Methodology Description

I. Data utilization

The competition organizer provided two datasets on HuggingFace:

- darrow-ai/LegalLensNLI
- darrow-ai/LegalLensNLI-SharedTask

Both datasets contain only a training split with 312 samples. Through converting to lowercase, removing punctuation and extra spaces, we found that about 160 samples are different between the two. However, in many of them, the names of the companies in LegalLensNLI premises and hypotheses are changed to DEFENDANT in LegalLensNLI-SharedTask.

We appended the 160 different examples from the second dataset to the first dataset and splitted with test size 0.4 into train_raw (283 examples) and validation set (189 examples), after that we appended the augmented versions of examples in train_raw that belonged to the first dataset (382 examples).

- Train: 665 examples (full_train.csv)
- Validation: 189 examples (test.csv)

II. Data Augmentation

We augmented the data using Langchain and GPT-4o-mini (via API) by paraphrasing both the hypothesis and premises into two different versions to simulate English language proficiency at two IELTS levels: 6.5 and 8.5.

Data Augmentation Process

1. Expand original dataset

We added 3 new columns for convenience

- Id: only original examples are given id, augmented versions will have id = ‘None'
- from_id: id of original version of augmented examples, from_id of original examples will be ‘None'
- augment_method: gpt4o_6.5 for 6.5 IELTS version and gpt4o_8.5 for 8.5 IELTS version

2. Pydantic Model for data validation

```
class Paraphrase(BaseModel):
    hypo_85_paraphrase: str = Field(description="The 8.5 IELTS version 
paraphrase of the hypothesis")
```

```
    pre_85_paraphrase: str = Field(description="The 8.5 IELTS version paraphrase of the premise")
    hypo_65_paraphrase: str = Field(description = "The 6.5 IELTS version paraphrase of the hypothesis")
    pre_65_paraphrase: str = Field(description = "The 6.5 IELTS version paraphrase of the premise")
```

3. Prompt

" I am doing a Natural Language Inference task and i need you to help me augment my training data for a richer dataset.

Here is the hypothesis {hypothesis} and here is the premise {premise}

Given a legal pair of hypothesis and premise. I need you to paraphrase them, both the hypothesis and premise each has 2 versions.

One version is as if you have the English level of a person with IELTS 8.5.

One version is as if you have the English level of a person with IELTS 6.5.

Please read and paraphrase carefully so that it does not lose meaning.

{format_instructions}"

4. Model

We used gpt-4o-mini model via API with corresponding settings:

- temperature: 0
- timeout = None,
- streaming = False
- verbose = True
- model_kwargs = {"seed": 42}

5. Paraphraser Class

- Purpose: paraphrase hypothesises and premises into 2 version
- Functions:

  __init__ (): initialize ChatOpenAI model
  _get_chain(): get chain and pass format instruction from PydanticOutputParser to prompt
  paraphrase(hypothesis: str, premise: str) -> Paraphrase: paraphrase hypothesis and premise and return output structure.

The full code for data augmentation step can be found in the 'data-augmentation.ipynb' notebook.

III. Training model

1. Create dataset:

We created the `LegalLensDataset` class to handle data processing for our model. This class inherits from PyTorch's `Dataset` class and implements the following key methods:

- `__init__`: Initializes the dataset with the data, tokenizer, maximum sequence length, and number of labels.
- `__getitem__`: Processes individual items, encoding the premise and hypothesis pairs, and returning the input tensors and labels.
- `__len__`: Returns the total number of samples in the dataset.

In addition, the class handles label encoding, where 'Contradict' is mapped to 0, 'Entailed' to 1, and 'Neutral' to 2.

For the evaluation dataset, as said earlier, we selected 40% of the raw data (combination of `darrow-ai/LegalLensNLI` and `darrow-ai/LegalLensNLI-SharedTask`). The remaining 60% of the data is combined with augmented data to create the training set. Overall, the dataset contains 665 training examples and 189 examples for the validation set.

2. Model Selection:

We conducted a comprehensive evaluation of several state-of-the-art pre-trained models to determine the most suitable architecture for our Legal NLI task. The models under consideration were:

1. LegalBERT *(nlpaueb/legal-bert-base-uncased)*
2. T5 *(google-t5/t5-base)*
3. DeBERTa *(MoritzLaurer/DeBERTa-v3-large-mnli-fever-anli-ling-wanli)*

The DeBERTa variant we selected was specifically fine-tuned on an extensive collection of NLI datasets, achieving state-of-the-art performance on NLI tasks on the Hugging Face hub at the time of its release (06.06.22).

Our rigorous testing process revealed that DeBERTa consistently delivered the most stable results across multiple training iterations, with F1-macro scores consistently ranging between 0.90 and 0.92 on our evaluation set. This stability was particularly crucial given our relatively small dataset and the paramount importance of consistent performance in legal NLI applications.

While LegalBERT occasionally yielded promising results, the consistent performance of DeBERTa across various runs made it the preferred choice for our task. The stability of

DeBERTa's performance provides a reliable foundation for legal text analysis, where consistency is essential for practical applications.

3. Training Process:

We used the Hugging Face Transformers library for model training. Here are the key components of our training process:

a) **Metrics**: We implemented a custom `compute_metrics` function to calculate F1-score (macro), precision, and recall during evaluation.

b) **Training Arguments**: We set up the training arguments using the `TrainingArguments` class from Transformers:

- Output directory: './results'
- Evaluation strategy: Per epoch
- Save strategy: Per epoch
- Learning rate: 5e-06
- Batch size: 1 (with gradient accumulation steps of 2)
- Number of epochs: 10
- Warmup ratio: 0.06
- Weight decay: 0.01
- Best model selection: Based on highest F1-macro score

d) **Trainer**: We used the Transformers `Trainer` class to manage the training process, providing it with our model, training arguments, datasets, and metric computation function.

4. **Training Results**: After training for 10 epochs, we found that the best checkpoint was consistently achieved at epoch 10. This checkpoint yielded an F1-macro score of 0.925 on our evaluation set, demonstrating strong performance on the Legal NLI task.
5. **Model Stability**: It's worth noting that while we experimented with LegalBERT and T5 models, DeBERTa showed the most stable performance across multiple training runs. This stability is crucial for ensuring reliable results in legal applications where consistency is paramount.

In conclusion, our training process, leveraging the DeBERTa model with carefully tuned hyperparameters and data augmentation techniques, resulted in a robust model for Legal Natural Language Inference. The highest F1-macro score of 0.925 on the evaluation set indicates strong performance, making this model suitable for this Legallens task.

4. Results

This is the evaluation metrics for the best Deberta’s checkpoint that we achieved:

Model name	Best epoch	F1-macro	Precision	Recall
MoritzLaurer/DeBERTa-v3-large-mnli-fever-anli-ling-wanli	9/10	0.925	0.926	0.928