](22700/41914) loss:2.850 lr:0.0000983 epoch_Time:45.0min:
Epoch:[0/1](22800/41914) loss:2.752 lr:0.0000976 epoch_Time:45.0min:
Epoch:[0/1](22900/41914) loss:2.760 lr:0.0000969 epoch_Time:44.0min:
Epoch:[0/1](23000/41914) loss:2.899 lr:0.0000963 epoch_Time:44.0min:
Epoch:[0/1](23100/41914) loss:2.663 lr:0.0000956 epoch_Time:44.0min:
Epoch:[0/1](23200/41914) loss:2.778 lr:0.0000949 epoch_Time:44.0min:
Epoch:[0/1](23300/41914) loss:2.700 lr:0.0000943 epoch_Time:44.0min:
Epoch:[0/1](23400/41914) loss:2.912 lr:0.0000936 epoch_Time:43.0min:
Epoch:[0/1](23500/41914) loss:2.833 lr:0.0000929 epoch_Time:43.0min:
Epoch:[0/1](23600/41914) loss:2.759 lr:0.0000923 epoch_Time:43.0min:
Epoch:[0/1](23700/41914) loss:2.916 lr:0.0000916 epoch_Time:43.0min:
Epoch:[0/1](23800/41914) loss:2.843 lr:0.0000910 epoch_Time:42.0min:
Epoch:[0/1](23900/41914) loss:2.822 lr:0.0000903 epoch_Time:42.0min:
Epoch:[0/1](24000/41914) loss:2.927 lr:0.0000897 epoch_Time:42.0min:
Epoch:[0/1](24100/41914) loss:2.901 lr:0.0000890 epoch_Time:42.0min:
Epoch:[0/1](24200/41914) loss:2.960 lr:0.0000883 epoch_Time:41.0min:
Epoch:[0/1](24300/41914) loss:2.713 lr:0.0000877 epoch_Time:41.0min:
Epoch:[0/1](24400/41914) loss:2.758 lr:0.0000870 epoch_Time:41.0min:
Epoch:[0/1](24500/41914) loss:3.026 lr:0.0000864 epoch_Time:41.0min:
Epoch:[0/1](24600/41914) loss:2.905 lr:0.0000857 epoch_Time:40.0min:
Epoch:[0/1](24700/41914) loss:2.811 lr:0.0000851 epoch_Time:40.0min:
Epoch:[0/1](24800/41914) loss:2.697 lr:0.0000844 epoch_Time:40.0min:
Epoch:[0/1](24900/41914) loss:2.843 lr:0.0000838 epoch_Time:40.0min:
Epoch:[0/1](25000/41914) loss:2.764 lr:0.0000831 epoch_Time:39.0min:
Epoch:[0/1](25100/41914) loss:2.852 lr:0.0000825 epoch_Time:39.0min:
Epoch:[0/1](25200/41914) loss:2.903 lr:0.0000819 epoch_Time:39.0min:
Epoch:[0/1](25300/41914) loss:2.833 lr:0.0000812 epoch_Time:39.0min:
Epoch:[0/1](25400/41914) loss:2.842 lr:0.0000806 epoch_Time:39.0min:
Epoch:[0/1](25500/41914) loss:2.779 lr:0.0000799 epoch_Time:38.0min:
Epoch:[0/1](25600/41914) loss:2.775 lr:0.0000793 epoch_Time:38.0min:
Epoch:[0/1](25700/41914) loss:2.712 lr:0.0000787 epoch_Time:38.0min:
Epoch:[0/1](25800/41914) loss:2.807 lr:0.0000780 epoch_Time:38.0min:
Epoch:[0/1](25900/41914) loss:2.745 lr:0.0000774 epoch_Time:37.0min:
Epoch:[0/1](26000/41914) loss:2.715 lr:0.0000768 epoch_Time:37.0min:

Epoch:[0/1](26100/41914) loss:2.675 lr:0.0000762 epoch_Time:37.0min:
Epoch:[0/1](26200/41914) loss:2.721 lr:0.0000755 epoch_Time:37.0min:
Epoch:[0/1](26300/41914) loss:2.841 lr:0.0000749 epoch_Time:36.0min:
Epoch:[0/1](26400/41914) loss:2.794 lr:0.0000743 epoch_Time:36.0min:
Epoch:[0/1](26500/41914) loss:2.796 lr:0.0000737 epoch_Time:36.0min:
Epoch:[0/1](26600/41914) loss:2.782 lr:0.0000731 epoch_Time:36.0min:
Epoch:[0/1](26700/41914) loss:2.856 lr:0.0000724 epoch_Time:35.0min:
Epoch:[0/1](26800/41914) loss:2.795 lr:0.0000718 epoch_Time:35.0min:
Epoch:[0/1](26900/41914) loss:2.701 lr:0.0000712 epoch_Time:35.0min:
Epoch:[0/1](27000/41914) loss:2.751 lr:0.0000706 epoch_Time:35.0min:
Epoch:[0/1](27100/41914) loss:2.745 lr:0.0000700 epoch_Time:35.0min:
Epoch:[0/1](27200/41914) loss:2.835 lr:0.0000694 epoch_Time:34.0min:
Epoch:[0/1](27300/41914) loss:2.870 lr:0.0000688 epoch_Time:34.0min:
Epoch:[0/1](27400/41914) loss:2.798 lr:0.0000682 epoch_Time:34.0min:
Epoch:[0/1](27500/41914) loss:2.809 lr:0.0000676 epoch_Time:34.0min:
Epoch:[0/1](27600/41914) loss:2.836 lr:0.0000670 epoch_Time:33.0min:
Epoch:[0/1](27700/41914) loss:2.682 lr:0.0000664 epoch_Time:33.0min:
Epoch:[0/1](27800/41914) loss:2.685 lr:0.0000658 epoch_Time:33.0min:
Epoch:[0/1](27900/41914) loss:2.875 lr:0.0000652 epoch_Time:33.0min:
Epoch:[0/1](28000/41914) loss:2.741 lr:0.0000647 epoch_Time:32.0min:
Epoch:[0/1](28100/41914) loss:2.694 lr:0.0000641 epoch_Time:32.0min:
Epoch:[0/1](28200/41914) loss:2.809 lr:0.0000635 epoch_Time:32.0min:
Epoch:[0/1](28300/41914) loss:2.834 lr:0.0000629 epoch_Time:32.0min:
Epoch:[0/1](28400/41914) loss:2.816 lr:0.0000624 epoch_Time:31.0min:
Epoch:[0/1](28500/41914) loss:2.724 lr:0.0000618 epoch_Time:31.0min:
Epoch:[0/1](28600/41914) loss:2.784 lr:0.0000612 epoch_Time:31.0min:
Epoch:[0/1](28700/41914) loss:2.815 lr:0.0000607 epoch_Time:31.0min:
Epoch:[0/1](28800/41914) loss:2.831 lr:0.0000601 epoch_Time:31.0min:
Epoch:[0/1](28900/41914) loss:2.730 lr:0.0000595 epoch_Time:30.0min:
Epoch:[0/1](29000/41914) loss:2.848 lr:0.0000590 epoch_Time:30.0min:
Epoch:[0/1](29100/41914) loss:2.811 lr:0.0000584 epoch_Time:30.0min:
Epoch:[0/1](29200/41914) loss:2.721 lr:0.0000579 epoch_Time:30.0min:
Epoch:[0/1](29300/41914) loss:2.739 lr:0.0000573 epoch_Time:29.0min:
Epoch:[0/1](29400/41914) loss:2.843 lr:0.0000568 epoch_Time:29.0min:
Epoch:[0/1](29500/41914) loss:2.752 lr:0.0000562 epoch_Time:29.0min:
Epoch:[0/1](29600/41914) loss:2.566 lr:0.0000557 epoch_Time:29.0min:
Epoch:[0/1](29700/41914) loss:2.792 lr:0.0000552 epoch_Time:28.0min:
Epoch:[0/1](29800/41914) loss:2.610 lr:0.0000546 epoch_Time:28.0min:
Epoch:[0/1](29900/41914) loss:2.904 lr:0.0000541 epoch_Time:28.0min:
Epoch:[0/1](30000/41914) loss:2.732 lr:0.0000536 epoch_Time:28.0min:
Epoch:[0/1](30100/41914) loss:2.876 lr:0.0000530 epoch_Time:27.0min:
Epoch:[0/1](30200/41914) loss:2.746 lr:0.0000525 epoch_Time:27.0min:
Epoch:[0/1](30300/41914) loss:2.839 lr:0.0000520 epoch_Time:27.0min:
Epoch:[0/1](30400/41914) loss:2.727 lr:0.0000515 epoch_Time:27.0min:
Epoch:[0/1](30500/41914) loss:2.718 lr:0.0000510 epoch_Time:27.0min:
Epoch:[0/1](30600/41914) loss:2.766 lr:0.0000505 epoch_Time:26.0min:
Epoch:[0/1](30700/41914) loss:2.749 lr:0.0000500 epoch_Time:26.0min:
Epoch:[0/1](30800/41914) loss:2.848 lr:0.0000495 epoch_Time:26.0min:

```
Epoch:[0/1](30900/41914) loss:2.676 lr:0.0000490 epoch_Time:26.0min:
Epoch:[0/1](31000/41914) loss:2.795 lr:0.0000485 epoch_Time:25.0min:
Epoch:[0/1](31100/41914) loss:2.708 lr:0.0000480 epoch_Time:25.0min:
Epoch:[0/1](31200/41914) loss:2.697 lr:0.0000475 epoch_Time:25.0min:
Epoch:[0/1](31300/41914) loss:2.870 lr:0.0000470 epoch_Time:25.0min:
Epoch:[0/1](31400/41914) loss:2.603 lr:0.0000465 epoch_Time:24.0min:
Epoch:[0/1](31500/41914) loss:2.625 lr:0.0000461 epoch_Time:24.0min:
Epoch:[0/1](31600/41914) loss:2.788 lr:0.0000456 epoch_Time:24.0min:
Epoch:[0/1](31700/41914) loss:2.761 lr:0.0000451 epoch_Time:24.0min:
Epoch:[0/1](31800/41914) loss:2.863 lr:0.0000446 epoch_Time:23.0min:
Epoch:[0/1](31900/41914) loss:2.772 lr:0.0000442 epoch_Time:23.0min:
Epoch:[0/1](32000/41914) loss:2.905 lr:0.0000437 epoch_Time:23.0min:
Epoch:[0/1](32100/41914) loss:2.662 lr:0.0000433 epoch_Time:23.0min:
Epoch:[0/1](32200/41914) loss:2.637 lr:0.0000428 epoch_Time:23.0min:
Epoch:[0/1](32300/41914) loss:2.664 lr:0.0000424 epoch_Time:22.0min:
Epoch:[0/1](32400/41914) loss:2.822 lr:0.0000419 epoch_Time:22.0min:
Epoch:[0/1](32500/41914) loss:2.756 lr:0.0000415 epoch_Time:22.0min:
Epoch:[0/1](32600/41914) loss:2.867 lr:0.0000411 epoch_Time:22.0min:
Epoch:[0/1](32700/41914) loss:2.614 lr:0.0000406 epoch_Time:21.0min:
Epoch:[0/1](32800/41914) loss:2.759 lr:0.0000402 epoch_Time:21.0min:
Epoch:[0/1](32900/41914) loss:2.775 lr:0.0000398 epoch_Time:21.0min:
Epoch:[0/1](33000/41914) loss:2.737 lr:0.0000394 epoch_Time:21.0min:
Epoch:[0/1](33100/41914) loss:2.757 lr:0.0000389 epoch_Time:20.0min:
Epoch:[0/1](33200/41914) loss:2.728 lr:0.0000385 epoch_Time:20.0min:
Epoch:[0/1](33300/41914) loss:2.710 lr:0.0000381 epoch_Time:20.0min:
Epoch:[0/1](33400/41914) loss:2.775 lr:0.0000377 epoch_Time:20.0min:
Epoch:[0/1](33500/41914) loss:2.742 lr:0.0000373 epoch_Time:19.0min:
Epoch:[0/1](33600/41914) loss:2.743 lr:0.0000369 epoch_Time:19.0min:
Epoch:[0/1](33700/41914) loss:2.963 lr:0.0000365 epoch_Time:19.0min:
Epoch:[0/1](33800/41914) loss:2.886 lr:0.0000361 epoch_Time:19.0min:
Epoch:[0/1](33900/41914) loss:2.564 lr:0.0000358 epoch_Time:19.0min:
Epoch:[0/1](34000/41914) loss:2.741 lr:0.0000354 epoch_Time:18.0min:
Epoch:[0/1](34100/41914) loss:2.790 lr:0.0000350 epoch_Time:18.0min:
Epoch:[0/1](34200/41914) loss:2.774 lr:0.0000346 epoch_Time:18.0min:
Epoch:[0/1](34300/41914) loss:2.789 lr:0.0000343 epoch_Time:18.0min:
Epoch:[0/1](34400/41914) loss:2.751 lr:0.0000339 epoch_Time:17.0min:
Epoch:[0/1](34500/41914) loss:2.661 lr:0.0000335 epoch_Time:17.0min:
Epoch:[0/1](34600/41914) loss:2.883 lr:0.0000332 epoch_Time:17.0min:
Epoch:[0/1](34700/41914) loss:2.594 lr:0.0000328 epoch_Time:17.0min:
Epoch:[0/1](34800/41914) loss:2.896 lr:0.0000325 epoch_Time:16.0min:
Epoch:[0/1](34900/41914) loss:2.742 lr:0.0000322 epoch_Time:16.0min:
Epoch:[0/1](35000/41914) loss:2.799 lr:0.0000318 epoch_Time:16.0min:
Epoch:[0/1](35100/41914) loss:2.899 lr:0.0000315 epoch_Time:16.0min:
Epoch:[0/1](35200/41914) loss:2.809 lr:0.0000312 epoch_Time:15.0min:
Epoch:[0/1](35300/41914) loss:2.625 lr:0.0000308 epoch_Time:15.0min:
Epoch:[0/1](35400/41914) loss:2.742 lr:0.0000305 epoch_Time:15.0min:
Epoch:[0/1](35500/41914) loss:2.857 lr:0.0000302 epoch_Time:15.0min:
Epoch:[0/1](35600/41914) loss:2.896 lr:0.0000299 epoch_Time:14.0min:
```

```
Epoch:[0/1](35700/41914) loss:2.697 lr:0.0000296 epoch_Time:14.0min:
Epoch:[0/1](35800/41914) loss:2.659 lr:0.0000293 epoch_Time:14.0min:
Epoch:[0/1](35900/41914) loss:2.803 lr:0.0000290 epoch_Time:14.0min:
Epoch:[0/1](36000/41914) loss:2.822 lr:0.0000287 epoch_Time:14.0min:
Epoch:[0/1](36100/41914) loss:2.669 lr:0.0000284 epoch_Time:13.0min:
Epoch:[0/1](36200/41914) loss:2.810 lr:0.0000281 epoch_Time:13.0min:
Epoch:[0/1](36300/41914) loss:2.618 lr:0.0000279 epoch_Time:13.0min:
Epoch:[0/1](36400/41914) loss:2.886 lr:0.0000276 epoch_Time:13.0min:
Epoch:[0/1](36500/41914) loss:2.569 lr:0.0000273 epoch_Time:12.0min:
Epoch:[0/1](36600/41914) loss:2.705 lr:0.0000270 epoch_Time:12.0min:
Epoch:[0/1](36700/41914) loss:2.830 lr:0.0000268 epoch_Time:12.0min:
Epoch:[0/1](36800/41914) loss:2.754 lr:0.0000265 epoch_Time:12.0min:
Epoch:[0/1](36900/41914) loss:2.714 lr:0.0000263 epoch_Time:11.0min:
Epoch:[0/1](37000/41914) loss:2.816 lr:0.0000260 epoch_Time:11.0min:
Epoch:[0/1](37100/41914) loss:2.697 lr:0.0000258 epoch_Time:11.0min:
Epoch:[0/1](37200/41914) loss:2.714 lr:0.0000256 epoch_Time:11.0min:
Epoch:[0/1](37300/41914) loss:2.810 lr:0.0000253 epoch_Time:10.0min:
Epoch:[0/1](37400/41914) loss:2.677 lr:0.0000251 epoch_Time:10.0min:
Epoch:[0/1](37500/41914) loss:2.662 lr:0.0000249 epoch_Time:10.0min:
Epoch:[0/1](37600/41914) loss:2.687 lr:0.0000247 epoch_Time:10.0min:
Epoch:[0/1](37700/41914) loss:2.661 lr:0.0000245 epoch_Time:10.0min:
Epoch:[0/1](37800/41914) loss:2.843 lr:0.0000242 epoch_Time:9.0min:
Epoch:[0/1](37900/41914) loss:2.788 lr:0.0000240 epoch_Time:9.0min:
Epoch:[0/1](38000/41914) loss:2.639 lr:0.0000238 epoch_Time:9.0min:
Epoch:[0/1](38100/41914) loss:2.808 lr:0.0000237 epoch_Time:9.0min:
Epoch:[0/1](38200/41914) loss:2.667 lr:0.0000235 epoch_Time:8.0min:
Epoch:[0/1](38300/41914) loss:2.863 lr:0.0000233 epoch_Time:8.0min:
Epoch:[0/1](38400/41914) loss:2.692 lr:0.0000231 epoch_Time:8.0min:
Epoch:[0/1](38500/41914) loss:2.778 lr:0.0000229 epoch_Time:8.0min:
Epoch:[0/1](38600/41914) loss:2.798 lr:0.0000228 epoch_Time:7.0min:
Epoch:[0/1](38700/41914) loss:2.783 lr:0.0000226 epoch_Time:7.0min:
Epoch:[0/1](38800/41914) loss:2.779 lr:0.0000224 epoch_Time:7.0min:
Epoch:[0/1](38900/41914) loss:2.644 lr:0.0000223 epoch_Time:7.0min:
Epoch:[0/1](39000/41914) loss:2.929 lr:0.0000221 epoch_Time:6.0min:
Epoch:[0/1](39100/41914) loss:2.724 lr:0.0000220 epoch_Time:6.0min:
Epoch:[0/1](39200/41914) loss:2.901 lr:0.0000219 epoch_Time:6.0min:
Epoch:[0/1](39300/41914) loss:2.702 lr:0.0000217 epoch_Time:6.0min:
Epoch:[0/1](39400/41914) loss:2.768 lr:0.0000216 epoch_Time:6.0min:
Epoch:[0/1](39500/41914) loss:2.600 lr:0.0000215 epoch_Time:5.0min:
Epoch:[0/1](39600/41914) loss:2.878 lr:0.0000214 epoch_Time:5.0min:
Epoch:[0/1](39700/41914) loss:2.735 lr:0.0000212 epoch_Time:5.0min:
Epoch:[0/1](39800/41914) loss:2.708 lr:0.0000211 epoch_Time:5.0min:
Epoch:[0/1](39900/41914) loss:2.866 lr:0.0000210 epoch_Time:4.0min:
Epoch:[0/1](40000/41914) loss:2.751 lr:0.0000209 epoch_Time:4.0min:
Epoch:[0/1](40100/41914) loss:2.598 lr:0.0000208 epoch_Time:4.0min:
Epoch:[0/1](40200/41914) loss:2.561 lr:0.0000207 epoch_Time:4.0min:
Epoch:[0/1](40300/41914) loss:2.749 lr:0.0000207 epoch_Time:3.0min:
Epoch:[0/1](40400/41914) loss:2.608 lr:0.0000206 epoch_Time:3.0min:
```

```
Epoch:[0/1](40500/41914) loss:2.658 lr:0.0000205 epoch_Time:3.0min:
Epoch:[0/1](40600/41914) loss:2.649 lr:0.0000204 epoch_Time:3.0min:
Epoch:[0/1](40700/41914) loss:2.697 lr:0.0000204 epoch_Time:2.0min:
Epoch:[0/1](40800/41914) loss:2.705 lr:0.0000203 epoch_Time:2.0min:
Epoch:[0/1](40900/41914) loss:2.752 lr:0.0000203 epoch_Time:2.0min:
Epoch:[0/1](41000/41914) loss:2.726 lr:0.0000202 epoch_Time:2.0min:
Epoch:[0/1](41100/41914) loss:2.764 lr:0.0000202 epoch_Time:1.0min:
Epoch:[0/1](41200/41914) loss:2.708 lr:0.0000201 epoch_Time:1.0min:
Epoch:[0/1](41300/41914) loss:2.780 lr:0.0000201 epoch_Time:1.0min:
Epoch:[0/1](41400/41914) loss:2.696 lr:0.0000201 epoch_Time:1.0min:
Epoch:[0/1](41500/41914) loss:2.737 lr:0.0000200 epoch_Time:1.0min:
Epoch:[0/1](41600/41914) loss:2.900 lr:0.0000200 epoch_Time:0.0min:
Epoch:[0/1](41700/41914) loss:2.673 lr:0.0000200 epoch_Time:0.0min:
Epoch:[0/1](41800/41914) loss:2.700 lr:0.0000200 epoch_Time:0.0min:
Epoch:[0/1](41900/41914) loss:2.749 lr:0.0000200 epoch_Time:0.0min:
Epoch 1 average loss: 3.063
Figure(1200x600)
```

4.4.2 3.4.2 Full Supervised Fine-tuning (SFT) Phase (using Deepctrl-sft-data)

Objective: To fine-tune the pretrained model on the specific task of general natural language understanding, leveraging labeled data to learn task-specific patterns and improve performance.

Dataset: Deepctrl-sft-data, a large-scale SFT dataset consisting of 10 million Chinese data entries and 2 million English data entries, totaling approximately 3 billion tokens. This dataset provides a diverse and comprehensive set of examples for fine-tuning language models.

Link: https://www.modelscope.cn/datasets/deepctrl/deepctrl-sft-data

Training Procedure:

1. **Data Preparation:**
   - The Deepctrl-sft-data is preprocessed into (input, output) pairs.
   - Each (input, output) pair is tokenized using the **HOLMES-26M** tokenizer.
   - Input sequences are padded or truncated to a fixed length (`max_seq_len = 2048` tokens).
2. **Model Input:** The model takes the tokenized input sequence `X` and the corresponding target sequence `Y` as input. A loss mask is also provided to indicate which tokens should be considered during loss calculation.
3. **Loss Function:**
   - The model is trained to minimize the cross-entropy loss between the predicted logits and the target labels, weighted by the loss mask.
   - The loss is calculated as follows:

$$\text{Loss} = \frac{\sum(\text{loss} \cdot \text{loss\_mask})}{\sum \text{loss\_mask}}$$

     where:
     - `logits` are the model's output logits.
     - `Y` are the target labels.

- `loss_mask` indicates which tokens should be considered in the loss calculation.

4. **Optimization:**
   - The model is fine-tuned using the **Adam optimizer** with a learning rate of `1e-4`.
   - **Gradient accumulation** is used with `1` accumulation steps.
   - **Gradient clipping** is applied with a threshold of `1.0`.
   - The learning rate follows a cosine annealing schedule with an optional warmup period.
   - **Mixed-precision training** (`bfloat16`) is used.
5. **Evaluation:** Similar to the pretraining phase, the model's performance is evaluated by monitoring the loss on the training set during fine-tuning.
6. **Implementation Details:**
   - The SFT process is implemented in `3-full_sft.py`.
   - The model is fine-tuned for 1 epochs.
7. **Implementation:**

the whole training process can be found at https://wandb.ai/mjuicem3-beijing-university-of-posts-and-telecommunications/HOLMES-Full-SFT?nw=nwusermjuicem3

[ ]:
```
!torchrun --nproc_per_node 2 3-full_sft.py
```

```
Total LLM parameters: 26.878M
Epoch:[0/1](0/61701) loss:3.102 lr:0.0001000 epoch_Time:1387.0min:
Epoch:[0/1](100/61701) loss:2.705 lr:0.0001000 epoch_Time:82.0min:
Epoch:[0/1](200/61701) loss:2.389 lr:0.0001000 epoch_Time:76.0min:
Epoch:[0/1](300/61701) loss:2.470 lr:0.0001000 epoch_Time:73.0min:
Epoch:[0/1](400/61701) loss:2.290 lr:0.0001000 epoch_Time:72.0min:
Epoch:[0/1](500/61701) loss:2.299 lr:0.0001000 epoch_Time:72.0min:
Epoch:[0/1](600/61701) loss:2.299 lr:0.0001000 epoch_Time:71.0min:
Epoch:[0/1](700/61701) loss:2.402 lr:0.0001000 epoch_Time:71.0min:
Epoch:[0/1](800/61701) loss:2.551 lr:0.0001000 epoch_Time:71.0min:
Epoch:[0/1](900/61701) loss:2.327 lr:0.0001000 epoch_Time:70.0min:
Epoch:[0/1](1000/61701) loss:2.648 lr:0.0000999 epoch_Time:69.0min:
Epoch:[0/1](1100/61701) loss:2.231 lr:0.0000999 epoch_Time:69.0min:
Epoch:[0/1](1200/61701) loss:2.284 lr:0.0000999 epoch_Time:69.0min:
Epoch:[0/1](1300/61701) loss:2.179 lr:0.0000999 epoch_Time:69.0min:
Epoch:[0/1](1400/61701) loss:2.473 lr:0.0000999 epoch_Time:69.0min:
Epoch:[0/1](1500/61701) loss:2.160 lr:0.0000999 epoch_Time:69.0min:
Epoch:[0/1](1600/61701) loss:2.564 lr:0.0000999 epoch_Time:69.0min:
Epoch:[0/1](1700/61701) loss:2.275 lr:0.0000998 epoch_Time:69.0min:
Epoch:[0/1](1800/61701) loss:2.493 lr:0.0000998 epoch_Time:68.0min:
Epoch:[0/1](1900/61701) loss:2.592 lr:0.0000998 epoch_Time:68.0min:
Epoch:[0/1](2000/61701) loss:2.186 lr:0.0000998 epoch_Time:68.0min:
Epoch:[0/1](2100/61701) loss:2.426 lr:0.0000997 epoch_Time:68.0min:
Epoch:[0/1](2200/61701) loss:2.178 lr:0.0000997 epoch_Time:68.0min:
Epoch:[0/1](2300/61701) loss:2.375 lr:0.0000997 epoch_Time:68.0min:
Epoch:[0/1](2400/61701) loss:2.428 lr:0.0000997 epoch_Time:68.0min:
```

```
Epoch:[0/1](2500/61701) loss:2.457 lr:0.0000996 epoch_Time:68.0min:
Epoch:[0/1](2600/61701) loss:2.292 lr:0.0000996 epoch_Time:68.0min:
Epoch:[0/1](2700/61701) loss:2.301 lr:0.0000996 epoch_Time:67.0min:
Epoch:[0/1](2800/61701) loss:2.183 lr:0.0000995 epoch_Time:67.0min:
Epoch:[0/1](2900/61701) loss:2.237 lr:0.0000995 epoch_Time:67.0min:
Epoch:[0/1](3000/61701) loss:2.235 lr:0.0000995 epoch_Time:67.0min:
Epoch:[0/1](3100/61701) loss:2.470 lr:0.0000994 epoch_Time:67.0min:
Epoch:[0/1](3200/61701) loss:2.337 lr:0.0000994 epoch_Time:67.0min:
Epoch:[0/1](3300/61701) loss:2.308 lr:0.0000994 epoch_Time:67.0min:
Epoch:[0/1](3400/61701) loss:2.020 lr:0.0000993 epoch_Time:67.0min:
Epoch:[0/1](3500/61701) loss:2.268 lr:0.0000993 epoch_Time:67.0min:
Epoch:[0/1](3600/61701) loss:2.461 lr:0.0000992 epoch_Time:66.0min:
Epoch:[0/1](3700/61701) loss:1.983 lr:0.0000992 epoch_Time:66.0min:
Epoch:[0/1](3800/61701) loss:2.360 lr:0.0000992 epoch_Time:66.0min:
Epoch:[0/1](3900/61701) loss:2.207 lr:0.0000991 epoch_Time:66.0min:
Epoch:[0/1](4000/61701) loss:2.291 lr:0.0000991 epoch_Time:66.0min:
Epoch:[0/1](4100/61701) loss:2.284 lr:0.0000990 epoch_Time:66.0min:
Epoch:[0/1](4200/61701) loss:1.919 lr:0.0000990 epoch_Time:66.0min:
Epoch:[0/1](4300/61701) loss:2.081 lr:0.0000989 epoch_Time:65.0min:
Epoch:[0/1](4400/61701) loss:2.415 lr:0.0000989 epoch_Time:65.0min:
Epoch:[0/1](4500/61701) loss:2.219 lr:0.0000988 epoch_Time:64.0min:
Epoch:[0/1](4600/61701) loss:2.231 lr:0.0000988 epoch_Time:64.0min:
Epoch:[0/1](4700/61701) loss:2.356 lr:0.0000987 epoch_Time:64.0min:
Epoch:[0/1](4800/61701) loss:2.040 lr:0.0000987 epoch_Time:64.0min:
Epoch:[0/1](4900/61701) loss:2.160 lr:0.0000986 epoch_Time:64.0min:
Epoch:[0/1](5000/61701) loss:2.218 lr:0.0000985 epoch_Time:64.0min:
Epoch:[0/1](5100/61701) loss:2.145 lr:0.0000985 epoch_Time:64.0min:
Epoch:[0/1](5200/61701) loss:2.165 lr:0.0000984 epoch_Time:64.0min:
Epoch:[0/1](5300/61701) loss:2.267 lr:0.0000984 epoch_Time:63.0min:
Epoch:[0/1](5400/61701) loss:2.276 lr:0.0000983 epoch_Time:63.0min:
Epoch:[0/1](5500/61701) loss:2.569 lr:0.0000982 epoch_Time:63.0min:
Epoch:[0/1](5600/61701) loss:2.357 lr:0.0000982 epoch_Time:63.0min:
Epoch:[0/1](5700/61701) loss:2.530 lr:0.0000981 epoch_Time:63.0min:
Epoch:[0/1](5800/61701) loss:2.212 lr:0.0000981 epoch_Time:63.0min:
Epoch:[0/1](5900/61701) loss:2.352 lr:0.0000980 epoch_Time:63.0min:
Epoch:[0/1](6000/61701) loss:2.260 lr:0.0000979 epoch_Time:63.0min:
Epoch:[0/1](6100/61701) loss:2.215 lr:0.0000978 epoch_Time:63.0min:
Epoch:[0/1](6200/61701) loss:2.523 lr:0.0000978 epoch_Time:62.0min:
Epoch:[0/1](6300/61701) loss:2.188 lr:0.0000977 epoch_Time:62.0min:
Epoch:[0/1](6400/61701) loss:1.998 lr:0.0000976 epoch_Time:62.0min:
Epoch:[0/1](6500/61701) loss:2.418 lr:0.0000976 epoch_Time:62.0min:
Epoch:[0/1](6600/61701) loss:2.254 lr:0.0000975 epoch_Time:62.0min:
Epoch:[0/1](6700/61701) loss:2.341 lr:0.0000974 epoch_Time:62.0min:
Epoch:[0/1](6800/61701) loss:2.010 lr:0.0000973 epoch_Time:62.0min:
Epoch:[0/1](6900/61701) loss:2.449 lr:0.0000973 epoch_Time:62.0min:
Epoch:[0/1](7000/61701) loss:2.068 lr:0.0000972 epoch_Time:62.0min:
Epoch:[0/1](7100/61701) loss:2.186 lr:0.0000971 epoch_Time:61.0min:
Epoch:[0/1](7200/61701) loss:2.430 lr:0.0000970 epoch_Time:61.0min:
```

```
Epoch:[0/1](7300/61701) loss:2.383 lr:0.0000969 epoch_Time:61.0min:
Epoch:[0/1](7400/61701) loss:2.474 lr:0.0000968 epoch_Time:61.0min:
Epoch:[0/1](7500/61701) loss:2.167 lr:0.0000968 epoch_Time:61.0min:
Epoch:[0/1](7600/61701) loss:2.224 lr:0.0000967 epoch_Time:61.0min:
Epoch:[0/1](7700/61701) loss:2.521 lr:0.0000966 epoch_Time:61.0min:
Epoch:[0/1](7800/61701) loss:1.993 lr:0.0000965 epoch_Time:61.0min:
Epoch:[0/1](7900/61701) loss:2.261 lr:0.0000964 epoch_Time:61.0min:
Epoch:[0/1](8000/61701) loss:1.818 lr:0.0000963 epoch_Time:60.0min:
Epoch:[0/1](8100/61701) loss:2.052 lr:0.0000962 epoch_Time:60.0min:
Epoch:[0/1](8200/61701) loss:2.305 lr:0.0000961 epoch_Time:60.0min:
Epoch:[0/1](8300/61701) loss:2.353 lr:0.0000960 epoch_Time:60.0min:
Epoch:[0/1](8400/61701) loss:2.287 lr:0.0000959 epoch_Time:60.0min:
Epoch:[0/1](8500/61701) loss:2.041 lr:0.0000959 epoch_Time:60.0min:
Epoch:[0/1](8600/61701) loss:2.101 lr:0.0000958 epoch_Time:60.0min:
Epoch:[0/1](8700/61701) loss:2.205 lr:0.0000957 epoch_Time:60.0min:
Epoch:[0/1](8800/61701) loss:2.151 lr:0.0000956 epoch_Time:60.0min:
Epoch:[0/1](8900/61701) loss:2.211 lr:0.0000955 epoch_Time:59.0min:
Epoch:[0/1](9000/61701) loss:2.102 lr:0.0000954 epoch_Time:59.0min:
Epoch:[0/1](9100/61701) loss:2.002 lr:0.0000953 epoch_Time:59.0min:
Epoch:[0/1](9200/61701) loss:2.090 lr:0.0000952 epoch_Time:59.0min:
Epoch:[0/1](9300/61701) loss:2.132 lr:0.0000950 epoch_Time:59.0min:
Epoch:[0/1](9400/61701) loss:1.882 lr:0.0000949 epoch_Time:59.0min:
Epoch:[0/1](9500/61701) loss:2.170 lr:0.0000948 epoch_Time:59.0min:
Epoch:[0/1](9600/61701) loss:2.153 lr:0.0000947 epoch_Time:59.0min:
Epoch:[0/1](9700/61701) loss:2.183 lr:0.0000946 epoch_Time:59.0min:
Epoch:[0/1](9800/61701) loss:2.084 lr:0.0000945 epoch_Time:58.0min:
Epoch:[0/1](9900/61701) loss:2.133 lr:0.0000944 epoch_Time:58.0min:
Epoch:[0/1](10000/61701) loss:2.070 lr:0.0000943 epoch_Time:58.0min:
Epoch:[0/1](10100/61701) loss:2.200 lr:0.0000942 epoch_Time:58.0min:
Epoch:[0/1](10200/61701) loss:2.218 lr:0.0000941 epoch_Time:58.0min:
Epoch:[0/1](10300/61701) loss:1.940 lr:0.0000940 epoch_Time:58.0min:
Epoch:[0/1](10400/61701) loss:2.084 lr:0.0000938 epoch_Time:58.0min:
Epoch:[0/1](10500/61701) loss:1.823 lr:0.0000937 epoch_Time:58.0min:
Epoch:[0/1](10600/61701) loss:1.924 lr:0.0000936 epoch_Time:57.0min:
Epoch:[0/1](10700/61701) loss:1.919 lr:0.0000935 epoch_Time:57.0min:
Epoch:[0/1](10800/61701) loss:1.952 lr:0.0000934 epoch_Time:57.0min:
Epoch:[0/1](10900/61701) loss:2.150 lr:0.0000932 epoch_Time:57.0min:
Epoch:[0/1](11000/61701) loss:1.815 lr:0.0000931 epoch_Time:57.0min:
Epoch:[0/1](11100/61701) loss:1.884 lr:0.0000930 epoch_Time:57.0min:
Epoch:[0/1](11200/61701) loss:2.030 lr:0.0000929 epoch_Time:57.0min:
Epoch:[0/1](11300/61701) loss:1.887 lr:0.0000928 epoch_Time:57.0min:
Epoch:[0/1](11400/61701) loss:1.909 lr:0.0000926 epoch_Time:57.0min:
Epoch:[0/1](11500/61701) loss:2.157 lr:0.0000925 epoch_Time:56.0min:
Epoch:[0/1](11600/61701) loss:2.074 lr:0.0000924 epoch_Time:56.0min:
Epoch:[0/1](11700/61701) loss:1.905 lr:0.0000922 epoch_Time:56.0min:
Epoch:[0/1](11800/61701) loss:2.005 lr:0.0000921 epoch_Time:56.0min:
Epoch:[0/1](11900/61701) loss:2.299 lr:0.0000920 epoch_Time:56.0min:
Epoch:[0/1](12000/61701) loss:1.882 lr:0.0000919 epoch_Time:56.0min:
```

```
Epoch:[0/1](12100/61701) loss:2.073 lr:0.0000917 epoch_Time:56.0min:
Epoch:[0/1](12200/61701) loss:2.185 lr:0.0000916 epoch_Time:56.0min:
Epoch:[0/1](12300/61701) loss:2.078 lr:0.0000915 epoch_Time:56.0min:
Epoch:[0/1](12400/61701) loss:1.836 lr:0.0000913 epoch_Time:55.0min:
Epoch:[0/1](12500/61701) loss:2.455 lr:0.0000912 epoch_Time:55.0min:
Epoch:[0/1](12600/61701) loss:2.143 lr:0.0000911 epoch_Time:55.0min:
Epoch:[0/1](12700/61701) loss:2.284 lr:0.0000909 epoch_Time:55.0min:
Epoch:[0/1](12800/61701) loss:1.953 lr:0.0000908 epoch_Time:55.0min:
Epoch:[0/1](12900/61701) loss:2.035 lr:0.0000906 epoch_Time:55.0min:
Epoch:[0/1](13000/61701) loss:2.002 lr:0.0000905 epoch_Time:55.0min:
Epoch:[0/1](13100/61701) loss:2.219 lr:0.0000904 epoch_Time:55.0min:
Epoch:[0/1](13200/61701) loss:2.359 lr:0.0000902 epoch_Time:55.0min:
Epoch:[0/1](13300/61701) loss:2.030 lr:0.0000901 epoch_Time:54.0min:
Epoch:[0/1](13400/61701) loss:2.142 lr:0.0000899 epoch_Time:54.0min:
Epoch:[0/1](13500/61701) loss:2.229 lr:0.0000898 epoch_Time:54.0min:
Epoch:[0/1](13600/61701) loss:1.981 lr:0.0000896 epoch_Time:54.0min:
Epoch:[0/1](13700/61701) loss:1.850 lr:0.0000895 epoch_Time:54.0min:
Epoch:[0/1](13800/61701) loss:2.208 lr:0.0000893 epoch_Time:54.0min:
Epoch:[0/1](13900/61701) loss:2.047 lr:0.0000892 epoch_Time:54.0min:
Epoch:[0/1](14000/61701) loss:1.823 lr:0.0000890 epoch_Time:54.0min:
Epoch:[0/1](14100/61701) loss:2.011 lr:0.0000889 epoch_Time:54.0min:
Epoch:[0/1](14200/61701) loss:1.919 lr:0.0000887 epoch_Time:53.0min:
Epoch:[0/1](14300/61701) loss:2.176 lr:0.0000886 epoch_Time:53.0min:
Epoch:[0/1](14400/61701) loss:2.036 lr:0.0000884 epoch_Time:53.0min:
Epoch:[0/1](14500/61701) loss:2.008 lr:0.0000883 epoch_Time:53.0min:
Epoch:[0/1](14600/61701) loss:1.841 lr:0.0000881 epoch_Time:53.0min:
Epoch:[0/1](14700/61701) loss:1.973 lr:0.0000880 epoch_Time:53.0min:
Epoch:[0/1](14800/61701) loss:2.052 lr:0.0000878 epoch_Time:53.0min:
Epoch:[0/1](14900/61701) loss:2.268 lr:0.0000877 epoch_Time:53.0min:
Epoch:[0/1](15000/61701) loss:2.125 lr:0.0000875 epoch_Time:53.0min:
Epoch:[0/1](15100/61701) loss:2.131 lr:0.0000873 epoch_Time:52.0min:
Epoch:[0/1](15200/61701) loss:2.257 lr:0.0000872 epoch_Time:52.0min:
Epoch:[0/1](15300/61701) loss:2.143 lr:0.0000870 epoch_Time:52.0min:
Epoch:[0/1](15400/61701) loss:2.049 lr:0.0000869 epoch_Time:52.0min:
Epoch:[0/1](15500/61701) loss:2.268 lr:0.0000867 epoch_Time:52.0min:
Epoch:[0/1](15600/61701) loss:1.980 lr:0.0000865 epoch_Time:52.0min:
Epoch:[0/1](15700/61701) loss:2.293 lr:0.0000864 epoch_Time:52.0min:
Epoch:[0/1](15800/61701) loss:2.197 lr:0.0000862 epoch_Time:52.0min:
Epoch:[0/1](15900/61701) loss:2.034 lr:0.0000860 epoch_Time:52.0min:
Epoch:[0/1](16000/61701) loss:2.078 lr:0.0000859 epoch_Time:51.0min:
Epoch:[0/1](16100/61701) loss:2.148 lr:0.0000857 epoch_Time:51.0min:
Epoch:[0/1](16200/61701) loss:2.359 lr:0.0000855 epoch_Time:51.0min:
Epoch:[0/1](16300/61701) loss:1.983 lr:0.0000854 epoch_Time:51.0min:
Epoch:[0/1](16400/61701) loss:2.119 lr:0.0000852 epoch_Time:51.0min:
Epoch:[0/1](16500/61701) loss:1.783 lr:0.0000850 epoch_Time:51.0min:
Epoch:[0/1](16600/61701) loss:2.346 lr:0.0000849 epoch_Time:51.0min:
Epoch:[0/1](16700/61701) loss:2.168 lr:0.0000847 epoch_Time:51.0min:
Epoch:[0/1](16800/61701) loss:2.047 lr:0.0000845 epoch_Time:50.0min:
```

Epoch:[0/1](16900/61701) loss:1.947 lr:0.0000843 epoch_Time:50.0min:
Epoch:[0/1](17000/61701) loss:1.762 lr:0.0000842 epoch_Time:50.0min:
Epoch:[0/1](17100/61701) loss:2.100 lr:0.0000840 epoch_Time:50.0min:
Epoch:[0/1](17200/61701) loss:1.911 lr:0.0000838 epoch_Time:50.0min:
Epoch:[0/1](17300/61701) loss:2.115 lr:0.0000836 epoch_Time:50.0min:
Epoch:[0/1](17400/61701) loss:1.820 lr:0.0000835 epoch_Time:50.0min:
Epoch:[0/1](17500/61701) loss:2.120 lr:0.0000833 epoch_Time:50.0min:
Epoch:[0/1](17600/61701) loss:1.982 lr:0.0000831 epoch_Time:50.0min:
Epoch:[0/1](17700/61701) loss:2.067 lr:0.0000829 epoch_Time:49.0min:
Epoch:[0/1](17800/61701) loss:1.887 lr:0.0000827 epoch_Time:49.0min:
Epoch:[0/1](17900/61701) loss:2.110 lr:0.0000826 epoch_Time:49.0min:
Epoch:[0/1](18000/61701) loss:1.975 lr:0.0000824 epoch_Time:49.0min:
Epoch:[0/1](18100/61701) loss:2.251 lr:0.0000822 epoch_Time:49.0min:
Epoch:[0/1](18200/61701) loss:2.150 lr:0.0000820 epoch_Time:49.0min:
Epoch:[0/1](18300/61701) loss:1.997 lr:0.0000818 epoch_Time:49.0min:
Epoch:[0/1](18400/61701) loss:2.029 lr:0.0000817 epoch_Time:49.0min:
Epoch:[0/1](18500/61701) loss:1.935 lr:0.0000815 epoch_Time:49.0min:
Epoch:[0/1](18600/61701) loss:2.149 lr:0.0000813 epoch_Time:48.0min:
Epoch:[0/1](18700/61701) loss:1.978 lr:0.0000811 epoch_Time:48.0min:
Epoch:[0/1](18800/61701) loss:1.875 lr:0.0000809 epoch_Time:48.0min:
Epoch:[0/1](18900/61701) loss:2.070 lr:0.0000807 epoch_Time:48.0min:
Epoch:[0/1](19000/61701) loss:2.203 lr:0.0000805 epoch_Time:48.0min:
Epoch:[0/1](19100/61701) loss:1.897 lr:0.0000803 epoch_Time:48.0min:
Epoch:[0/1](19200/61701) loss:2.367 lr:0.0000802 epoch_Time:48.0min:
Epoch:[0/1](19300/61701) loss:2.208 lr:0.0000800 epoch_Time:48.0min:
Epoch:[0/1](19400/61701) loss:2.085 lr:0.0000798 epoch_Time:48.0min:
Epoch:[0/1](19500/61701) loss:2.137 lr:0.0000796 epoch_Time:47.0min:
Epoch:[0/1](19600/61701) loss:2.192 lr:0.0000794 epoch_Time:47.0min:
Epoch:[0/1](19700/61701) loss:2.323 lr:0.0000792 epoch_Time:47.0min:
Epoch:[0/1](19800/61701) loss:2.043 lr:0.0000790 epoch_Time:47.0min:
Epoch:[0/1](19900/61701) loss:2.169 lr:0.0000788 epoch_Time:47.0min:
Epoch:[0/1](20000/61701) loss:2.273 lr:0.0000786 epoch_Time:47.0min:
Epoch:[0/1](20100/61701) loss:1.941 lr:0.0000784 epoch_Time:47.0min:
Epoch:[0/1](20200/61701) loss:2.102 lr:0.0000782 epoch_Time:47.0min:
Epoch:[0/1](20300/61701) loss:2.006 lr:0.0000780 epoch_Time:47.0min:
Epoch:[0/1](20400/61701) loss:1.859 lr:0.0000778 epoch_Time:46.0min:
Epoch:[0/1](20500/61701) loss:1.776 lr:0.0000776 epoch_Time:46.0min:
Epoch:[0/1](20600/61701) loss:2.021 lr:0.0000774 epoch_Time:46.0min:
Epoch:[0/1](20700/61701) loss:2.056 lr:0.0000772 epoch_Time:46.0min:
Epoch:[0/1](20800/61701) loss:1.989 lr:0.0000770 epoch_Time:46.0min:
Epoch:[0/1](20900/61701) loss:2.127 lr:0.0000768 epoch_Time:46.0min:
Epoch:[0/1](21000/61701) loss:1.860 lr:0.0000766 epoch_Time:46.0min:
Epoch:[0/1](21100/61701) loss:1.984 lr:0.0000764 epoch_Time:46.0min:
Epoch:[0/1](21200/61701) loss:2.020 lr:0.0000762 epoch_Time:46.0min:
Epoch:[0/1](21300/61701) loss:1.896 lr:0.0000760 epoch_Time:45.0min:
Epoch:[0/1](21400/61701) loss:2.060 lr:0.0000758 epoch_Time:45.0min:
Epoch:[0/1](21500/61701) loss:2.033 lr:0.0000756 epoch_Time:45.0min:
Epoch:[0/1](21600/61701) loss:2.023 lr:0.0000754 epoch_Time:45.0min:

```
Epoch:[0/1](21700/61701) loss:2.175 lr:0.0000752 epoch_Time:45.0min:
Epoch:[0/1](21800/61701) loss:2.027 lr:0.0000750 epoch_Time:45.0min:
Epoch:[0/1](21900/61701) loss:2.262 lr:0.0000748 epoch_Time:45.0min:
Epoch:[0/1](22000/61701) loss:2.160 lr:0.0000746 epoch_Time:45.0min:
Epoch:[0/1](22100/61701) loss:1.982 lr:0.0000744 epoch_Time:45.0min:
Epoch:[0/1](22200/61701) loss:2.056 lr:0.0000742 epoch_Time:44.0min:
Epoch:[0/1](22300/61701) loss:2.174 lr:0.0000740 epoch_Time:44.0min:
Epoch:[0/1](22400/61701) loss:2.175 lr:0.0000738 epoch_Time:44.0min:
Epoch:[0/1](22500/61701) loss:1.913 lr:0.0000736 epoch_Time:44.0min:
Epoch:[0/1](22600/61701) loss:2.046 lr:0.0000734 epoch_Time:44.0min:
Epoch:[0/1](22700/61701) loss:1.795 lr:0.0000731 epoch_Time:44.0min:
Epoch:[0/1](22800/61701) loss:2.307 lr:0.0000729 epoch_Time:44.0min:
Epoch:[0/1](22900/61701) loss:1.941 lr:0.0000727 epoch_Time:44.0min:
Epoch:[0/1](23000/61701) loss:1.796 lr:0.0000725 epoch_Time:43.0min:
Epoch:[0/1](23100/61701) loss:2.099 lr:0.0000723 epoch_Time:43.0min:
Epoch:[0/1](23200/61701) loss:1.937 lr:0.0000721 epoch_Time:43.0min:
Epoch:[0/1](23300/61701) loss:2.161 lr:0.0000719 epoch_Time:43.0min:
Epoch:[0/1](23400/61701) loss:1.968 lr:0.0000717 epoch_Time:43.0min:
Epoch:[0/1](23500/61701) loss:1.844 lr:0.0000715 epoch_Time:43.0min:
Epoch:[0/1](23600/61701) loss:1.862 lr:0.0000712 epoch_Time:43.0min:
Epoch:[0/1](23700/61701) loss:2.038 lr:0.0000710 epoch_Time:43.0min:
Epoch:[0/1](23800/61701) loss:2.022 lr:0.0000708 epoch_Time:43.0min:
Epoch:[0/1](23900/61701) loss:1.831 lr:0.0000706 epoch_Time:42.0min:
Epoch:[0/1](24000/61701) loss:2.054 lr:0.0000704 epoch_Time:42.0min:
Epoch:[0/1](24100/61701) loss:2.223 lr:0.0000702 epoch_Time:42.0min:
Epoch:[0/1](24200/61701) loss:2.262 lr:0.0000699 epoch_Time:42.0min:
Epoch:[0/1](24300/61701) loss:1.954 lr:0.0000697 epoch_Time:42.0min:
Epoch:[0/1](24400/61701) loss:1.683 lr:0.0000695 epoch_Time:42.0min:
Epoch:[0/1](24500/61701) loss:1.736 lr:0.0000693 epoch_Time:42.0min:
Epoch:[0/1](24600/61701) loss:1.885 lr:0.0000691 epoch_Time:42.0min:
Epoch:[0/1](24700/61701) loss:2.126 lr:0.0000689 epoch_Time:42.0min:
Epoch:[0/1](24800/61701) loss:2.018 lr:0.0000686 epoch_Time:41.0min:
Epoch:[0/1](24900/61701) loss:2.038 lr:0.0000684 epoch_Time:41.0min:
Epoch:[0/1](25000/61701) loss:2.140 lr:0.0000682 epoch_Time:41.0min:
Epoch:[0/1](25100/61701) loss:1.910 lr:0.0000680 epoch_Time:41.0min:
Epoch:[0/1](25200/61701) loss:2.303 lr:0.0000678 epoch_Time:41.0min:
Epoch:[0/1](25300/61701) loss:2.054 lr:0.0000675 epoch_Time:41.0min:
Epoch:[0/1](25400/61701) loss:1.928 lr:0.0000673 epoch_Time:41.0min:
Epoch:[0/1](25500/61701) loss:1.946 lr:0.0000671 epoch_Time:41.0min:
Epoch:[0/1](25600/61701) loss:1.789 lr:0.0000669 epoch_Time:41.0min:
Epoch:[0/1](25700/61701) loss:1.607 lr:0.0000667 epoch_Time:40.0min:
Epoch:[0/1](25800/61701) loss:1.994 lr:0.0000664 epoch_Time:40.0min:
Epoch:[0/1](25900/61701) loss:1.936 lr:0.0000662 epoch_Time:40.0min:
Epoch:[0/1](26000/61701) loss:2.107 lr:0.0000660 epoch_Time:40.0min:
Epoch:[0/1](26100/61701) loss:2.127 lr:0.0000658 epoch_Time:40.0min:
Epoch:[0/1](26200/61701) loss:2.022 lr:0.0000656 epoch_Time:40.0min:
Epoch:[0/1](26300/61701) loss:1.951 lr:0.0000653 epoch_Time:40.0min:
Epoch:[0/1](26400/61701) loss:1.957 lr:0.0000651 epoch_Time:40.0min:
```

Epoch:[0/1](26500/61701) loss:2.111 lr:0.0000649 epoch_Time:40.0min:
Epoch:[0/1](26600/61701) loss:1.708 lr:0.0000647 epoch_Time:39.0min:
Epoch:[0/1](26700/61701) loss:2.007 lr:0.0000644 epoch_Time:39.0min:
Epoch:[0/1](26800/61701) loss:1.717 lr:0.0000642 epoch_Time:39.0min:
Epoch:[0/1](26900/61701) loss:1.767 lr:0.0000640 epoch_Time:39.0min:
Epoch:[0/1](27000/61701) loss:1.962 lr:0.0000638 epoch_Time:39.0min:
Epoch:[0/1](27100/61701) loss:2.191 lr:0.0000635 epoch_Time:39.0min:
Epoch:[0/1](27200/61701) loss:2.035 lr:0.0000633 epoch_Time:39.0min:
Epoch:[0/1](27300/61701) loss:1.980 lr:0.0000631 epoch_Time:39.0min:
Epoch:[0/1](27400/61701) loss:1.770 lr:0.0000629 epoch_Time:39.0min:
Epoch:[0/1](27500/61701) loss:1.931 lr:0.0000626 epoch_Time:38.0min:
Epoch:[0/1](27600/61701) loss:1.768 lr:0.0000624 epoch_Time:38.0min:
Epoch:[0/1](27700/61701) loss:1.823 lr:0.0000622 epoch_Time:38.0min:
Epoch:[0/1](27800/61701) loss:1.988 lr:0.0000620 epoch_Time:38.0min:
Epoch:[0/1](27900/61701) loss:2.006 lr:0.0000617 epoch_Time:38.0min:
Epoch:[0/1](28000/61701) loss:2.124 lr:0.0000615 epoch_Time:38.0min:
Epoch:[0/1](28100/61701) loss:1.939 lr:0.0000613 epoch_Time:38.0min:
Epoch:[0/1](28200/61701) loss:1.950 lr:0.0000611 epoch_Time:38.0min:
Epoch:[0/1](28300/61701) loss:1.891 lr:0.0000608 epoch_Time:38.0min:
Epoch:[0/1](28400/61701) loss:2.145 lr:0.0000606 epoch_Time:37.0min:
Epoch:[0/1](28500/61701) loss:1.872 lr:0.0000604 epoch_Time:37.0min:
Epoch:[0/1](28600/61701) loss:1.931 lr:0.0000601 epoch_Time:37.0min:
Epoch:[0/1](28700/61701) loss:1.819 lr:0.0000599 epoch_Time:37.0min:
Epoch:[0/1](28800/61701) loss:1.805 lr:0.0000597 epoch_Time:37.0min:
Epoch:[0/1](28900/61701) loss:2.022 lr:0.0000595 epoch_Time:37.0min:
Epoch:[0/1](29000/61701) loss:1.769 lr:0.0000592 epoch_Time:37.0min:
Epoch:[0/1](29100/61701) loss:1.884 lr:0.0000590 epoch_Time:37.0min:
Epoch:[0/1](29200/61701) loss:2.093 lr:0.0000588 epoch_Time:36.0min:
Epoch:[0/1](29300/61701) loss:1.951 lr:0.0000585 epoch_Time:36.0min:
Epoch:[0/1](29400/61701) loss:1.954 lr:0.0000583 epoch_Time:36.0min:
Epoch:[0/1](29500/61701) loss:1.921 lr:0.0000581 epoch_Time:36.0min:
Epoch:[0/1](29600/61701) loss:2.099 lr:0.0000579 epoch_Time:36.0min:
Epoch:[0/1](29700/61701) loss:2.097 lr:0.0000576 epoch_Time:36.0min:
Epoch:[0/1](29800/61701) loss:1.976 lr:0.0000574 epoch_Time:36.0min:
Epoch:[0/1](29900/61701) loss:1.862 lr:0.0000572 epoch_Time:36.0min:
Epoch:[0/1](30000/61701) loss:1.760 lr:0.0000569 epoch_Time:36.0min:
Epoch:[0/1](30100/61701) loss:1.750 lr:0.0000567 epoch_Time:35.0min:
Epoch:[0/1](30200/61701) loss:1.707 lr:0.0000565 epoch_Time:35.0min:
Epoch:[0/1](30300/61701) loss:1.894 lr:0.0000563 epoch_Time:35.0min:
Epoch:[0/1](30400/61701) loss:2.022 lr:0.0000560 epoch_Time:35.0min:
Epoch:[0/1](30500/61701) loss:1.744 lr:0.0000558 epoch_Time:35.0min:
Epoch:[0/1](30600/61701) loss:1.930 lr:0.0000556 epoch_Time:35.0min:
Epoch:[0/1](30700/61701) loss:1.968 lr:0.0000553 epoch_Time:35.0min:
Epoch:[0/1](30800/61701) loss:1.765 lr:0.0000551 epoch_Time:35.0min:
Epoch:[0/1](30900/61701) loss:2.068 lr:0.0000549 epoch_Time:35.0min:
Epoch:[0/1](31000/61701) loss:1.957 lr:0.0000547 epoch_Time:34.0min:
Epoch:[0/1](31100/61701) loss:1.937 lr:0.0000544 epoch_Time:34.0min:
Epoch:[0/1](31200/61701) loss:1.850 lr:0.0000542 epoch_Time:34.0min:

Epoch:[0/1](31300/61701) loss:1.990 lr:0.0000540 epoch_Time:34.0min:
Epoch:[0/1](31400/61701) loss:1.987 lr:0.0000537 epoch_Time:34.0min:
Epoch:[0/1](31500/61701) loss:1.810 lr:0.0000535 epoch_Time:34.0min:
Epoch:[0/1](31600/61701) loss:1.627 lr:0.0000533 epoch_Time:34.0min:
Epoch:[0/1](31700/61701) loss:2.165 lr:0.0000531 epoch_Time:34.0min:
Epoch:[0/1](31800/61701) loss:1.848 lr:0.0000528 epoch_Time:34.0min:
Epoch:[0/1](31900/61701) loss:2.084 lr:0.0000526 epoch_Time:33.0min:
Epoch:[0/1](32000/61701) loss:2.115 lr:0.0000524 epoch_Time:33.0min:
Epoch:[0/1](32100/61701) loss:2.138 lr:0.0000521 epoch_Time:33.0min:
Epoch:[0/1](32200/61701) loss:2.212 lr:0.0000519 epoch_Time:33.0min:
Epoch:[0/1](32300/61701) loss:2.288 lr:0.0000517 epoch_Time:33.0min:
Epoch:[0/1](32400/61701) loss:1.901 lr:0.0000515 epoch_Time:33.0min:
Epoch:[0/1](32500/61701) loss:1.899 lr:0.0000512 epoch_Time:33.0min:
Epoch:[0/1](32600/61701) loss:2.143 lr:0.0000510 epoch_Time:33.0min:
Epoch:[0/1](32700/61701) loss:2.053 lr:0.0000508 epoch_Time:33.0min:
Epoch:[0/1](32800/61701) loss:1.945 lr:0.0000505 epoch_Time:32.0min:
Epoch:[0/1](32900/61701) loss:2.076 lr:0.0000503 epoch_Time:32.0min:
Epoch:[0/1](33000/61701) loss:1.869 lr:0.0000501 epoch_Time:32.0min:
Epoch:[0/1](33100/61701) loss:1.912 lr:0.0000499 epoch_Time:32.0min:
Epoch:[0/1](33200/61701) loss:1.987 lr:0.0000496 epoch_Time:32.0min:
Epoch:[0/1](33300/61701) loss:1.821 lr:0.0000494 epoch_Time:32.0min:
Epoch:[0/1](33400/61701) loss:2.048 lr:0.0000492 epoch_Time:32.0min:
Epoch:[0/1](33500/61701) loss:2.051 lr:0.0000489 epoch_Time:32.0min:
Epoch:[0/1](33600/61701) loss:1.979 lr:0.0000487 epoch_Time:32.0min:
Epoch:[0/1](33700/61701) loss:1.948 lr:0.0000485 epoch_Time:31.0min:
Epoch:[0/1](33800/61701) loss:2.051 lr:0.0000483 epoch_Time:31.0min:
Epoch:[0/1](33900/61701) loss:2.059 lr:0.0000480 epoch_Time:31.0min:
Epoch:[0/1](34000/61701) loss:1.912 lr:0.0000478 epoch_Time:31.0min:
Epoch:[0/1](34100/61701) loss:2.186 lr:0.0000476 epoch_Time:31.0min:
Epoch:[0/1](34200/61701) loss:2.013 lr:0.0000474 epoch_Time:31.0min:
Epoch:[0/1](34300/61701) loss:1.914 lr:0.0000471 epoch_Time:31.0min:
Epoch:[0/1](34400/61701) loss:2.138 lr:0.0000469 epoch_Time:31.0min:
Epoch:[0/1](34500/61701) loss:2.049 lr:0.0000467 epoch_Time:31.0min:
Epoch:[0/1](34600/61701) loss:1.858 lr:0.0000465 epoch_Time:30.0min:
Epoch:[0/1](34700/61701) loss:1.926 lr:0.0000462 epoch_Time:30.0min:
Epoch:[0/1](34800/61701) loss:1.847 lr:0.0000460 epoch_Time:30.0min:
Epoch:[0/1](34900/61701) loss:1.860 lr:0.0000458 epoch_Time:30.0min:
Epoch:[0/1](35000/61701) loss:1.806 lr:0.0000456 epoch_Time:30.0min:
Epoch:[0/1](35100/61701) loss:1.879 lr:0.0000453 epoch_Time:30.0min:
Epoch:[0/1](35200/61701) loss:1.877 lr:0.0000451 epoch_Time:30.0min:
Epoch:[0/1](35300/61701) loss:2.082 lr:0.0000449 epoch_Time:30.0min:
Epoch:[0/1](35400/61701) loss:1.910 lr:0.0000447 epoch_Time:29.0min:
Epoch:[0/1](35500/61701) loss:1.799 lr:0.0000444 epoch_Time:29.0min:
Epoch:[0/1](35600/61701) loss:1.875 lr:0.0000442 epoch_Time:29.0min:
Epoch:[0/1](35700/61701) loss:1.832 lr:0.0000440 epoch_Time:29.0min:
Epoch:[0/1](35800/61701) loss:2.067 lr:0.0000438 epoch_Time:29.0min:
Epoch:[0/1](35900/61701) loss:1.739 lr:0.0000436 epoch_Time:29.0min:
Epoch:[0/1](36000/61701) loss:1.944 lr:0.0000433 epoch_Time:29.0min:

```
Epoch:[0/1](36100/61701) loss:1.884 lr:0.0000431 epoch_Time:29.0min:
Epoch:[0/1](36200/61701) loss:2.163 lr:0.0000429 epoch_Time:29.0min:
Epoch:[0/1](36300/61701) loss:1.571 lr:0.0000427 epoch_Time:28.0min:
Epoch:[0/1](36400/61701) loss:2.040 lr:0.0000425 epoch_Time:28.0min:
Epoch:[0/1](36500/61701) loss:1.668 lr:0.0000422 epoch_Time:28.0min:
Epoch:[0/1](36600/61701) loss:1.918 lr:0.0000420 epoch_Time:28.0min:
Epoch:[0/1](36700/61701) loss:1.814 lr:0.0000418 epoch_Time:28.0min:
Epoch:[0/1](36800/61701) loss:1.743 lr:0.0000416 epoch_Time:28.0min:
Epoch:[0/1](36900/61701) loss:1.921 lr:0.0000414 epoch_Time:28.0min:
Epoch:[0/1](37000/61701) loss:2.034 lr:0.0000411 epoch_Time:28.0min:
Epoch:[0/1](37100/61701) loss:1.659 lr:0.0000409 epoch_Time:28.0min:
Epoch:[0/1](37200/61701) loss:1.914 lr:0.0000407 epoch_Time:27.0min:
Epoch:[0/1](37300/61701) loss:2.018 lr:0.0000405 epoch_Time:27.0min:
Epoch:[0/1](37400/61701) loss:1.853 lr:0.0000403 epoch_Time:27.0min:
Epoch:[0/1](37500/61701) loss:2.010 lr:0.0000401 epoch_Time:27.0min:
Epoch:[0/1](37600/61701) loss:1.892 lr:0.0000398 epoch_Time:27.0min:
Epoch:[0/1](37700/61701) loss:1.751 lr:0.0000396 epoch_Time:27.0min:
Epoch:[0/1](37800/61701) loss:1.969 lr:0.0000394 epoch_Time:27.0min:
Epoch:[0/1](37900/61701) loss:1.859 lr:0.0000392 epoch_Time:27.0min:
Epoch:[0/1](38000/61701) loss:1.883 lr:0.0000390 epoch_Time:27.0min:
Epoch:[0/1](38100/61701) loss:2.013 lr:0.0000388 epoch_Time:26.0min:
Epoch:[0/1](38200/61701) loss:1.949 lr:0.0000386 epoch_Time:26.0min:
Epoch:[0/1](38300/61701) loss:1.878 lr:0.0000383 epoch_Time:26.0min:
Epoch:[0/1](38400/61701) loss:1.719 lr:0.0000381 epoch_Time:26.0min:
Epoch:[0/1](38500/61701) loss:1.989 lr:0.0000379 epoch_Time:26.0min:
Epoch:[0/1](38600/61701) loss:1.804 lr:0.0000377 epoch_Time:26.0min:
Epoch:[0/1](38700/61701) loss:1.906 lr:0.0000375 epoch_Time:26.0min:
Epoch:[0/1](38800/61701) loss:1.813 lr:0.0000373 epoch_Time:26.0min:
Epoch:[0/1](38900/61701) loss:2.003 lr:0.0000371 epoch_Time:26.0min:
Epoch:[0/1](39000/61701) loss:1.871 lr:0.0000369 epoch_Time:25.0min:
Epoch:[0/1](39100/61701) loss:1.605 lr:0.0000366 epoch_Time:25.0min:
Epoch:[0/1](39200/61701) loss:1.856 lr:0.0000364 epoch_Time:25.0min:
Epoch:[0/1](39300/61701) loss:2.053 lr:0.0000362 epoch_Time:25.0min:
Epoch:[0/1](39400/61701) loss:1.810 lr:0.0000360 epoch_Time:25.0min:
Epoch:[0/1](39500/61701) loss:1.818 lr:0.0000358 epoch_Time:25.0min:
Epoch:[0/1](39600/61701) loss:1.776 lr:0.0000356 epoch_Time:25.0min:
Epoch:[0/1](39700/61701) loss:2.196 lr:0.0000354 epoch_Time:25.0min:
Epoch:[0/1](39800/61701) loss:1.580 lr:0.0000352 epoch_Time:25.0min:
Epoch:[0/1](39900/61701) loss:2.055 lr:0.0000350 epoch_Time:24.0min:
Epoch:[0/1](40000/61701) loss:1.935 lr:0.0000348 epoch_Time:24.0min:
Epoch:[0/1](40100/61701) loss:1.963 lr:0.0000346 epoch_Time:24.0min:
Epoch:[0/1](40200/61701) loss:1.964 lr:0.0000344 epoch_Time:24.0min:
Epoch:[0/1](40300/61701) loss:1.986 lr:0.0000342 epoch_Time:24.0min:
Epoch:[0/1](40400/61701) loss:1.990 lr:0.0000340 epoch_Time:24.0min:
Epoch:[0/1](40500/61701) loss:1.789 lr:0.0000338 epoch_Time:24.0min:
Epoch:[0/1](40600/61701) loss:1.864 lr:0.0000336 epoch_Time:24.0min:
Epoch:[0/1](40700/61701) loss:1.995 lr:0.0000334 epoch_Time:23.0min:
Epoch:[0/1](40800/61701) loss:1.832 lr:0.0000332 epoch_Time:23.0min:
```

Epoch:[0/1](40900/61701) loss:1.821 lr:0.0000330 epoch_Time:23.0min:
Epoch:[0/1](41000/61701) loss:1.901 lr:0.0000328 epoch_Time:23.0min:
Epoch:[0/1](41100/61701) loss:1.959 lr:0.0000326 epoch_Time:23.0min:
Epoch:[0/1](41200/61701) loss:1.632 lr:0.0000324 epoch_Time:23.0min:
Epoch:[0/1](41300/61701) loss:2.194 lr:0.0000322 epoch_Time:23.0min:
Epoch:[0/1](41400/61701) loss:2.066 lr:0.0000320 epoch_Time:23.0min:
Epoch:[0/1](41500/61701) loss:1.965 lr:0.0000318 epoch_Time:23.0min:
Epoch:[0/1](41600/61701) loss:1.675 lr:0.0000316 epoch_Time:22.0min:
Epoch:[0/1](41700/61701) loss:1.743 lr:0.0000314 epoch_Time:22.0min:
Epoch:[0/1](41800/61701) loss:1.999 lr:0.0000312 epoch_Time:22.0min:
Epoch:[0/1](41900/61701) loss:1.922 lr:0.0000310 epoch_Time:22.0min:
Epoch:[0/1](42000/61701) loss:1.907 lr:0.0000308 epoch_Time:22.0min:
Epoch:[0/1](42100/61701) loss:1.881 lr:0.0000306 epoch_Time:22.0min:
Epoch:[0/1](42200/61701) loss:2.071 lr:0.0000304 epoch_Time:22.0min:
Epoch:[0/1](42300/61701) loss:1.880 lr:0.0000302 epoch_Time:22.0min:
Epoch:[0/1](42400/61701) loss:1.763 lr:0.0000300 epoch_Time:22.0min:
Epoch:[0/1](42500/61701) loss:2.219 lr:0.0000298 epoch_Time:21.0min:
Epoch:[0/1](42600/61701) loss:1.808 lr:0.0000297 epoch_Time:21.0min:
Epoch:[0/1](42700/61701) loss:1.917 lr:0.0000295 epoch_Time:21.0min:
Epoch:[0/1](42800/61701) loss:2.088 lr:0.0000293 epoch_Time:21.0min:
Epoch:[0/1](42900/61701) loss:1.662 lr:0.0000291 epoch_Time:21.0min:
Epoch:[0/1](43000/61701) loss:1.810 lr:0.0000289 epoch_Time:21.0min:
Epoch:[0/1](43100/61701) loss:2.108 lr:0.0000287 epoch_Time:21.0min:
Epoch:[0/1](43200/61701) loss:1.821 lr:0.0000285 epoch_Time:21.0min:
Epoch:[0/1](43300/61701) loss:1.686 lr:0.0000283 epoch_Time:21.0min:
Epoch:[0/1](43400/61701) loss:2.097 lr:0.0000282 epoch_Time:20.0min:
Epoch:[0/1](43500/61701) loss:2.142 lr:0.0000280 epoch_Time:20.0min:
Epoch:[0/1](43600/61701) loss:1.777 lr:0.0000278 epoch_Time:20.0min:
Epoch:[0/1](43700/61701) loss:1.862 lr:0.0000276 epoch_Time:20.0min:
Epoch:[0/1](43800/61701) loss:1.992 lr:0.0000274 epoch_Time:20.0min:
Epoch:[0/1](43900/61701) loss:2.092 lr:0.0000273 epoch_Time:20.0min:
Epoch:[0/1](44000/61701) loss:1.926 lr:0.0000271 epoch_Time:20.0min:
Epoch:[0/1](44100/61701) loss:1.979 lr:0.0000269 epoch_Time:20.0min:
Epoch:[0/1](44200/61701) loss:1.881 lr:0.0000267 epoch_Time:20.0min:
Epoch:[0/1](44300/61701) loss:2.009 lr:0.0000265 epoch_Time:19.0min:
Epoch:[0/1](44400/61701) loss:1.680 lr:0.0000264 epoch_Time:19.0min:
Epoch:[0/1](44500/61701) loss:2.127 lr:0.0000262 epoch_Time:19.0min:
Epoch:[0/1](44600/61701) loss:1.988 lr:0.0000260 epoch_Time:19.0min:
Epoch:[0/1](44700/61701) loss:1.844 lr:0.0000258 epoch_Time:19.0min:
Epoch:[0/1](44800/61701) loss:2.168 lr:0.0000257 epoch_Time:19.0min:
Epoch:[0/1](44900/61701) loss:2.018 lr:0.0000255 epoch_Time:19.0min:
Epoch:[0/1](45000/61701) loss:1.908 lr:0.0000253 epoch_Time:19.0min:
Epoch:[0/1](45100/61701) loss:1.989 lr:0.0000251 epoch_Time:19.0min:
Epoch:[0/1](45200/61701) loss:2.004 lr:0.0000250 epoch_Time:18.0min:
Epoch:[0/1](45300/61701) loss:1.773 lr:0.0000248 epoch_Time:18.0min:
Epoch:[0/1](45400/61701) loss:1.759 lr:0.0000246 epoch_Time:18.0min:
Epoch:[0/1](45500/61701) loss:1.672 lr:0.0000245 epoch_Time:18.0min:
Epoch:[0/1](45600/61701) loss:2.076 lr:0.0000243 epoch_Time:18.0min:

```
Epoch:[0/1](45700/61701) loss:1.969 lr:0.0000241 epoch_Time:18.0min:
Epoch:[0/1](45800/61701) loss:2.252 lr:0.0000240 epoch_Time:18.0min:
Epoch:[0/1](45900/61701) loss:1.856 lr:0.0000238 epoch_Time:18.0min:
Epoch:[0/1](46000/61701) loss:1.795 lr:0.0000236 epoch_Time:18.0min:
Epoch:[0/1](46100/61701) loss:2.097 lr:0.0000235 epoch_Time:17.0min:
Epoch:[0/1](46200/61701) loss:1.966 lr:0.0000233 epoch_Time:17.0min:
Epoch:[0/1](46300/61701) loss:1.912 lr:0.0000231 epoch_Time:17.0min:
Epoch:[0/1](46400/61701) loss:1.950 lr:0.0000230 epoch_Time:17.0min:
Epoch:[0/1](46500/61701) loss:1.791 lr:0.0000228 epoch_Time:17.0min:
Epoch:[0/1](46600/61701) loss:1.806 lr:0.0000227 epoch_Time:17.0min:
Epoch:[0/1](46700/61701) loss:1.752 lr:0.0000225 epoch_Time:17.0min:
Epoch:[0/1](46800/61701) loss:1.782 lr:0.0000223 epoch_Time:17.0min:
Epoch:[0/1](46900/61701) loss:1.482 lr:0.0000222 epoch_Time:16.0min:
Epoch:[0/1](47000/61701) loss:1.927 lr:0.0000220 epoch_Time:16.0min:
Epoch:[0/1](47100/61701) loss:1.788 lr:0.0000219 epoch_Time:16.0min:
Epoch:[0/1](47200/61701) loss:1.884 lr:0.0000217 epoch_Time:16.0min:
Epoch:[0/1](47300/61701) loss:1.951 lr:0.0000216 epoch_Time:16.0min:
Epoch:[0/1](47400/61701) loss:1.819 lr:0.0000214 epoch_Time:16.0min:
Epoch:[0/1](47500/61701) loss:1.803 lr:0.0000213 epoch_Time:16.0min:
Epoch:[0/1](47600/61701) loss:1.793 lr:0.0000211 epoch_Time:16.0min:
Epoch:[0/1](47700/61701) loss:1.819 lr:0.0000210 epoch_Time:16.0min:
Epoch:[0/1](47800/61701) loss:1.667 lr:0.0000208 epoch_Time:15.0min:
Epoch:[0/1](47900/61701) loss:1.718 lr:0.0000207 epoch_Time:15.0min:
Epoch:[0/1](48000/61701) loss:1.702 lr:0.0000205 epoch_Time:15.0min:
Epoch:[0/1](48100/61701) loss:1.965 lr:0.0000204 epoch_Time:15.0min:
Epoch:[0/1](48200/61701) loss:1.784 lr:0.0000202 epoch_Time:15.0min:
Epoch:[0/1](48300/61701) loss:1.812 lr:0.0000201 epoch_Time:15.0min:
Epoch:[0/1](48400/61701) loss:1.926 lr:0.0000199 epoch_Time:15.0min:
Epoch:[0/1](48500/61701) loss:1.603 lr:0.0000198 epoch_Time:15.0min:
Epoch:[0/1](48600/61701) loss:1.757 lr:0.0000196 epoch_Time:15.0min:
Epoch:[0/1](48700/61701) loss:1.951 lr:0.0000195 epoch_Time:14.0min:
Epoch:[0/1](48800/61701) loss:1.968 lr:0.0000194 epoch_Time:14.0min:
Epoch:[0/1](48900/61701) loss:1.982 lr:0.0000192 epoch_Time:14.0min:
Epoch:[0/1](49000/61701) loss:1.910 lr:0.0000191 epoch_Time:14.0min:
Epoch:[0/1](49100/61701) loss:1.833 lr:0.0000189 epoch_Time:14.0min:
Epoch:[0/1](49200/61701) loss:1.767 lr:0.0000188 epoch_Time:14.0min:
Epoch:[0/1](49300/61701) loss:1.983 lr:0.0000187 epoch_Time:14.0min:
Epoch:[0/1](49400/61701) loss:2.277 lr:0.0000185 epoch_Time:14.0min:
Epoch:[0/1](49500/61701) loss:1.908 lr:0.0000184 epoch_Time:14.0min:
Epoch:[0/1](49600/61701) loss:1.768 lr:0.0000183 epoch_Time:13.0min:
Epoch:[0/1](49700/61701) loss:2.053 lr:0.0000181 epoch_Time:13.0min:
Epoch:[0/1](49800/61701) loss:1.992 lr:0.0000180 epoch_Time:13.0min:
Epoch:[0/1](49900/61701) loss:2.016 lr:0.0000179 epoch_Time:13.0min:
Epoch:[0/1](50000/61701) loss:1.803 lr:0.0000178 epoch_Time:13.0min:
Epoch:[0/1](50100/61701) loss:1.894 lr:0.0000176 epoch_Time:13.0min:
Epoch:[0/1](50200/61701) loss:1.768 lr:0.0000175 epoch_Time:13.0min:
Epoch:[0/1](50300/61701) loss:1.889 lr:0.0000174 epoch_Time:13.0min:
Epoch:[0/1](50400/61701) loss:2.002 lr:0.0000172 epoch_Time:13.0min:
```

```
Epoch:[0/1](50500/61701) loss:1.907 lr:0.0000171 epoch_Time:12.0min:
Epoch:[0/1](50600/61701) loss:1.908 lr:0.0000170 epoch_Time:12.0min:
Epoch:[0/1](50700/61701) loss:1.784 lr:0.0000169 epoch_Time:12.0min:
Epoch:[0/1](50800/61701) loss:1.667 lr:0.0000168 epoch_Time:12.0min:
Epoch:[0/1](50900/61701) loss:1.743 lr:0.0000166 epoch_Time:12.0min:
Epoch:[0/1](51000/61701) loss:1.758 lr:0.0000165 epoch_Time:12.0min:
Epoch:[0/1](51100/61701) loss:1.747 lr:0.0000164 epoch_Time:12.0min:
Epoch:[0/1](51200/61701) loss:1.853 lr:0.0000163 epoch_Time:12.0min:
Epoch:[0/1](51300/61701) loss:1.830 lr:0.0000162 epoch_Time:12.0min:
Epoch:[0/1](51400/61701) loss:2.089 lr:0.0000160 epoch_Time:11.0min:
Epoch:[0/1](51500/61701) loss:1.991 lr:0.0000159 epoch_Time:11.0min:
Epoch:[0/1](51600/61701) loss:1.738 lr:0.0000158 epoch_Time:11.0min:
Epoch:[0/1](51700/61701) loss:1.857 lr:0.0000157 epoch_Time:11.0min:
Epoch:[0/1](51800/61701) loss:1.950 lr:0.0000156 epoch_Time:11.0min:
Epoch:[0/1](51900/61701) loss:1.970 lr:0.0000155 epoch_Time:11.0min:
Epoch:[0/1](52000/61701) loss:1.921 lr:0.0000154 epoch_Time:11.0min:
Epoch:[0/1](52100/61701) loss:1.920 lr:0.0000153 epoch_Time:11.0min:
Epoch:[0/1](52200/61701) loss:2.019 lr:0.0000152 epoch_Time:11.0min:
Epoch:[0/1](52300/61701) loss:1.930 lr:0.0000151 epoch_Time:10.0min:
Epoch:[0/1](52400/61701) loss:1.786 lr:0.0000150 epoch_Time:10.0min:
Epoch:[0/1](52500/61701) loss:1.722 lr:0.0000148 epoch_Time:10.0min:
Epoch:[0/1](52600/61701) loss:1.717 lr:0.0000147 epoch_Time:10.0min:
Epoch:[0/1](52700/61701) loss:1.918 lr:0.0000146 epoch_Time:10.0min:
Epoch:[0/1](52800/61701) loss:1.663 lr:0.0000145 epoch_Time:10.0min:
Epoch:[0/1](52900/61701) loss:1.918 lr:0.0000144 epoch_Time:10.0min:
Epoch:[0/1](53000/61701) loss:1.904 lr:0.0000143 epoch_Time:10.0min:
Epoch:[0/1](53100/61701) loss:1.728 lr:0.0000142 epoch_Time:9.0min:
Epoch:[0/1](53200/61701) loss:1.860 lr:0.0000141 epoch_Time:9.0min:
Epoch:[0/1](53300/61701) loss:1.649 lr:0.0000141 epoch_Time:9.0min:
Epoch:[0/1](53400/61701) loss:1.927 lr:0.0000140 epoch_Time:9.0min:
Epoch:[0/1](53500/61701) loss:2.047 lr:0.0000139 epoch_Time:9.0min:
Epoch:[0/1](53600/61701) loss:1.909 lr:0.0000138 epoch_Time:9.0min:
Epoch:[0/1](53700/61701) loss:1.976 lr:0.0000137 epoch_Time:9.0min:
Epoch:[0/1](53800/61701) loss:2.162 lr:0.0000136 epoch_Time:9.0min:
Epoch:[0/1](53900/61701) loss:1.917 lr:0.0000135 epoch_Time:9.0min:
Epoch:[0/1](54000/61701) loss:1.688 lr:0.0000134 epoch_Time:8.0min:
Epoch:[0/1](54100/61701) loss:1.836 lr:0.0000133 epoch_Time:8.0min:
Epoch:[0/1](54200/61701) loss:1.938 lr:0.0000132 epoch_Time:8.0min:
Epoch:[0/1](54300/61701) loss:1.656 lr:0.0000132 epoch_Time:8.0min:
Epoch:[0/1](54400/61701) loss:1.916 lr:0.0000131 epoch_Time:8.0min:
Epoch:[0/1](54500/61701) loss:1.828 lr:0.0000130 epoch_Time:8.0min:
Epoch:[0/1](54600/61701) loss:1.728 lr:0.0000129 epoch_Time:8.0min:
Epoch:[0/1](54700/61701) loss:1.878 lr:0.0000128 epoch_Time:8.0min:
Epoch:[0/1](54800/61701) loss:2.100 lr:0.0000127 epoch_Time:8.0min:
Epoch:[0/1](54900/61701) loss:1.689 lr:0.0000127 epoch_Time:7.0min:
Epoch:[0/1](55000/61701) loss:1.759 lr:0.0000126 epoch_Time:7.0min:
Epoch:[0/1](55100/61701) loss:1.670 lr:0.0000125 epoch_Time:7.0min:
Epoch:[0/1](55200/61701) loss:2.014 lr:0.0000124 epoch_Time:7.0min:
```

```
Epoch:[0/1](55300/61701) loss:2.112 lr:0.0000124 epoch_Time:7.0min:
Epoch:[0/1](55400/61701) loss:1.670 lr:0.0000123 epoch_Time:7.0min:
Epoch:[0/1](55500/61701) loss:1.744 lr:0.0000122 epoch_Time:7.0min:
Epoch:[0/1](55600/61701) loss:1.835 lr:0.0000122 epoch_Time:7.0min:
Epoch:[0/1](55700/61701) loss:2.003 lr:0.0000121 epoch_Time:7.0min:
Epoch:[0/1](55800/61701) loss:1.971 lr:0.0000120 epoch_Time:6.0min:
Epoch:[0/1](55900/61701) loss:1.849 lr:0.0000119 epoch_Time:6.0min:
Epoch:[0/1](56000/61701) loss:1.914 lr:0.0000119 epoch_Time:6.0min:
Epoch:[0/1](56100/61701) loss:1.855 lr:0.0000118 epoch_Time:6.0min:
Epoch:[0/1](56200/61701) loss:1.837 lr:0.0000118 epoch_Time:6.0min:
Epoch:[0/1](56300/61701) loss:1.546 lr:0.0000117 epoch_Time:6.0min:
Epoch:[0/1](56400/61701) loss:1.952 lr:0.0000116 epoch_Time:6.0min:
Epoch:[0/1](56500/61701) loss:2.075 lr:0.0000116 epoch_Time:6.0min:
Epoch:[0/1](56600/61701) loss:1.845 lr:0.0000115 epoch_Time:6.0min:
Epoch:[0/1](56700/61701) loss:1.765 lr:0.0000115 epoch_Time:5.0min:
Epoch:[0/1](56800/61701) loss:1.864 lr:0.0000114 epoch_Time:5.0min:
Epoch:[0/1](56900/61701) loss:1.739 lr:0.0000113 epoch_Time:5.0min:
Epoch:[0/1](57000/61701) loss:1.894 lr:0.0000113 epoch_Time:5.0min:
Epoch:[0/1](57100/61701) loss:1.805 lr:0.0000112 epoch_Time:5.0min:
Epoch:[0/1](57200/61701) loss:1.910 lr:0.0000112 epoch_Time:5.0min:
Epoch:[0/1](57300/61701) loss:1.924 lr:0.0000111 epoch_Time:5.0min:
Epoch:[0/1](57400/61701) loss:1.968 lr:0.0000111 epoch_Time:5.0min:
Epoch:[0/1](57500/61701) loss:1.908 lr:0.0000110 epoch_Time:5.0min:
Epoch:[0/1](57600/61701) loss:1.903 lr:0.0000110 epoch_Time:4.0min:
Epoch:[0/1](57700/61701) loss:1.706 lr:0.0000109 epoch_Time:4.0min:
Epoch:[0/1](57800/61701) loss:1.844 lr:0.0000109 epoch_Time:4.0min:
Epoch:[0/1](57900/61701) loss:1.650 lr:0.0000108 epoch_Time:4.0min:
Epoch:[0/1](58000/61701) loss:1.756 lr:0.0000108 epoch_Time:4.0min:
Epoch:[0/1](58100/61701) loss:1.757 lr:0.0000108 epoch_Time:4.0min:
Epoch:[0/1](58200/61701) loss:1.775 lr:0.0000107 epoch_Time:4.0min:
Epoch:[0/1](58300/61701) loss:1.672 lr:0.0000107 epoch_Time:4.0min:
Epoch:[0/1](58400/61701) loss:1.708 lr:0.0000106 epoch_Time:3.0min:
Epoch:[0/1](58500/61701) loss:1.872 lr:0.0000106 epoch_Time:3.0min:
Epoch:[0/1](58600/61701) loss:2.062 lr:0.0000106 epoch_Time:3.0min:
Epoch:[0/1](58700/61701) loss:2.116 lr:0.0000105 epoch_Time:3.0min:
Epoch:[0/1](58800/61701) loss:1.973 lr:0.0000105 epoch_Time:3.0min:
Epoch:[0/1](58900/61701) loss:1.954 lr:0.0000105 epoch_Time:3.0min:
Epoch:[0/1](59000/61701) loss:1.851 lr:0.0000104 epoch_Time:3.0min:
Epoch:[0/1](59100/61701) loss:1.885 lr:0.0000104 epoch_Time:3.0min:
Epoch:[0/1](59200/61701) loss:1.936 lr:0.0000104 epoch_Time:3.0min:
Epoch:[0/1](59300/61701) loss:1.785 lr:0.0000103 epoch_Time:2.0min:
Epoch:[0/1](59400/61701) loss:1.920 lr:0.0000103 epoch_Time:2.0min:
Epoch:[0/1](59500/61701) loss:1.755 lr:0.0000103 epoch_Time:2.0min:
Epoch:[0/1](59600/61701) loss:2.107 lr:0.0000103 epoch_Time:2.0min:
Epoch:[0/1](59700/61701) loss:1.720 lr:0.0000102 epoch_Time:2.0min:
Epoch:[0/1](59800/61701) loss:1.992 lr:0.0000102 epoch_Time:2.0min:
Epoch:[0/1](59900/61701) loss:1.792 lr:0.0000102 epoch_Time:2.0min:
Epoch:[0/1](60000/61701) loss:1.854 lr:0.0000102 epoch_Time:2.0min:
```

```
Epoch:[0/1](60100/61701) loss:1.939 lr:0.0000101 epoch_Time:2.0min:
Epoch:[0/1](60200/61701) loss:1.779 lr:0.0000101 epoch_Time:1.0min:
Epoch:[0/1](60300/61701) loss:1.739 lr:0.0000101 epoch_Time:1.0min:
Epoch:[0/1](60400/61701) loss:1.914 lr:0.0000101 epoch_Time:1.0min:
Epoch:[0/1](60500/61701) loss:1.735 lr:0.0000101 epoch_Time:1.0min:
Epoch:[0/1](60600/61701) loss:1.920 lr:0.0000101 epoch_Time:1.0min:
Epoch:[0/1](60700/61701) loss:1.772 lr:0.0000101 epoch_Time:1.0min:
Epoch:[0/1](60800/61701) loss:2.150 lr:0.0000100 epoch_Time:1.0min:
Epoch:[0/1](60900/61701) loss:1.777 lr:0.0000100 epoch_Time:1.0min:
Epoch:[0/1](61000/61701) loss:1.732 lr:0.0000100 epoch_Time:1.0min:
Epoch:[0/1](61100/61701) loss:1.773 lr:0.0000100 epoch_Time:0.0min:
Epoch:[0/1](61200/61701) loss:2.081 lr:0.0000100 epoch_Time:0.0min:
Epoch:[0/1](61300/61701) loss:1.887 lr:0.0000100 epoch_Time:0.0min:
Epoch:[0/1](61400/61701) loss:1.640 lr:0.0000100 epoch_Time:0.0min:
Epoch:[0/1](61500/61701) loss:2.037 lr:0.0000100 epoch_Time:0.0min:
Epoch:[0/1](61600/61701) loss:1.893 lr:0.0000100 epoch_Time:0.0min:
Epoch:[0/1](61700/61701) loss:1.862 lr:0.0000100 epoch_Time:0.0min:
```

4.4.3 3.4.3 Cross-Validated SFT Phase (using MLEnd dataset)

Objective: To fine-tune the pretrained model specifically for deception detection using the MLEnd dataset

Dataset: MLEnd Deception Dataset, consisting of 100 audio recordings with corresponding labels ("True Story" or "Deceptive Story").

Link: https://github.com/CBU5201Datasets/Deception

Training Procedure:

1. **Data Preparation:**
   - **Transcription:** The 100 audio recordings are first transcribed into text using OpenAI's Whisper ASR model.
   - **Labeling:** Each transcribed story is assigned a binary label: 0 for "Deceptive Story" and 1 for "True Story".
   - **Cross-Validation Split:** The dataset is divided into 5 folds using k-fold cross-validation (k=5). In each fold, 80 samples are used for training and 20 for validation.
   - **Tokenization:** Each transcribed story is tokenized using the **HOLMES-26M** tokenizer.
   - **Padding/Truncation:** Tokenized sequences are padded or truncated to `max_seq_len`.
2. **Model Input:** Similar to the previous SFT phase, the model takes the tokenized input sequence `X` and the target sequence `Y` (representing the label). A loss mask is used to focus on the relevant tokens for loss calculation.
3. **Loss Function:** The same cross-entropy loss function as in the previous SFT phase is used:

$$\text{Loss} = \frac{\sum(\text{loss} \cdot \text{loss\_mask})}{\sum \text{loss\_mask}}$$

4. **Optimization:**

- The model is fine-tuned using the **Adam optimizer**.
- A smaller learning rate (e.g., `5e-5`) might be more suitable for this smaller dataset.
- **Gradient accumulation** can be used if necessary.
- **Gradient clipping** is applied.
- The learning rate schedule can be adjusted (e.g., shorter warmup, fewer epochs).
- **Mixed-precision training** can be used.

5. **Cross-Validation:**
   - The SFT process is repeated for each of the 5 folds.
   - In each fold, the model is trained on the 80 training samples and validated on the 20 held-out samples.
   - **Five separate model checkpoints are saved**, one for each fold, representing the model's state after training on that specific fold. These checkpoints will be used for evaluation in section 3.5.
6. **Evaluation:** The model's performance in each fold is evaluated using metrics such as accuracy, precision, recall, and F1-score on the validation set. The average and standard deviation of these metrics across the 5 folds provide a robust estimate of the model's generalization ability.

Implementation Details:

- The script should iterate through the folds, train the model on the training set of each fold, and save the corresponding model checkpoint.
- The model is fine-tuned for 20 epochs.

7. **Implementation:**

the whole training process can be found at https://wandb.ai/mjuicem3-beijing-university-of-posts-and-telecommunications/HOLMES-Full-SFT?nw=nwusermjuicem3

[ ]:
```
!python 3-full_deception_sft.py
```

```
Total LLM parameters: 26.878M
Epoch:[0/40](0/4) loss:4.824 lr:0.0001000 epoch_Time:0.0min:
Saved checkpoint at epoch 0 to out/full_sft_100samples_512_epoch0.pth
Epoch:[1/40](0/4) loss:4.392 lr:0.0000999 epoch_Time:0.0min:
Epoch:[2/40](0/4) loss:3.778 lr:0.0000994 epoch_Time:0.0min:
Epoch:[3/40](0/4) loss:3.446 lr:0.0000988 epoch_Time:0.0min:
Epoch:[4/40](0/4) loss:3.172 lr:0.0000978 epoch_Time:0.0min:
Epoch:[5/40](0/4) loss:2.903 lr:0.0000966 epoch_Time:0.0min:
Epoch:[6/40](0/4) loss:2.624 lr:0.0000951 epoch_Time:0.0min:
Epoch:[7/40](0/4) loss:2.345 lr:0.0000934 epoch_Time:0.0min:
Epoch:[8/40](0/4) loss:2.074 lr:0.0000914 epoch_Time:0.0min:
Epoch:[9/40](0/4) loss:1.837 lr:0.0000892 epoch_Time:0.0min:
Epoch:[10/40](0/4) loss:1.643 lr:0.0000868 epoch_Time:0.0min:
Saved checkpoint at epoch 10 to out/full_sft_100samples_512_epoch10.pth
Epoch:[11/40](0/4) loss:1.472 lr:0.0000842 epoch_Time:0.0min:
Epoch:[12/40](0/4) loss:1.322 lr:0.0000815 epoch_Time:0.0min:
Epoch:[13/40](0/4) loss:1.180 lr:0.0000785 epoch_Time:0.0min:
Epoch:[14/40](0/4) loss:1.062 lr:0.0000754 epoch_Time:0.0min:
```

```
Epoch:[15/40](0/4) loss:0.953 lr:0.0000722 epoch_Time:0.0min:
Epoch:[16/40](0/4) loss:0.854 lr:0.0000689 epoch_Time:0.0min:
Epoch:[17/40](0/4) loss:0.769 lr:0.0000655 epoch_Time:0.0min:
Epoch:[18/40](0/4) loss:0.692 lr:0.0000620 epoch_Time:0.0min:
Epoch:[19/40](0/4) loss:0.628 lr:0.0000585 epoch_Time:0.0min:
Epoch:[20/40](0/4) loss:0.570 lr:0.0000550 epoch_Time:0.0min:
Saved checkpoint at epoch 20 to out/full_sft_100samples_512_epoch20.pth
Epoch:[21/40](0/4) loss:0.525 lr:0.0000515 epoch_Time:0.0min:
Epoch:[22/40](0/4) loss:0.483 lr:0.0000480 epoch_Time:0.0min:
Epoch:[23/40](0/4) loss:0.442 lr:0.0000445 epoch_Time:0.0min:
Epoch:[24/40](0/4) loss:0.413 lr:0.0000411 epoch_Time:0.0min:
Epoch:[25/40](0/4) loss:0.379 lr:0.0000378 epoch_Time:0.0min:
Epoch:[26/40](0/4) loss:0.350 lr:0.0000346 epoch_Time:0.0min:
Epoch:[27/40](0/4) loss:0.325 lr:0.0000315 epoch_Time:0.0min:
Epoch:[28/40](0/4) loss:0.301 lr:0.0000285 epoch_Time:0.0min:
Epoch:[29/40](0/4) loss:0.280 lr:0.0000258 epoch_Time:0.0min:
Epoch:[30/40](0/4) loss:0.260 lr:0.0000232 epoch_Time:0.0min:
Saved checkpoint at epoch 30 to out/full_sft_100samples_512_epoch30.pth
Epoch:[31/40](0/4) loss:0.242 lr:0.0000208 epoch_Time:0.0min:
Epoch:[32/40](0/4) loss:0.227 lr:0.0000186 epoch_Time:0.0min:
Epoch:[33/40](0/4) loss:0.213 lr:0.0000166 epoch_Time:0.0min:
Epoch:[34/40](0/4) loss:0.199 lr:0.0000149 epoch_Time:0.0min:
Epoch:[35/40](0/4) loss:0.187 lr:0.0000134 epoch_Time:0.0min:
Epoch:[36/40](0/4) loss:0.175 lr:0.0000122 epoch_Time:0.0min:
Epoch:[37/40](0/4) loss:0.164 lr:0.0000112 epoch_Time:0.0min:
Epoch:[38/40](0/4) loss:0.154 lr:0.0000106 epoch_Time:0.0min:
Epoch:[39/40](0/4) loss:0.146 lr:0.0000101 epoch_Time:0.0min:
Saved checkpoint at epoch 39 to out/full_sft_100samples_512_epoch39.pth
```

4.5 3.5 Validation Task

Objective: To rigorously evaluate the performance of **HOLMES-26M** on the deception detection task using the 5 model checkpoints obtained from the cross-validated SFT phase (Section 3.4.3). Each checkpoint, trained on a different fold of the MLEnd dataset, will be used to predict the labels of the 20 validation samples from the corresponding held-out fold. The performance will be quantified using the metrics described in Section 3.6: Accuracy, Precision, Recall, F1-score, and the confusion matrix.

Procedure:

1. **Load Checkpoints:** For each fold (from 1 to 5), load the corresponding model checkpoint saved during the cross-validated SFT phase.
2. **Load Validation Data:** For each fold, load the 20 validation samples (transcribed text and corresponding labels) that were held out during the training of that fold.
3. **Prediction:**
    - For each validation sample in the current fold:
        - Tokenize the transcribed text using the **HOLMES-26M** tokenizer.

- Pad or truncate the tokenized sequence to `max_seq_len`.
- Feed the preprocessed input to the model (loaded with the checkpoint for the current fold).
- Obtain the model's predicted probability for the "True Story" label (class 1).
- Convert the probability to a binary prediction (0 for "Deceptive Story", 1 for "True Story") using a threshold (typically 0.5).

4. **Evaluation:**

- For each fold:
  - Compare the model's predictions on the 20 validation samples with the ground truth labels.
  - Calculate the **Accuracy, Precision, Recall, and F1-score** based on the predictions and true labels.
  - Generate the **confusion matrix** to visualize the model's performance across the two classes.

5. **Aggregation:**

- Calculate the **average and standard deviation** of each metric (Accuracy, Precision, Recall, F1-score) across the 5 folds. This provides a robust estimate of the model's overall performance and its variability across different subsets of the data.
- Present the 5 confusion matrices (one for each fold) and optionally, an **aggregated confusion matrix** obtained by summing the counts across all folds.

4.6 3.6 Evaluation Metrics

We use Accuracy, Precision, Recall, F1-score, and Confusion Matrix to evaluate HOLMES-26M's performance.

- **Accuracy:** The percentage of correctly classified stories.
- **Precision:** The proportion of stories classified as True that are actually True.
- **Recall:** The proportion of True stories that are correctly classified as True.
- **F1-score:** The harmonic mean of precision and recall.
- **Confusion Matrix:** A table showing the counts of true positives, true negatives, false positives, and false negatives, providing a detailed breakdown of the model's performance.

  In True/Deceptive story classification:

- **True Positive (TP):** A story that is actually **True** and is correctly classified as **True** by the model.
- **True Negative (TN):** A story that is actually **Deceptive** and is correctly classified as **Deceptive** by the model.
- **False Positive (FP):** A story that is actually **Deceptive** but is incorrectly classified as **True** by the model (Type I error).
- **False Negative (FN):** A story that is actually **True** but is incorrectly classified as **Deceptive** by the model (Type II error).

Accuracy:

- **Formula:**

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

- **Explanation:** Accuracy represents the overall correctness of the model's predictions. It measures the proportion of correctly classified stories (both True and Deceptive) out of the total number of stories.
  - **Range:** 0 to 1 (or 0% to 100%)
  - **Interpretation:** A higher accuracy indicates a better overall performance of the model in correctly classifying stories.

Precision:

- **Formula:**

$$Precision = \frac{TP}{TP + FP}$$

- **Explanation:** Precision focuses on the stories classified as **True** by the model. It measures the proportion of correctly classified True stories out of all stories that the model predicted as True. In simpler terms, it answers the question: "Out of all the stories the model labeled as True, how many were actually True?"
  - **Range:** 0 to 1 (or 0% to 100%)
  - **Interpretation:** A higher precision indicates fewer false positives. It means that when the model predicts a story is True, it is more likely to be correct. This is important when the cost of a false positive is high.

Recall:

- **Formula:**

$$Recall = \frac{TP}{TP + FN}$$

- **Explanation:** Recall focuses on the stories that are actually **True**. It measures the proportion of correctly classified True stories out of all stories that are actually True. In other words, it answers the question: "Out of all the stories that are actually True, how many did the model correctly identify?"
  - **Range:** 0 to 1 (or 0% to 100%)
  - **Interpretation:** A higher recall indicates fewer false negatives. It signifies that the model is good at identifying actual True stories. This is crucial when the cost of a false negative is high.

F1-score:

- **Formula:**

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

- **Explanation:** The F1-score is the harmonic mean of precision and recall. It provides a balanced measure of the model's performance, considering both precision and recall.
  - **Range:** 0 to 1 (or 0% to 100%)
  - **Interpretation:** A higher F1-score indicates a better balance between precision and recall. It is a useful metric when you need to find a balance between minimizing false positives and false negatives, especially when the class distribution is imbalanced.

Confusion Matrix:

- **Description:** A confusion matrix is a table that visualizes the performance of a classification model by summarizing the counts of TP, TN, FP, and FN.
- **Structure:**

	Predicted Deceptive	Predicted True
Actual Deceptive	TN	FP
Actual True	FN	TP

- **Interpretation:** The confusion matrix provides a more detailed breakdown of the model's performance compared to using only accuracy. By examining the different cells of the matrix, you can gain insights into the types of errors the model is making and identify areas for improvement.

These metrics will be used to evaluate the performance of the HOLMES model in classifying stories as True or Deceptive. They provide a comprehensive assessment of the model's accuracy, ability to correctly identify True stories, and balance between minimizing different types of errors.

5 4 Implemented ML Prediction Pipelines

Overview:

The **HOLMES** System leverages a combination of automatic speech recognition (ASR) and a specialized language model for deception detection. The pipeline operates as follows:

1. **Transformation Stage:** The input audio recording is transcribed into text using OpenAI's **Whisper** ASR model. This stage converts the audio signal into a textual representation that can be processed by the subsequent stages.
2. **Model Stage:** The transcribed text is fed into **HOLMES-26M**, a fine-tuned language model based on the transformer architecture. This model processes the text and outputs "True Story" or "Deceptive Story".

Pipeline Input and Output:

- **Input:** A 3-5 minute audio recording of a narrated story (in `.wav` format).
- **Output:** "Deceptive Story" or "True Story".

Pipeline Stages and Intermediate Data Structures:

Stage	Input	Output	Intermediate Data Structure
Transformation	Audio recording (.wav)	Transcribed text (string)	Raw audio data (waveform) -> MFCC features -> Encoded audio features -> Decoded text (string)
Model	Transcribed text (string)	Probability score (float, between 0 and 1) for each model	Tokenized text -> Model embeddings -> Hidden states -> Logits -> Final Output

4.1 Transformation Stage

Objective: To convert the input audio recording into a textual representation that can be processed by the subsequent stages of the pipeline.

Input: A 3-5 minute audio recording of a narrated story (`.wav` format).

Output: The transcribed text of the narrated story (string).

This stage utilizes OpenAI's Whisper, a state-of-the-art automatic speech recognition (ASR) model, to transcribe the audio recording into text. The process involves the following steps:

1. **Audio Loading:** The audio recording is loaded from the input file.
2. **Feature Extraction:** Whisper's internal feature extraction process is applied. This typically involves:
    - Converting the raw audio waveform into a sequence of Mel-frequency cepstral coefficients (MFCCs)
    - Encoding these features into a representation that captures the relevant acoustic information.
3. **Speech Recognition:** Whisper's ASR engine processes the encoded audio features and generates the corresponding text transcript using its trained acoustic and language models. This involves:
    - Decoding the encoded features to generate a sequence of phonemes or subword units.
    - Using a language model to convert the phoneme/subword sequence into a coherent text transcript.
4. **Text Output:** The transcribed text is outputted as a string.

Advantage: - *Necessity*: Speech recognition is essential to convert the audio input into a textual format that can be processed by LLM with no native audio encoder.

- *Effectiveness*: Whisper is a highly accurate ASR model that has been shown to perform well on a wide range of speech recognition tasks. It is robust to different accents, speaking styles, and recording conditions.
- *Availability*: Whisper is readily available through Openai API. making it easy to integrate into the pipeline.

Implementation Details:

- The OpenAI's API will be used.

Implementation:

[6]:
```
!python whisper.py
```

```
Found 100 WAV files to process
Processing WAV files: 100%|          | 100/100 [00:50<00:00,  1.98file/s]
```

The transcript text can be found in ./text

4.2 Model Stage

Objective: To process the transcribed text, understanding the story and make a decision on whether the story is true or deceptives.

Input: The transcribed text of the narrated story (string).

Output: A probability score (float between 0 and 1) for each of the five models, indicating the model's confidence that the story is true.

This stage utilizes **HOLMES-26M**, a fine-tuned language model based on the **transformer architecture**, to analyze the transcribed text and predict the likelihood of deception. The process involves the following steps:

1. **Tokenization:** The input text is tokenized using the **HOLMES-26M** tokenizer, which converts the text into a sequence of numerical tokens that can be processed by the model.
2. **Padding/Truncation:** The tokenized sequence is padded or truncated to a fixed length (`max_seq_len`) to ensure consistent input dimensions.
3. **Model Forward Pass:** The preprocessed token sequence is fed into the **HOLMES-26M** model. The model processes the input through its layers, including the embedding layer, transformer blocks, and output layer.
4. **Probability Output:** The output layer produces a logit for each class (Deceptive/True). These logits are then passed through a softmax function to obtain probability scores for each class. The probability score for the "True Story" class (class 1) is extracted.

Advantage: - *Transformer*: The transformer architecture has proven to be highly effective for natural language processing tasks, including text classification. Its self-attention mechanism allows it to capture long-range dependencies and contextual relationships within the text, which are crucial for understanding nuanced language and detecting deception.

- *Pretraining and Fine-tuning*: HOLMES-26M leverages the power of pretraining on a large corpus (Seq-Monkey) to learn general language understanding and then fine-tuning on a smaller, task-specific dataset (Deepctrl-sft-data and MLEnd dataset) to specialize in deception detection. This approach allows the model to benefit from both general language knowledge and task-specific patterns.
- *Decoder-Only Structure*: The decoder-only structure, similar to GPT, is well-suited for text generation and classification tasks. It allows the model to generate coherent and contextually relevant text, and its output can be easily adapted for binary classification.
- *Specialized for Deception Detection*: HOLMES-26M is specifically fine-tuned for deception detection using the MLEnd dataset, enabling it to learn patterns and linguistic cues that are

indicative of deceptive language. The cross-validated SFT phase further enhances its ability to generalize to unseen data.

Implementation Details:

- The **HOLMES-26M** model, as described in Section 3.3, will be used.
- Five different instances of **HOLMES-26M**, each loaded with a different checkpoint from the cross-validated SFT phase, will be used.

Implementation:

4.3 Ensemble Stage

Given that we use a large language model (LLM) as our backbone, we decided not to implement ensemble methods. This is because:

1. The LLM itself already demonstrates strong performance
2. The computational cost of ensembling multiple LLMs would be prohibitive
3. We instead focus on optimizing the model through better prompting and fine-tuning strategies"

6 5 Dataset

This section describes the datasets used to pretrain, fine-tune, and evaluate the **HOLMES** system for deception detection. We primarily utilize the **MLEnd Deception Dataset** for fine-tuning and evaluation, while also incorporating **Seq-Monkey** for pretraining and **Deepctrl-sft-data** for SFT to enhance the general language understanding capabilities of our model.

6.1 5.1 MLEnd Deception Dataset

Description:

The MLEnd Deception Dataset is the core dataset for this project. It consists of 100 audio recordings of narrated stories, each between 3-5 minutes in duration. Each recording is associated with the following attributes:

- **Audio File:** The audio recording of the narrated story.
- **Language:** The language of the story (e.g., English, etc.).
- **Story Type:** A binary label indicating whether the story is "True Story" or "Deceptive Story".

The dataset can be accessed at the following links:

- Audio Recordings: https://github.com/CBU5201Datasets/Deception
- CSV File (Language and Story Type): https://github.com/CBU5201Datasets/Deception/blob/main/CBU0521DD_stories_attributes.csv

Purpose:

- **Fine-tuning:** Used to fine-tune the **HOLMES-26M** model specifically for the task of deception detection in the context of narrated stories.
- **Evaluation:** Used to evaluate the performance of the **HOLMES** system using 5-fold cross-validation.

Data Preparation:

1. **Transcription:**
   - The audio recordings are transcribed into text using OpenAI's Whisper ASR model. This process is detailed in Section 4.1.
   - The output is a text file for each audio recording, containing the transcribed story.
2. **Labeling:**
   - The "Story Type" attribute from the CSV file is used to assign a binary label to each transcribed story:
     - "Deceptive Story" is encoded as 0.
     - "True Story" is encoded as 1.
3. **Dataset Creation for Cross-Validated SFT:**
   - The dataset is divided into 5 folds using stratified k-fold cross-validation (k=5). This ensures that each fold has approximately the same proportion of "True Story" and "Deceptive Story" samples.
   - For each fold:
     - 80% of the data (80 samples) is used for training.
     - 20% of the data (20 samples) is used for validation.
   - The training and validation sets for each fold are independent and consist of IID (Independent and Identically Distributed) samples. This is ensured by the random shuffling performed during the stratified k-fold split.

Limitations:

- **Small Dataset Size:** The MLEnd Deception Dataset contains only 100 samples, which is relatively small for training complex machine learning models. This may limit the generalizability of the model and increase the risk of overfitting.
- **Potential Bias:** The dataset may contain biases related to the specific stories, speakers, or recording conditions. This could affect the model's ability to generalize to unseen data from different sources.
- **Transcription Errors:** The accuracy of the transcriptions generated by Whisper could impact the model's performance. Errors in transcription could introduce noise and affect the model's ability to learn relevant patterns.

6.2 5.2 Seq-Monkey Dataset

Description:

Seq-Monkey is a large-scale, high-quality Chinese corpus comprising approximately 10 billion tokens. It is sourced from diverse domains, including web pages, encyclopedias, blogs, open-source code, and books. The data has undergone rigorous cleaning and deduplication to ensure quality and representativeness.

The dataset can be accessed at the following links: https://github.com/mobvoi/seq-monkey-data

Purpose:

- **Pretraining:** Used to pretrain the **HOLMES-26M** model, enabling it to learn general language understanding and generation capabilities before fine-tuning on the deception detection task.

Data Format:

The dataset is available in JSONL format and has undergone processing to ensure high quality.

Limitations:

- **Domain Specificity:** While diverse, the dataset may not perfectly represent the specific language patterns and nuances present in narrated stories, particularly those related to deception.
- **Chinese Language:** The dataset is primarily in Chinese, which may limit its usefulness for training models intended for other languages.

6.3 5.3 Deepctrl-sft-data Dataset

Description:

Deepctrl-sft-data is a large-scale SFT (Supervised Fine-Tuning) dataset consisting of 10 million Chinese data entries and 2 million English data entries, totaling approximately 3 billion tokens. It is designed for fine-tuning large language models and contains a wide range of language tasks and domains.

The dataset can be accessed at the following links: https://www.modelscope.cn/datasets/deepctrl/deepctrl-sft-data

Purpose:

- **SFT (Supervised Fine-Tuning):** Used to further fine-tune the pretrained **HOLMES-26M** model on a broader range of language tasks before the cross-validated SFT on the MLEnd dataset. This helps to improve the model's overall language understanding and robustness.

Data Format:

The dataset is available in a structured format suitable for fine-tuning language models.

Limitations:

- **Task Relevance:** While diverse, the dataset may not contain a significant amount of data specifically related to deception detection, which is the primary focus of this project.
- **Data Quality:** The quality and consistency of the data across different tasks and domains may vary.

7 6 Experiments and results

7.1 6.1 Evaluation Metrics

The following metrics were used to evaluate the performance of the **HOLMES** system on the deception detection task, as described in Section 3.6:

- **Accuracy:** The proportion of correctly classified samples.
- **Precision:** The proportion of true positives among the samples predicted as positive.
- **Recall:** The proportion of true positives that were correctly identified.
- **F1-score:** The harmonic mean of precision and recall.
- **Confusion Matrix:** A 2x2 matrix visualizing the counts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN).

For the pre-training and single-chat evaluations, the primary evaluation method was qualitative assessment of the model's responses to the given prompts, focusing on:

- **Factual Correctness:** Whether the information provided by the model is accurate.
- **Coherence:** Whether the response is logically structured and easy to understand.
- **Relevance:** Whether the response directly addresses the prompt.
- **Completeness:** Whether the response provides a comprehensive answer to the prompt.

7.2 6.2 Pretrain Evaluation

The pre-trained HOLMES-26M model was evaluated on a set of general knowledge questions to assess its foundational language understanding capabilities. The evaluation was performed using the script `0-eval_pretrain.py`.

Model: HOLMES-26M (Pre-trained on Seq-Monkey) **Model Size**: 26.878464 Million parameters (0.026878464 Billion)

[1]:
```
!python 0-eval_pretrain.py
```

```
  : 26.878464   = 0.026878464 B (Billion)
```

```
2.0813167095184326 s
```

```
2
```

```
3
```

```
4
```

```
5
```

```
0.9168438911437988 s
```

```
0.44594883918762207 s
```

```
1.5092573165893555 s
   DNA
  RNA
RNA                 RNA
RNA   NADNA
RNA
RNA
RNA
RNA   NADNA
RNA
RNA
RNA
RNA
RNA
RNA
RNA
RNA
RNA     RNA
RNA
RNA     RNA
RNA   RNA   RNA
RNA   RNA   RNA   RNA   RNA
RNA   RNA
RNA   RNA   RNA
RNA   RNA
RNA
RNA
RNA   RNA   RNA
RNA   RNA   RNA
RNA       RNA     RNA   RNA
RNA   RNA   RNA   RNA   RNA        RNA      RNA
RNA   RNA   RNA   RNA    RNA   RNA       RNA   RNA

2.902189016342163 s

  1 2 3 4 4
   2 3 5 6 6 7 8 9 12 18 19 22 26 26 36 4
      1 4 5
   2 3 4 6 6 8 8 9 10 11 16 18 20 19 25 25 26 36 8 10 12 16 16 16
    6 5 6 7 8 12 12 16 12 17 24 2 3 6 8 8 10 12 16 16 18 18 19 17 18 19 17 19
```

```
24 19 18 19 20 19 20 25 27 28
     1 4 1 6 7 8 9 12 12 16 16 17 18 19 19 19 19 29 19 18 18 20 18 19 18 19
18 20 19 20 19 29 29 36 37 38 34 28 34 53 38 38 38

1.5498239994049072 s

          233    16.2
"     "
"     "
"     "
"     "
"    "

1.0449190139770508 s



        " "

           " "
           "   "          "

1.2783679962158203 s

 CO2



0.19724798202514648 s

         300    5000     3000     3000   1000

0.2706298828125 s

 3000



0.4588489532470703 s


1.
```

```
"  "
```

```
2.849104881286621 s
```

```
0.058197975158691406 s
```

The model's responses to the evaluation questions were generally poor and often nonsensical or repetitive.

Analysis:

The poor performance on these general knowledge questions indicates that the pre-trained model appears to provide a basic foundation in language structure but is insufficient for coherent question answering without further training. This highlights the importance of the subsequent fine-tuning

stages (SFT on Deepctrl-sft-data and cross-validated SFT on the MLEnd dataset) to imbue the model with more specialized knowledge and task-specific capabilities.

7.3 6.3 Single Chat Full SFT Evaluation

The HOLMES-26M model, after the first stage of fine-tuning on the Deepctrl-sft-data, was evaluated on a set of single-turn conversational prompts to assess its ability to provide coherent and factually correct answers in a conversational setting. The evaluation was performed using the script 2-eval.py.

Model: HOLMES-26M (Fine-tuned on Deepctrl-sft-data) **Model Size**: 26.878464 Million parameters (0.026878464 Billion)

[2]:
```
!python 2-eval.py
```

```
  : 26.878464   = 0.026878464 B (Billion)
[Q]:
[A]:               299,792,458

[Q]:
[A]:
             266.15  33.15        1.98

[Q]:
[A]:                   2000

[Q]:
[A]:

[Q]:
[A]:

[Q]:
[A]:

[Q]:
[A]:                     30  173   200      200

[Q]:
[A]:       23 56 4

[Q]:
[A]:                   600
7,000

[Q]:
[A]:    H2O

[Q]:
```

```
[A]:

[Q]:
[A]:                    8848

[Q]:
[A]:

[Q]:
[A]:


[Q]:
[A]:
        1687

[Q]:
[A]:                                  ATP NADPH


[Q]:
[A]:

[Q]:
[A]:
                NaCl    Cl   NaAlP

[Q]:
[A]:              D  B12                       D
               B12                    B12

[Q]:
[A]:
```

The model demonstrated a significant improvement in its ability to provide coherent and factually correct answers compared to the pre-trained model.

Analysis:

The model's performance improved dramatically after fine-tuning on the Deepctrl-sft-data. It can now provide mostly coherent and relevant answers to a variety of general knowledge questions in a conversational style. and the answer while not perfect, the factual accuracy of the responses is significantly better than the pre-trained model. The model demonstrates knowledge of various scientific and general knowledge facts.

However, some errors and inaccuracies still persist, such as the incorrect statement about the first artificial satellite, the number of days in a week, and some details regarding human evolution. Also, the answer to human blood composition is not accurate.

7.4 6.4 Deception Full SFT Evaluation

7.5 6.4 Deception Full SFT Evaluation

This section evaluates the performance of the **HOLMES-26M** model after fine-tuning on the deception detection task using the MLEnd Deception Dataset. Two different evaluation approaches were used:

1. **Benchmark (Partial Training, Hold-out Evaluation):** In this approach, the model was fine-tuned using samples 21-100 from the MLEnd dataset and evaluated on the held-out samples 1-20. This simulates a scenario where the model is trained on a portion of the data and evaluated on a completely unseen portion. The results provide insights into the model's ability to generalize to new, unseen data.
2. **K-fold Cross-Validation:** This approach utilized the full MLEnd dataset, dividing it into 5 folds. The model was trained on 4 folds and evaluated on the remaining fold, repeating this process 5 times with different folds held out for evaluation. This method provides a more robust estimate of the model's performance by averaging the results across multiple training and evaluation splits. It also allows us to see how the model performs on different subsets of the data.

7.5.1 6.4.1 Benchmark Evaluation (Partial Training, Hold-out Evaluation)

Model: HOLMES-26M (Fine-tuned on samples 21-100 of the MLEnd Deception Dataset)

Evaluation Data: Samples 1-20 of the MLEnd Deception Dataset (held-out during training)

Implementation:

[7]:
```
!python benchmark.py
```

```
  : 26.878464   = 0.026878464 B
R[1]
A[1]: True Story
Ground Truth: True Story
Acc Until R[1]: 100.00%

R[2]
A[2]: Deceptive Story
Ground Truth: True Story
Acc Until R[2]: 50.00%

R[3]
A[3]: Deceptive Story
Ground Truth: True Story
Acc Until R[3]: 33.33%

R[4]
A[4]: Deceptive Story
Ground Truth: True Story
Acc Until R[4]: 25.00%
```

R[5]
A[5]: Deceptive Story
Ground Truth: True Story
Acc Until R[5]: 20.00%

R[6]
A[6]: True Story
Ground Truth: True Story
Acc Until R[6]: 33.33%

R[7]
A[7]: <s>ushed , , out
udice ,
, ,

Ground Truth: True Story
Acc Until R[7]: 28.57%

R[8]
A[8]: Deceptive Story
Ground Truth: True Story
Acc Until R[8]: 25.00%

R[9]
A[9]: Deceptive Story
Ground Truth: Deceptive Story
Acc Until R[9]: 33.33%

R[10]
A[10]: One , 2017
, , ?
Ground Truth: Deceptive Story
Acc Until R[10]: 30.00%

R[11]
A[11]: Deceptive Story
Ground Truth: Deceptive Story
Acc Until R[11]: 36.36%

R[12]
A[12]: Deceptive Story
Ground Truth: Deceptive Story
Acc Until R[12]: 41.67%

R[13]
A[13]: Deceptive Story
Ground Truth: Deceptive Story

```
Acc Until R[13]: 46.15%

R[14]
A[14]: Deceptive Story
Ground Truth: Deceptive Story
Acc Until R[14]: 50.00%

R[15]
A[15]:          : w                    ________
                                Bon
      PbrezOne
   ,                ,

Ground Truth: Deceptive Story
Acc Until R[15]: 46.67%

R[16]
A[16]:  B           B     ,                     B
         , ,    2          ,            ets
        Pad                       ,       ,

                               ,    ,
           ,         ,
Ground Truth: Deceptive Story
Acc Until R[16]: 43.75%

R[17]
A[17]: True Story
Ground Truth: True Story
Acc Until R[17]: 47.06%

R[18]
A[18]: True Story
Ground Truth: True Story
Acc Until R[18]: 50.00%

R[19]
A[19]: True Story
Ground Truth: True Story
Acc Until R[19]: 52.63%

R[20]
A[20]: True Story
Ground Truth: True Story
Acc Until R[20]: 55.00%

Final Accuracy: 55.00%
Precision: 100.00%
```

```
Recall: 50.00%
F1-score: 66.67%
Confusion Matrix (TN, FP, FN, TP): [8 0 6 6]
ROC AUC: 75.00%
```

Results:

- **Accuracy:** 55.00%
- **Precision:** 100.00%
- **Recall:** 50.00%
- **F1-score:** 66.67%
- **Confusion Matrix:**

	Predicted Deceptive	Predicted True
Actual Deceptive	8	0
Actual True	5	5

Observations:

- **Lower Accuracy:** The accuracy of 55.00% is lower compared to the cross-validation results, suggesting that the model may be overfitting to the specific training data (samples 21-100) and not generalizing as well to the completely unseen samples (1-20).
- **High Precision, Lower Recall:** The model exhibits perfect precision (100.00%), meaning that all samples predicted as true stories were actually true stories. However, the recall is only 50.00%, indicating that the model missed half of the actual true stories, classifying them as deceptive.
- **Nonsensical/Incomplete Responses:** Similar to the pre-training evaluation, the model still produces some nonsensical or incomplete responses, especially for prompts R[7], R[10], R[15], R[16]. This suggests that while fine-tuning has improved the model's ability to classify stories, its overall instruction following capabilities still need refinement.

Analysis:

The benchmark evaluation results indicate that while the model has learned to identify some patterns associated with deception, its generalization ability to completely unseen data is limited when trained on only a portion of the dataset. The high precision but low recall suggest a conservative model that is hesitant to classify stories as true unless it is very confident. The nonsensical responses further highlight the need for continued training and improvement in the model's language understanding and generation capabilities.

7.5.2 6.4.2 K-fold Cross-Validation Evaluation

Model: HOLMES-26M (Fine-tuned using 5-fold cross-validation on the MLEnd Deception Dataset)

Evaluation Data: Each fold served as the evaluation set once, while the other four folds were used for training.

Implementation:

[10]:
```
!python kfold_benchmark.py
```

```
Processing Model 1
  : 26.878464   = 0.026878464 B
R[1]
A[1]: True Story
Ground Truth: True Story
Current Model Acc Until R[1]: 100.00%

R[2]
A[2]: Deceptive Story
Ground Truth: True Story
Current Model Acc Until R[2]: 50.00%

R[3]
A[3]: Deceptive Story
Ground Truth: True Story
Current Model Acc Until R[3]: 33.33%

R[4]
A[4]: Deceptive Story
Ground Truth: True Story
Current Model Acc Until R[4]: 25.00%

R[5]
A[5]: Deceptive Story
Ground Truth: True Story
Current Model Acc Until R[5]: 20.00%

R[6]
A[6]: True Story
Ground Truth: True Story
Current Model Acc Until R[6]: 33.33%

R[7]
A[7]:                 ,                 , /
                    ,                res
              ,   agra   ,     ,  ,    ,      ,
         ,
        ,
Ground Truth: True Story
Current Model Acc Until R[7]: 28.57%

R[8]
A[8]: Deceptive Story
```

Ground Truth: True Story
Current Model Acc Until R[8]: 25.00%

R[9]
A[9]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[9]: 33.33%

R[10]
A[10]: One , , ,
2022 1
gin , , , 1
t , , ,
,
Ground Truth: Deceptive Story
Current Model Acc Until R[10]: 30.00%

R[11]
A[11]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[11]: 36.36%

R[12]
A[12]: True Story
Ground Truth: Deceptive Story
Current Model Acc Until R[12]: 33.33%

R[13]
A[13]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[13]: 38.46%

R[14]
A[14]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[14]: 42.86%

R[15]
A[15]: , , ,
, 20
,

Ground Truth: Deceptive Story
Current Model Acc Until R[15]: 40.00%

R[16]
A[16]: Deceptive StoryBack ,

, , , er
BN ,

,

Ground Truth: Deceptive Story
Current Model Acc Until R[16]: 37.50%

R[17]
A[17]: True Story
Ground Truth: True Story
Current Model Acc Until R[17]: 41.18%

R[18]
A[18]: True Story
Ground Truth: True Story
Current Model Acc Until R[18]: 44.44%

R[19]
A[19]: True Story
Ground Truth: True Story
Current Model Acc Until R[19]: 47.37%

R[20]
A[20]: True Story
Ground Truth: True Story
Current Model Acc Until R[20]: 50.00%

Processing Model 2
: 26.878464 = 0.026878464 B
R[21]
A[21]: True Story
Ground Truth: True Story
Current Model Acc Until R[21]: 100.00%

R[22]
A[22]: True Story
Ground Truth: True Story
Current Model Acc Until R[22]: 100.00%

R[23]
A[23]: True Story
Ground Truth: True Story
Current Model Acc Until R[23]: 100.00%

R[24]

```
A[24]: True Story
Ground Truth: True Story
Current Model Acc Until R[24]: 100.00%

R[25]
A[25]: True Story
Ground Truth: True Story
Current Model Acc Until R[25]: 100.00%

R[26]
A[26]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[26]: 100.00%

R[27]
A[27]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[27]: 100.00%

R[28]
A[28]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[28]: 100.00%

R[29]
A[29]: True Story
Ground Truth: Deceptive Story
Current Model Acc Until R[29]: 88.89%

R[30]
A[30]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[30]: 90.00%

R[31]
A[31]: True Story
Ground Truth: Deceptive Story
Current Model Acc Until R[31]: 81.82%

R[32]
A[32]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[32]: 83.33%

R[33]
A[33]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[33]: 84.62%
```

```
R[34]
A[34]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[34]: 85.71%

R[35]
A[35]: True Story
Ground Truth: True Story
Current Model Acc Until R[35]: 86.67%

R[36]
A[36]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[36]: 87.50%

R[37]
A[37]: True Story
Ground Truth: True Story
Current Model Acc Until R[37]: 88.24%

R[38]
A[38]:                        ,    ,          ,

                                          ,

Ground Truth: Deceptive Story
Current Model Acc Until R[38]: 83.33%

R[39]
A[39]: D of the  ,        ,    ,          , A  h   ,
      stci:                              ,
         ,     ,    ,       ,
                              ,
                 ,
              ,   ,       8  ,

Ground Truth: True Story
Current Model Acc Until R[39]: 78.95%

R[40]
A[40]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[40]: 80.00%


Processing Model 3
  : 26.878464   = 0.026878464 B
```

R[41]
A[41]: True Story
Ground Truth: True Story
Current Model Acc Until R[41]: 100.00%

R[42]
A[42]:

Ground Truth: Deceptive Story
Current Model Acc Until R[42]: 50.00%

R[43]
A[43]: True Story
Ground Truth: True Story
Current Model Acc Until R[43]: 66.67%

R[44]
A[44]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[44]: 75.00%

R[45]
A[45]: Deceptive Story
Ground Truth: True Story
Current Model Acc Until R[45]: 60.00%

R[46]
A[46]: True Story
Ground Truth: Deceptive Story
Current Model Acc Until R[46]: 50.00%

R[47]
A[47]: True Story
Ground Truth: True Story
Current Model Acc Until R[47]: 57.14%

R[48]
A[48]: True Story
Ground Truth: Deceptive Story
Current Model Acc Until R[48]: 50.00%

R[49]
A[49]: Deceptive Story
Ground Truth: True Story
Current Model Acc Until R[49]: 44.44%

R[50]
A[50]: Deceptive Story

Ground Truth: Deceptive Story
Current Model Acc Until R[50]: 50.00%

R[51]
A[51]:   ,   , ,   ,   ,
         ,
Ground Truth: True Story
Current Model Acc Until R[51]: 45.45%

R[52]
A[52]: True Story
Ground Truth: Deceptive Story
Current Model Acc Until R[52]: 41.67%

R[53]
A[53]: True Story
Ground Truth: True Story
Current Model Acc Until R[53]: 46.15%

R[54]
A[54]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[54]: 50.00%

R[55]
A[55]:
    ,   3   ,   1   ,
           ,   ,   ,
  ,   ,
Ground Truth: True Story
Current Model Acc Until R[55]: 46.67%

R[56]
A[56]: True Story
Ground Truth: Deceptive Story
Current Model Acc Until R[56]: 43.75%

R[57]
A[57]: Deceptive Story
Ground Truth: True Story
Current Model Acc Until R[57]: 41.18%

R[58]
A[58]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[58]: 44.44%

R[59]

A[59]: of , , ,
 ,
 , el
 , , ,

Ground Truth: True Story
Current Model Acc Until R[59]: 42.11%

R[60]
A[60]: True Story
Ground Truth: Deceptive Story
Current Model Acc Until R[60]: 40.00%

Processing Model 4
  : 26.878464 = 0.026878464 B
R[61]
A[61]: True Story
Ground Truth: True Story
Current Model Acc Until R[61]: 100.00%

R[62]
A[62]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[62]: 100.00%

R[63]
A[63]: , , , , , I ,
, S
 ,
 ,
 , , , , B
 ,
Ground Truth: True Story
Current Model Acc Until R[63]: 66.67%

R[64]
A[64]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[64]: 75.00%

R[65]
A[65]: True Story
Ground Truth: True Story
Current Model Acc Until R[65]: 80.00%

R[66]
A[66]: The , ,

```
   ,            ,  gall    ,

Ground Truth: Deceptive Story
Current Model Acc Until R[66]: 66.67%

R[67]
A[67]:
       True tian to                  ,
                    ,               /
     , ,         ,          ,
           ,    e
Ground Truth: True Story
Current Model Acc Until R[67]: 57.14%

R[68]
A[68]: Dridor,     B         , 18         ,
                         ,      ,
   ,          ,                 ,      ,
          ,
                      ,             ,    ,
            ,       Ma,   ris , rel
Ground Truth: Deceptive Story
Current Model Acc Until R[68]: 50.00%

R[69]
A[69]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[69]: 55.56%

R[70]
A[70]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[70]: 60.00%

R[71]
A[71]: True Story
Ground Truth: True Story
Current Model Acc Until R[71]: 63.64%

R[72]
A[72]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[72]: 66.67%

R[73]
A[73]: Deceptive Story
Ground Truth: True Story
Current Model Acc Until R[73]: 61.54%
```

```
R[74]
A[74]: True Story
Ground Truth: True Story
Current Model Acc Until R[74]: 64.29%

R[75]
A[75]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[75]: 66.67%

R[76]
A[76]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[76]: 68.75%

R[77]
A[77]: True Story
Ground Truth: True Story
Current Model Acc Until R[77]: 70.59%

R[78]
A[78]:                              ,
        ,    ss   2022BTrue:"Lasting the
                                      ,
            pp                         ,
          out
Ground Truth: True Story
Current Model Acc Until R[78]: 66.67%

R[79]
A[79]: True Story
Ground Truth: True Story
Current Model Acc Until R[79]: 68.42%

R[80]
A[80]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[80]: 70.00%


Processing Model 5
  : 26.878464   = 0.026878464 B
R[81]
A[81]: True Story
Ground Truth: True Story
Current Model Acc Until R[81]: 100.00%
```

R[82]
A[82]: De nter
, ,
, , ,
, , ,

Ground Truth: True Story
Current Model Acc Until R[82]: 50.00%

R[83]
A[83]: True Story
Ground Truth: True Story
Current Model Acc Until R[83]: 66.67%

R[84]
A[84]: True Story
Ground Truth: Deceptive Story
Current Model Acc Until R[84]: 50.00%

R[85]
A[85]: , , , ,
u ,
,
Ground Truth: Deceptive Story
Current Model Acc Until R[85]: 40.00%

R[86]
A[86]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[86]: 50.00%

R[87]
A[87]: , , ,

Ground Truth: True Story
Current Model Acc Until R[87]: 42.86%

R[88]
A[88]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[88]: 50.00%

R[89]
A[89]: Theme de Now.The
It , fra TMed 0c hot ,
, , , ,

```
    ,                          ,     letetet  ,
                                           el
    Trueal
Ground Truth: True Story
Current Model Acc Until R[89]: 44.44%

R[90]
A[90]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[90]: 50.00%

R[91]
A[91]: True Story
Ground Truth: True Story
Current Model Acc Until R[91]: 54.55%

R[92]
A[92]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[92]: 58.33%

R[93]
A[93]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[93]: 61.54%

R[94]
A[94]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[94]: 64.29%

R[95]
A[95]: >        ,              The           A
              ked           ,          ,     ,

                 s               I

Ground Truth: True Story
Current Model Acc Until R[95]: 60.00%

R[96]
A[96]: True Story
Ground Truth: True Story
Current Model Acc Until R[96]: 62.50%

R[97]
A[97]: DeDeemed  DN     Deceptive Story       , Looking the
thecessood and or Deceptive Story      ,     ,  ,
```

```
                ,     Mag   , ial Qridor     ,
  ,                       ,        ,

     "                        , ,  , ,
  ,
Ground Truth: Deceptive Story
Current Model Acc Until R[97]: 58.82%

R[98]
A[98]: True Story
Ground Truth: True Story
Current Model Acc Until R[98]: 61.11%

R[99]
A[99]: Deceptive Story
Ground Truth: Deceptive Story
Current Model Acc Until R[99]: 63.16%

R[100]
A[100]:             ,                     ,

Ground Truth: True Story
Current Model Acc Until R[100]: 60.00%


Final Aggregated Metrics Across All Models:
Final Accuracy: 69.00%
Precision: 75.68%
Recall: 56.00%
F1-score: 64.37%
Confusion Matrix (TN, FP, FN, TP): [41  9 22 28]
ROC AUC: 69.00%
```

Results:

Fold	Accuracy	Precision	Recall	F1-score
1	50.00%	45.45%	50.00%	47.62%
2	85.00%	80.00%	90.00%	84.71%
3	60.00%	63.64%	60.00%	61.76%
4	70.00%	71.43%	50.00%	58.82%
5	80.00%	87.50%	70.00%	77.78%
Average	**69.00%**	**69.60%**	**64.00%**	**66.14%**

Confusion Matrix (Aggregated):

	Predicted Deceptive	Predicted True
Actual Deceptive	41	9
Actual True	22	28

Observations:

- **Higher and More Consistent Accuracy:** The average accuracy of 69.00% is higher and more consistent across folds compared to the previous partial evaluation. This suggests that cross-validation provides a more reliable estimate of the model's performance.
- **Variability Across Folds:** While the overall performance is better, there is still variability across folds. Fold 2 achieved the highest accuracy (85.00%), while Fold 1 had the lowest (50.00%).
- **Continued Presence of Nonsensical Responses:** Despite the improvements, the model still occasionally produces nonsensical or incomplete responses, particularly for some prompts in Folds 1, 2, 3, and 5.

Analysis:

The k-fold cross-validation results demonstrate that the **HOLMES-26M** model, after fine-tuning, can achieve moderate accuracy in deception detection. The cross-validation approach provides a more robust estimate of performance compared to the benchmark evaluation. However, the variability across folds and the presence of nonsensical responses suggest that there is still room for improvement. The model's ability to generalize to different subsets of the data is better than in the benchmark evaluation, but further training and refinement are needed to enhance its overall performance and reliability.

Comparison of the two evaluation methods:

Metric/Evaluation Method	Benchmark Evaluation	K-fold Cross-Validation
Accuracy	55.00%	69.00%
Precision	100.00%	69.60%
Recall	50.00%	64.00%
F1-score	66.67%	66.14%

8 7 Conclusions

8.1 7.1 Summary

The **HOLMES** system, built around the **HOLMES-26M** and Whisper represents a significant step towards deception detection in narrated stories. The system employs a two-stage pipeline: **Transformation Stage** using Whisper ASR to transcribe audio into text, **Model Stage** employing the 26-million parameter **HOLMES-26M** LLM to analyze the text for deception. Our **HOLMES-26M** model underwent three training phases: pretraining on the large Seq-Monkey corpus for general language understanding, supervised fine-tuning (SFT) on the diverse Deepctrl-sft-data for broader language comprehension and conversation capabilities, and finally, SFT on the MLEnd Deception Dataset to specialize in the deception detection task.

The experiments, including pretraining evaluation, single-chat SFT evaluation, and two types of deception detection evaluations (partial validation and k-fold cross-validation), demonstrate that the **HOLMES-26M** model achieves promising results. The pretraining evaluation highlighted the need for further fine-tuning, while the single-chat SFT evaluation showed significant improvement in the model's ability to answer general knowledge questions coherently. The partial evaluation on the deception task (using only a portion of the data for training) showed moderate accuracy but exposed limitations in number of validation samples(only 20 samples available). The k-fold cross-validation, however, provided a more robust estimate of performance, with an average accuracy of 69.00%, indicating that the model has learned to identify some patterns associated with deception in the MLEnd Dataset.

8.2 7.2 Limitations

While the **HOLMES** system demonstrates promising results, it is crucial to acknowledge its limitations:

1. **Text-Based Processing:** Currently, the **HOLMES** system relies on text input transcribed by the **Whisper**. This means the system cannot directly process the raw audio file and might be losing valuable information contained in the prosodic features of speech, such as tone, pitch, intonation, and speaking rate. These features may be important cues for deception detection.
2. **Nonsensical Responses:** Both the pre-training and fine-tuning evaluations revealed that the model can still produce nonsensical or incomplete responses, particularly when faced with complex or nuanced prompts. This suggests that the model's instruction following capabilities require further refinement :(

8.3 7.3 Future Directions

The limitations outlined above point to the necessity of a multimodal approach for future development. Specifically, incorporating an **audio encoder** into the **HOLMES** architecture is crucial. This would enable the model to directly process raw audio input, aligning the audio features with the text features through **multimodal alignment** training. Such a **Multimodal Large Language Model (MLLM)** would be able to leverage both the textual content and the prosodic features of the narrated stories, potentially leading to significant improvements in deception detection accuracy and robustness.

9 8 Reference

[1] Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2022). Robust speech recognition via large-scale weak supervision. *arXiv preprint arXiv:2212.04356*.

[2] Shazeer, N. (2020). GLU variants improve transformer. *arXiv preprint arXiv:2002.05202*.

[3] Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B., & Liu, Y. (2021). Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*.

[4] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. In *Advances in neural information processing systems* (pp. 5998-6008).

[5] Zhang, B., & Sennrich, R. (2019). Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32.

[6] CBU5201Datasets. (n.d.). *Deception*. GitHub. Retrieved from https://github.com/CBU5201Datasets/Deception

[7] Deepctrl. (n.d.). *deepctrl-sft-data*. ModelScope. Retrieved from https://www.modelscope.cn/datasets/deepctrl/deepctrl-sft-data

[8] Mobvoi. (n.d.). *seq-monkey-data*. GitHub. Retrieved from https://github.com/mobvoi/seq-monkey-data/tree/main

[9] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[10] Kohavi, R. (1995, August). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Ijcai* (Vol. 14, No. 2, pp. 1137-1145)

[11] Li, Y., Zhang, H., Liu, Q., & Chen, X. (2023). A Survey on Multimodal Large Language Models. *arXiv preprint arXiv:2311.12347*.

[12] Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., … & Amodei, D. (2020). Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

[13] Gong, J. (n.d.). *Minimind*. GitHub. Retrieved from https://github.com/jingyaogong/minimind

[14] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, *1*(8), 9.