Department of Computer Science and Information Engineering,
National Taitung University, Taiwan

LLM workshop: Fact Check

27 Nov, 2024

Github

Overview

What is Large Language Models

A large language model (LLM) is a type of computational model designed for natural language processing tasks such as language generation.

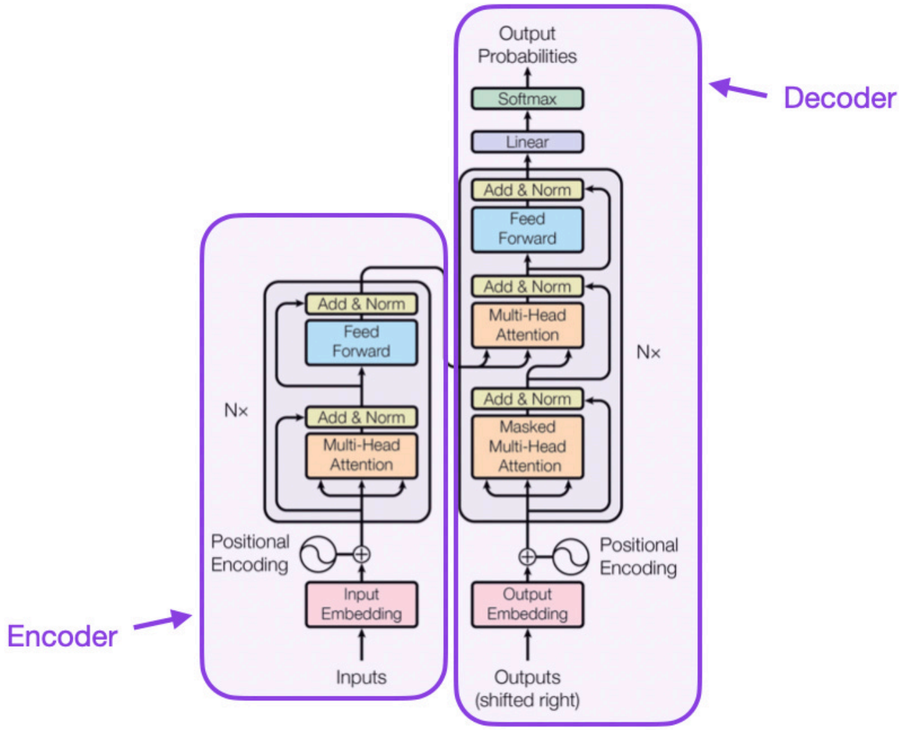
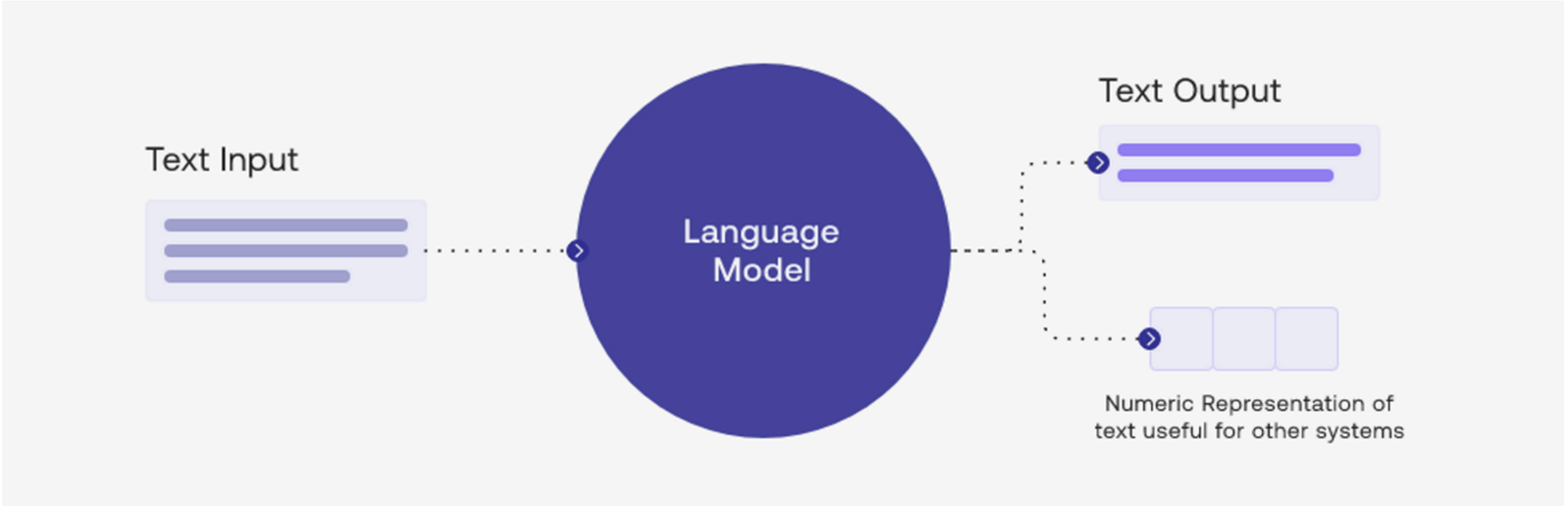
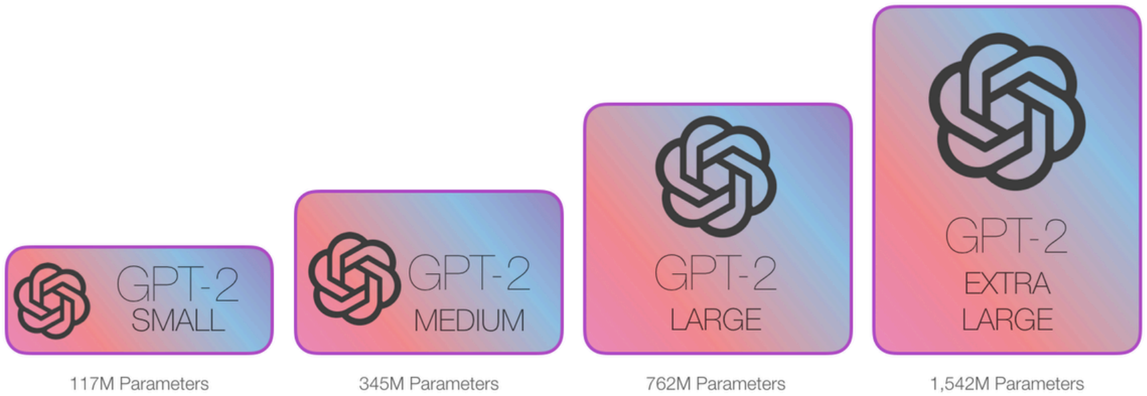


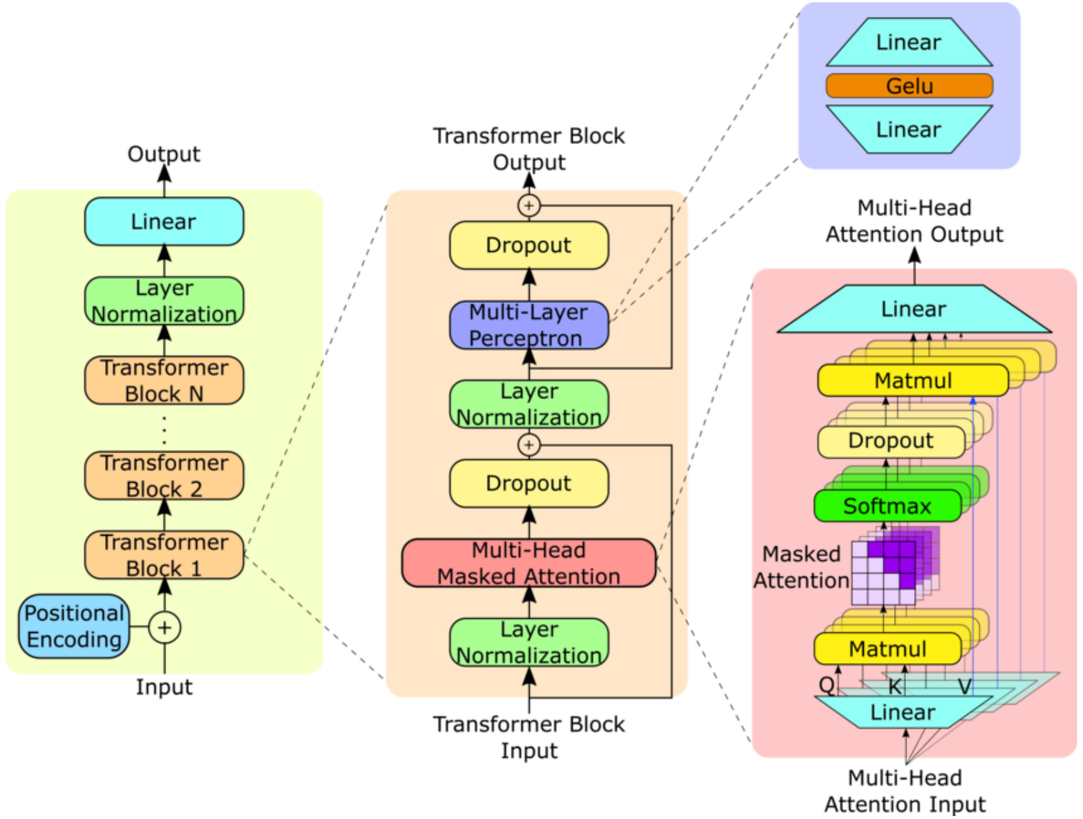
Figure 1: The Transformer - model architecture.

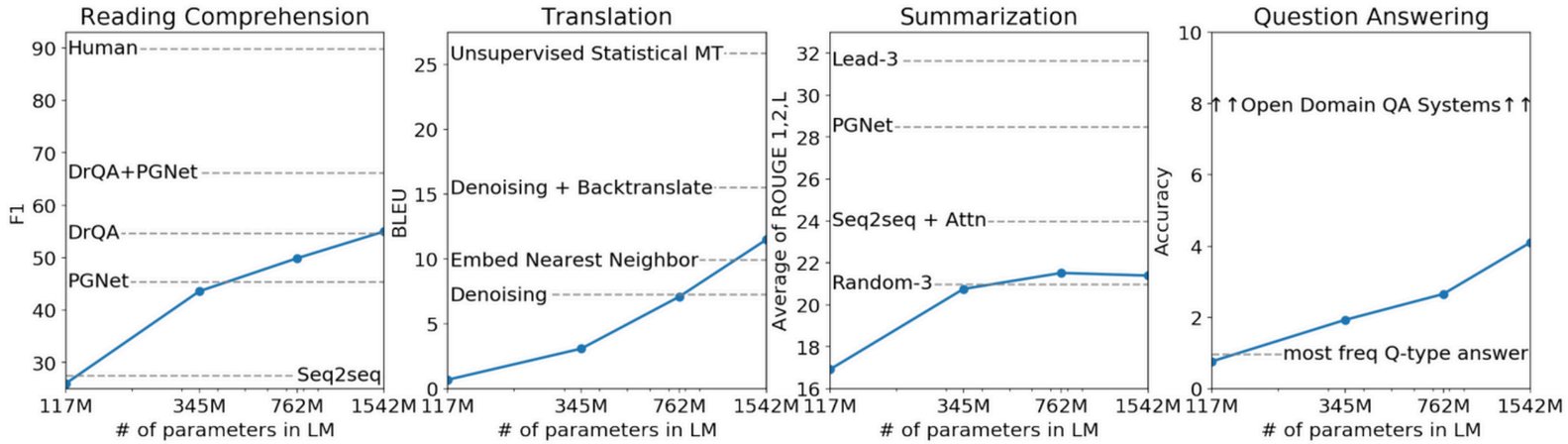


Introduction

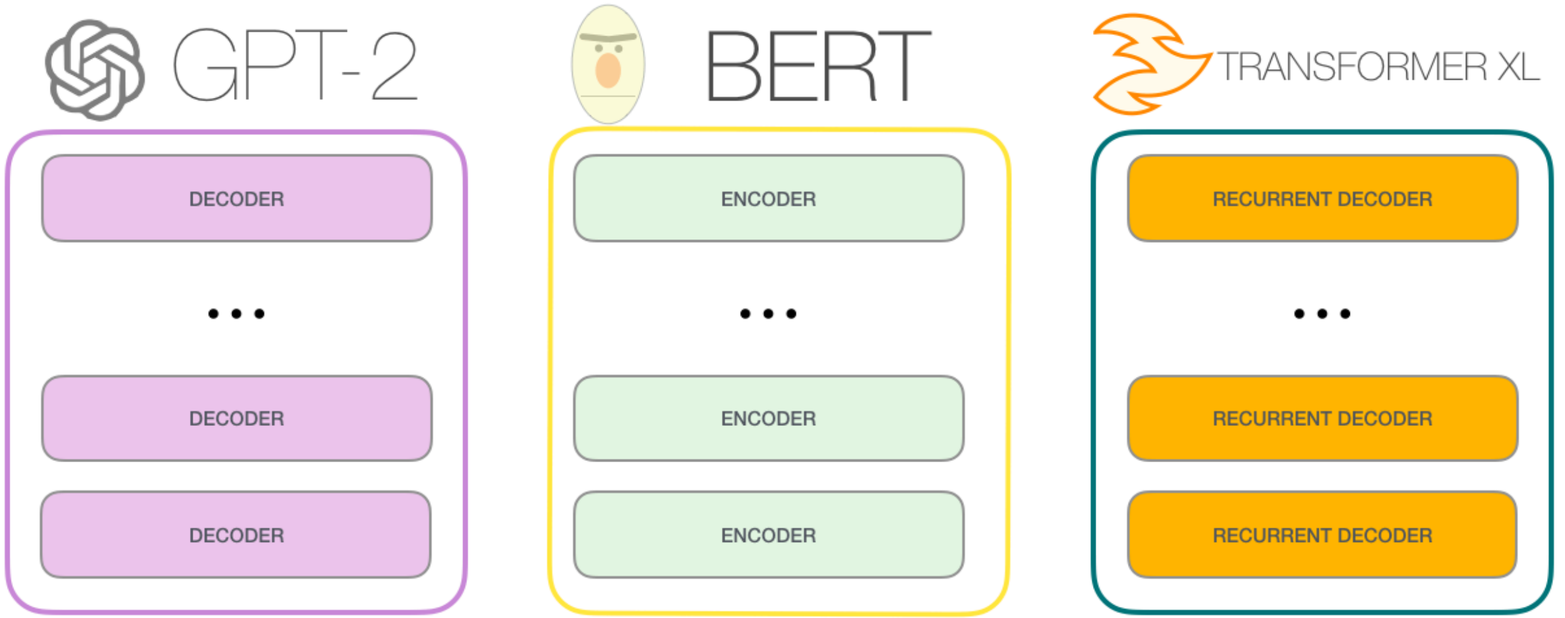
GPT-2





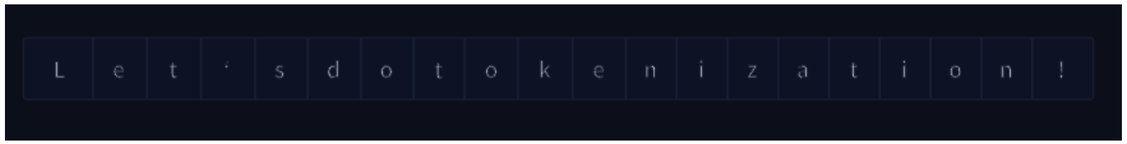


Overview of mainstream LLM architecture

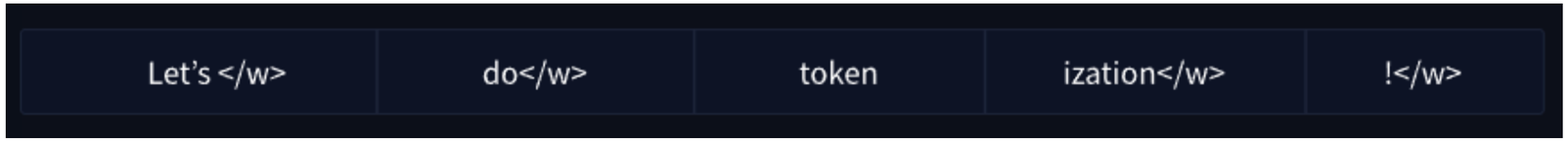


Tokenization

Word-based



Character-based



subword tokenization

The Limitations of LLMs

- Hallucinations
- Limited knowledge
- Difficulty with certain linguistic elements
- Bias and stereotyping

GPT-2 response

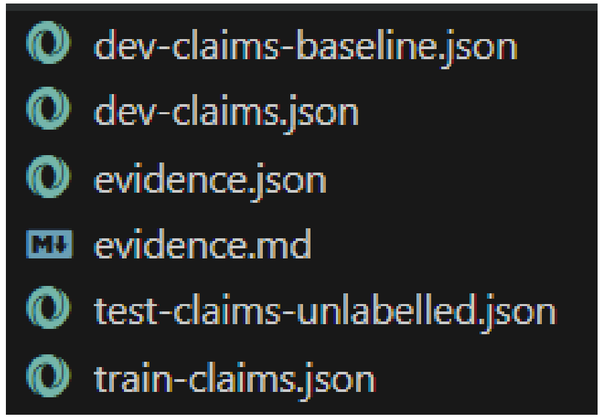
```
What is AI?

AI is a new field of research that seeks to understand how our brains work and how we interact with the world around us. The goal of AI research is to better understand the h
uman brain and to develop new ways to use it to solve problems. In this article, we will explore how AI can help us understand what it means to be human, and what we can lear
n from it. We will also look at the challenges AI poses to our everyday lives, including how it can be used to improve our health and well-being.
```

Dataset-overview

Claim: The Earth's climate sensitivity is so low that a doubling of atmospheric CO2 will result in a surface temperature change on the order of 1°C or less.

Evidence:

1. In his first paper on the matter, he estimated that global temperature would rise by around 5 to 6 °C (9.0 to 10.8 °F) if the quantity of CO 2 was doubled.
2. The 1990 IPCC First Assessment Report estimated that equilibrium climate sensitivity to a doubling of CO2 lay between 1.5 and 4.5 °C (2.7 and 8.1 °F), with a "best guess in the light of current knowledge" of 2.5 °C (4.5 °F).



- [train-claims,dev-claims].json: JSON files for the labelled training and development set;
- [test-claims-unlabelled].json: JSON file for the unlabelled test set;
- evidence.json: JSON file containing a large number of evidence passages (i.e. the "knowledge source");
- dev-claims-baseline.json: JSON file containing predictions of a baseline system on the development set;
- eval.py: Python script to evaluate system performance (see "Evaluation" below for more details).

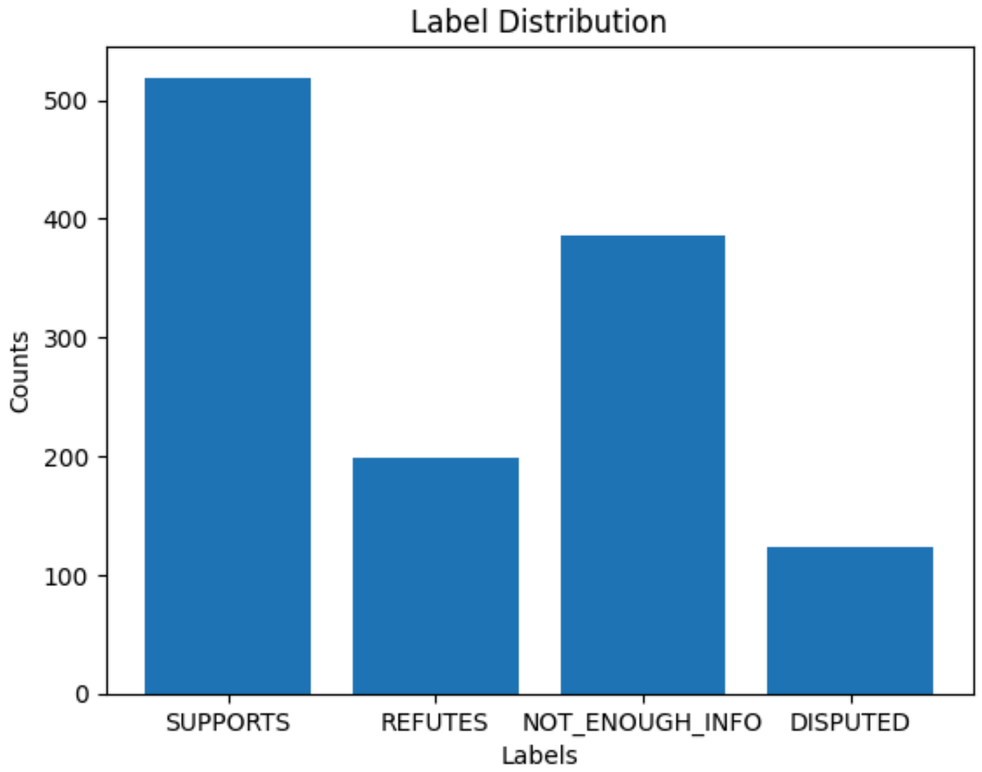
```
train_set['claim-1937']
✓ 0.0s
{'claim_text': 'Not only is there no scientific evidence that CO2 is a pollutant, higher CO2 concentrations actually help ecosystems support more plant and animal life.',
 'claim_label': 'DISPUTED',
 'evidences': ['evidence-442946', 'evidence-1194317', 'evidence-12171']}
```

Dataset-train

```
train_set['claim-1937']
```

✓ 0.0s

```
{'claim_text': 'Not only is there no scientific evidence that CO2 is a pollutant, higher CO2 concentrations actually help ecosystems support more plant and animal life.',
 'claim_label': 'DISPUTED',
 'evidences': ['evidence-442946', 'evidence-1194317', 'evidence-12171']}
```




```
100%|██████████| 1228/1228 [00:00<00:00, 103811.45it/s]
Supports: 519 Refutes: 199 Not enough info: 386 Disputed: 124
```

Total number of claims: 1228

Dataset-evidence

```
train_set['claim-1937']
```

✓ 0.0s

```
{'claim_text': 'Not only is there no scientific evidence that CO2 is a pollutant, higher CO2 concentrations actually help ecosystems support more plant and animal life.',
 'claim_label': 'DISPUTED',
 'evidences': ['evidence-442946', 'evidence-1194317', 'evidence-12171']}
```

```
evidence_set = json.load(open('./data/evidence.json'))
len(evidence_set), type(evidence_set)
```

✓ 0.6s

```
(1208827, dict)
```

```
list(evidence_set.keys())[:5]
```

✓ 0.0s

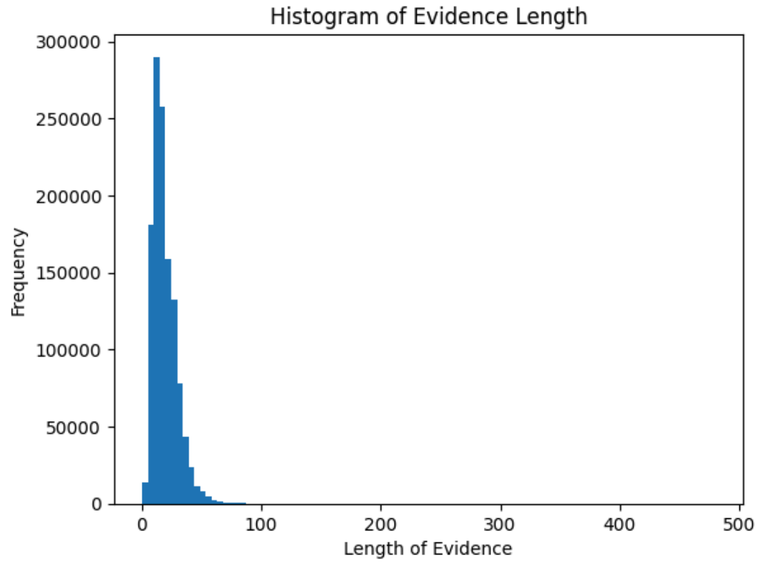
```
['evidence-0', 'evidence-1', 'evidence-2', 'evidence-3', 'evidence-4']
```

```
evidence_set['evidence-0']
```

✓ 0.0s

```
'John Bennet Lawes, English entrepreneur and agricultural scientist'
```

```
Max length of evidence: 479
Index of Max length of evidence: 358371
Total number of evidence: 1208827
```



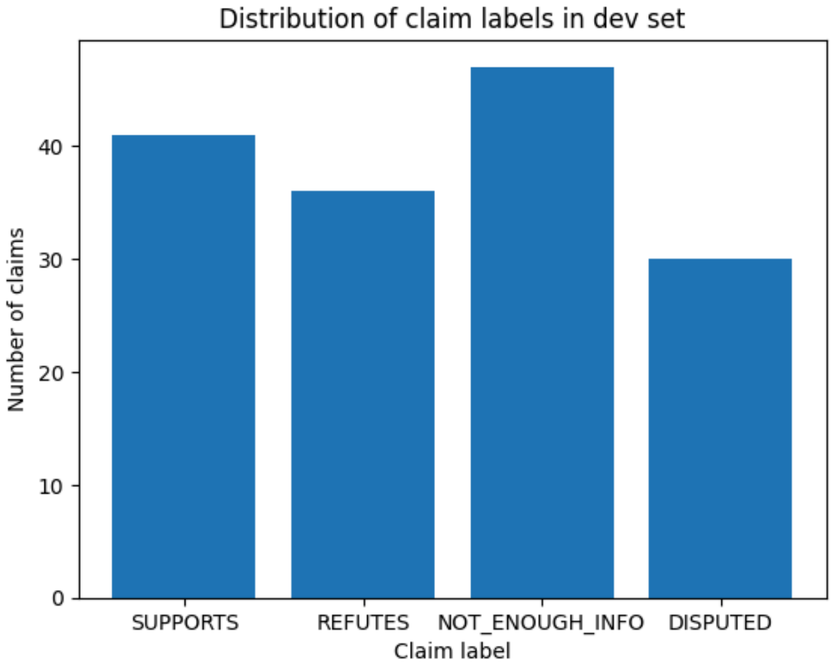
Dataset-dev

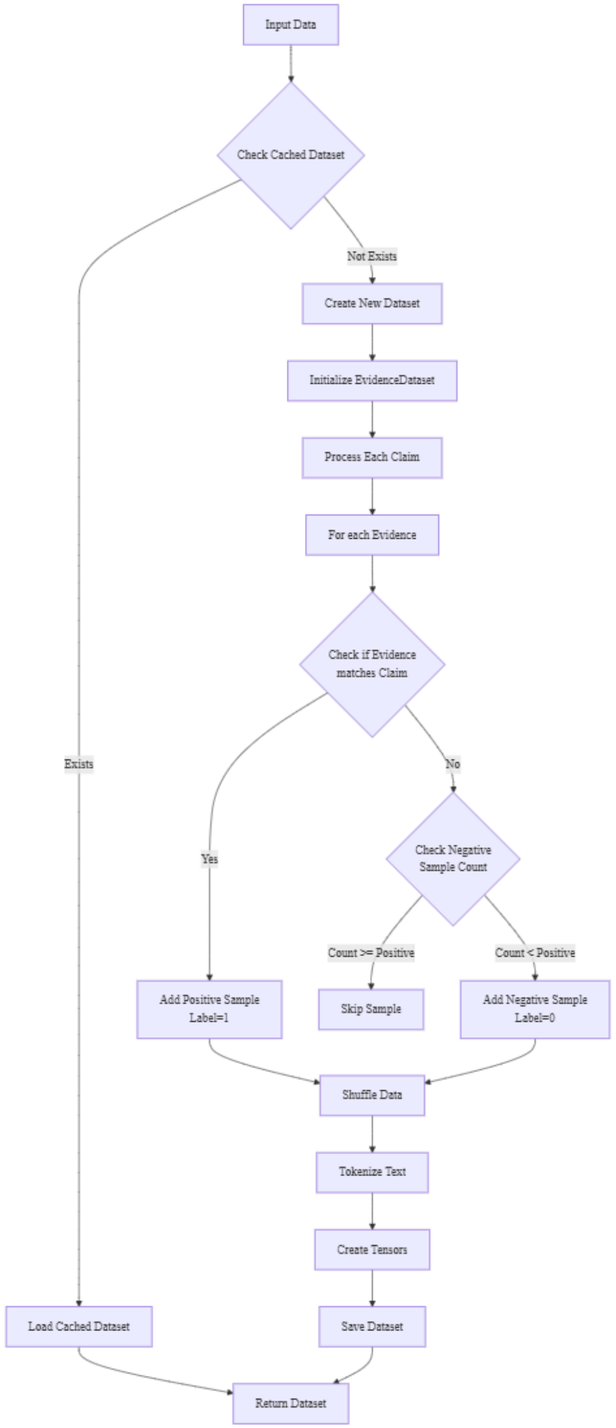
```
dev_set['claim-752']
```

[3] ✓ 0.0s

```
{'claim_text': '[South Australia] has the most expensive electricity in the world.',
 'claim_label': 'NOT_ENOUGH_INFO',
 'evidences': ['evidence-67732',
  'evidence-572512',
  'evidence-909871',
  'evidence-596058',
  'evidence-66394',
  'evidence-212071']}
```

```
Max number of sentences in a claim: 65
Total number of dev claims: 154
```

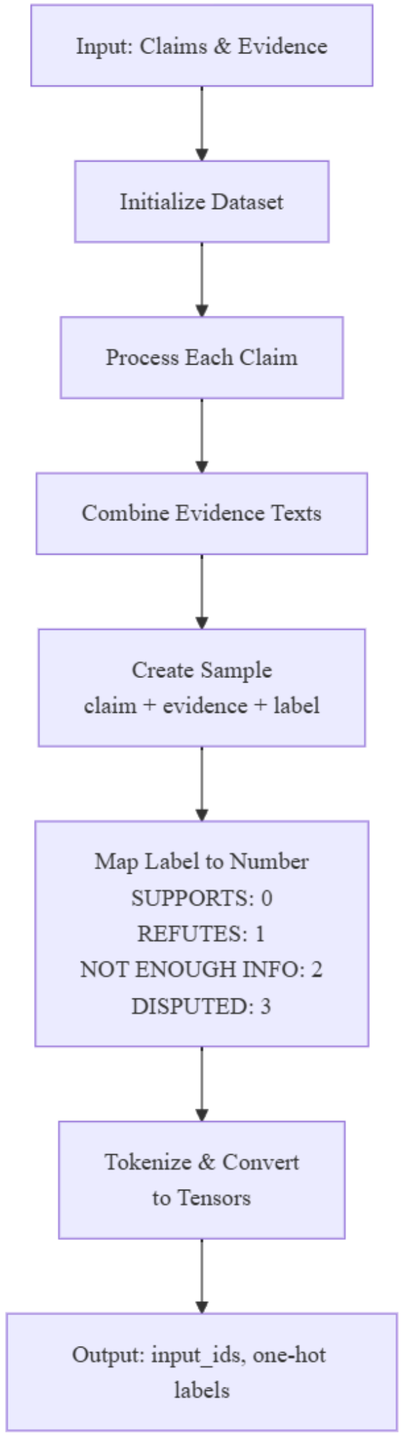




Task I

1. Using Model to find the most relevant evidence to the claim.

```python
class RelevantModel(torch.nn.Module):
    def __init__(self):
        super().__init__()
        self.claim_encoder = torch.nn.TransformerEncoderLayer(d_model=512, nhead=8)
        self.evidence_encoder = torch.nn.TransformerEncoderLayer(d_model=512, nhead=8)
        self.linear = torch.nn.Linear(1024, 1)

    def forward(self, claim_input_ids, evidence_input_ids):
        claim_output = self.claim_encoder(claim_input_ids)
        evidence_output = self.evidence_encoder(evidence_input_ids)
        out_score = self.linear(torch.cat((claim_output, evidence_output), dim=1))
        return out_score
```



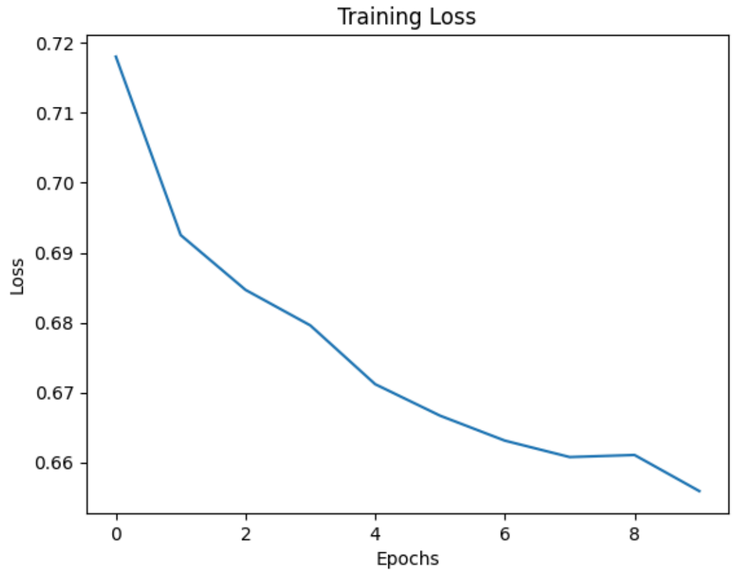
Task II

2. Using Language Model to classify the claim_label.

```python
class SmallLanguageModel(torch.nn.Module):
    def __init__(self):
        super().__init__()
        self.bert = GPT2Model.from_pretrained('gpt2')
        self.linear = torch.nn.Linear(768, 4)

    def forward(self, input_ids):
        outputs = self.bert(input_ids)
        return self.linear(outputs.last_hidden_state[:, 0, :])
```

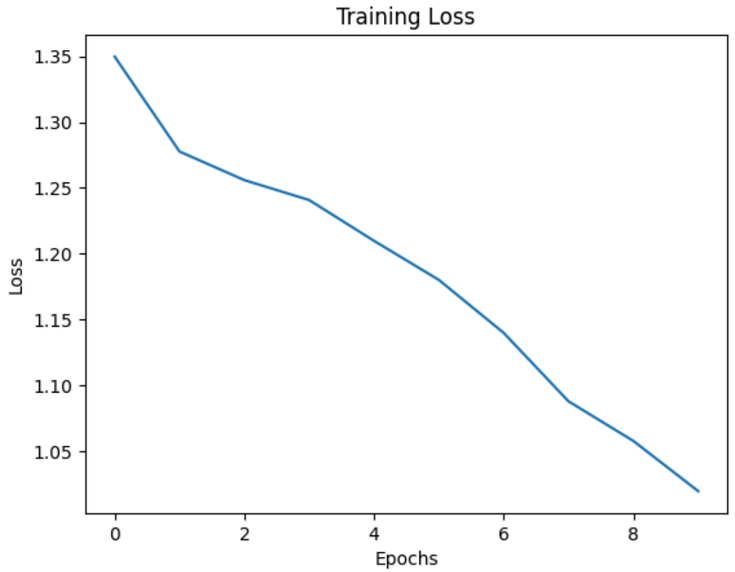
Result & Discussion

```
[[2359 1763]
 [1344 2778]]
              precision    recall  f1-score   support

           0       0.64      0.57      0.60      4122
           1       0.61      0.67      0.64      4122

    accuracy                           0.62      8244
   macro avg       0.62      0.62      0.62      8244
weighted avg       0.62      0.62      0.62      8244
```



Task I

```
[[140  26 260  93]
 [ 51   0 101  47]
 [201  55  88  42]
 [ 34   7  78   5]]
              precision    recall  f1-score   support

           0       0.33      0.27      0.30       519
           1       0.00      0.00      0.00       199
           2       0.17      0.23      0.19       386
           3       0.03      0.04      0.03       124

    accuracy                           0.19      1228
   macro avg       0.13      0.13      0.13      1228
weighted avg       0.19      0.19      0.19      1228
```



Task II

Conclusion

```
sec513@sec513-Q670M-D3H-DDR4:~/Desktop/code/COMP90042_2024$ python3 eval.py --predictions ./data/dev-claims-baseline.json --groundtruth ./data/dev-claims.json
Evidence Retrieval F-score (F)    = 0.3377705627705628
Claim Classification Accuracy (A) = 0.35064935064935066
Harmonic Mean of F and A          = 0.3440894901357093
sec513@sec513-Q670M-D3H-DDR4:~/Desktop/code/COMP90042_2024$ python3 eval.py --predictions ./data/dev-claims-baseline_my.json --groundtruth ./data/dev-claims.json
Evidence Retrieval F-score (F)    = 0.3377705627705628
Claim Classification Accuracy (A) = 0.3246753246753247
Harmonic Mean of F and A          = 0.3310935103125922
sec513@sec513-Q670M-D3H-DDR4:~/Desktop/code/COMP90042_2024$
```

- Others LLM
- Different method
- Longer EPOCHs

Department of Computer Science and Engineering,
National Taitung University, Taiwan

Thank You

27 Nov, 2024