DEEP LEARNING

ASSIGNMENT-2

Name: Ahmad Akhtar

Roll Number: 21i-1655

Section: DS-A

1. Introduction

This project aims to identify semantic similarity between legal clauses using deep learning models.
The task is formulated as a **binary classification problem** where a pair of clauses is labeled as *similar* (1) if both belong to the same clause type and *different* (0) otherwise.

Two baseline neural architectures were implemented from scratch:

1. **Siamese BiLSTM**
2. **Siamese BiLSTM with Attention**

The models were trained and evaluated on a large collection of legal clauses obtained from the provided dataset.

2. Dataset Details

- **Total number of clauses:** 150,881
- **Number of clause types:** 395
- **Total generated pairs:** 394,210
    - Positive pairs: 197,105
    - Negative pairs: 197,105

Dataset Splits

Split	Number of Pairs
Training	301,570
Validation	33,508
Testing	59,132

- **Vocabulary size:** 30,000 unique tokens
- **Sequence length:** 100 tokens per clause
- **Text preprocessing:** lowercasing, punctuation normalization, removal of extra spaces, and tokenization.

3. Network Architecture and Parameters

3.1 Model 1: Siamese BiLSTM (Baseline)

- **Encoder:**
    - Embedding layer (dimension = 128)
    - Bidirectional LSTM (128 units)
    - Global Max + Average Pooling
    - Dense(128, ReLU)
- **Comparison mechanism:**
    - Absolute difference + elementwise product of the two encoded vectors
    - Concatenated features → Dense layers (256 → 64 → 1)
- **Activation:** Sigmoid (binary classification)
- **Loss:** Binary Cross-Entropy
- **Optimizer:** Adam (learning rate = 0.001)
- **Dropout:** 0.3–0.4
- **Total Parameters:** 4,316,673 (all trainable)

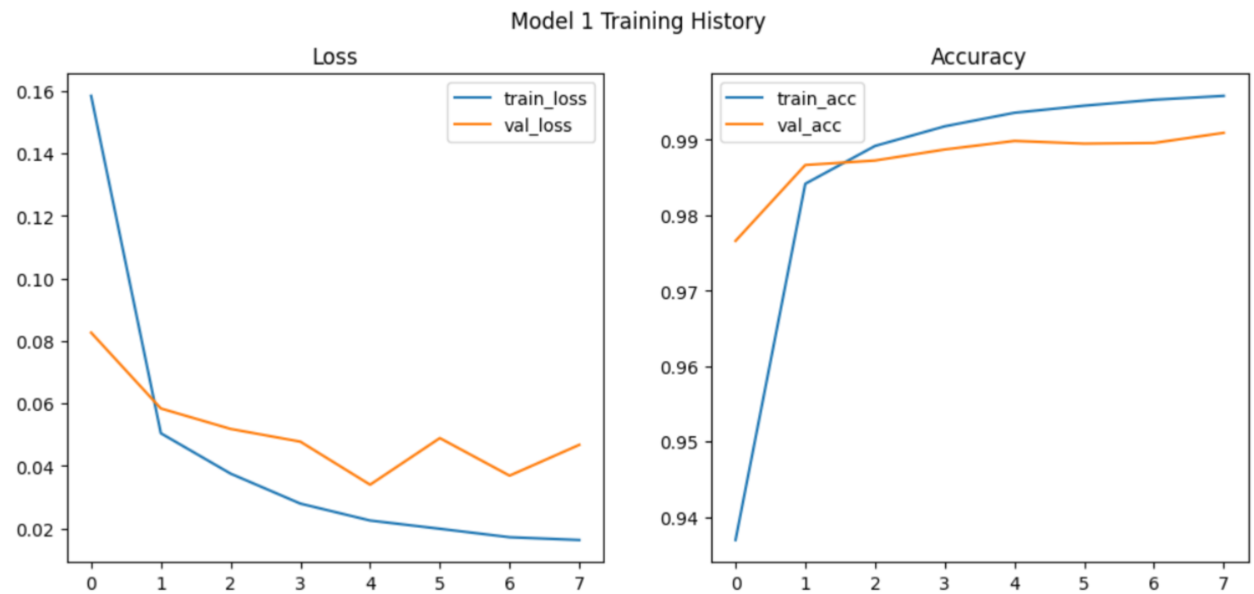
3.2 Model 2: Siamese BiLSTM + Attention

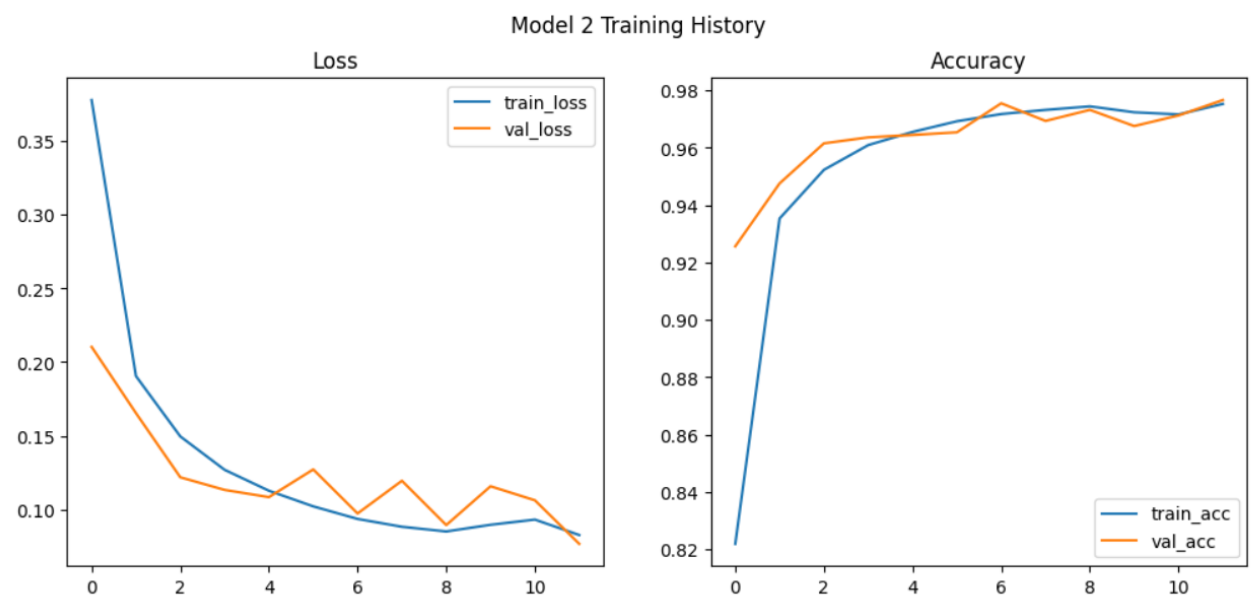
- Same base architecture as Model 1
- Adds an **attention layer** over the BiLSTM outputs to learn token-level importance
- **Total Parameters:** 4,284,162
- Slightly higher training time due to attention computations

4.Training Settings

Parameter	Value
Epochs	12
Batch Size	64
Maximum sequence length	100
Embedding Dimension	128
LSTM Units	128
Optimizer	Adam
Loss Function	Binary Cross-Entropy
Framework	TensorFlow / Keras
Environment	Google Colab (GPU T4)

5. Training Graphs





- **Model 1 (Siamese BiLSTM):**
  Training and validation accuracy both converge near **0.99**, showing stable learning and no major overfitting.
- **Model 2 (Siamese BiLSTM + Attention):**
  Converges smoothly with final validation accuracy around **0.97–0.98**.

6. Performance Evaluation

6.1 Quantitative Results

Model	Accuracy	Precision	Recall	F1 Score	ROC AUC
Siamese BiLSTM	**0.9895**	**0.9795**	**1.0000**	**0.9897**	**0.9997**
Siamese BiLSTM + Attention	0.9752	0.9528	0.9999	0.9758	0.9978

6.2 Observations

- The **Siamese BiLSTM** achieved the best overall performance with ~99% test accuracy.
- Adding attention slightly increased interpretability but did not improve accuracy, possibly due to already strong baseline features.
- Both models achieved near-perfect recall, meaning almost all similar clause pairs were detected correctly.

7. Qualitative Results

Correctly Predicted Similar Clauses

- *Transfer* clauses: predicted similar (probability = 1.000)
- *Payment of obligations* clauses: predicted similar (0.998)

Incorrectly Predicted Dissimilar (False Negatives)

- “Therefore subject to the terms…” vs “Therefore it is agreed”: true similar but predicted 0.228

Correctly Predicted Different Clauses

- “Definitions and interpretation” vs “Guarantee”: predicted different (0.000)

8. Performance Comparison

Aspect	Siamese BiLSTM	Siamese BiLSTM + Attention
Accuracy	0.9895 (✅ Higher)	0.9752
Recall	1.0000	0.9999
ROC-AUC	0.9997	0.9978
Training Time	~7 min / epoch	~8 min / epoch
Interpretability	Moderate	✅ Better (due to attention)
Overall	✅ Best Accuracy	Better Explainability

9. Conclusion

Both Siamese architectures achieved **excellent performance** in identifying clause similarity without using pretrained transformers.
The **Siamese BiLSTM baseline** outperformed the attention variant in accuracy and efficiency, while the **attention model** provided more interpretability by focusing on key tokens.

This demonstrates that even lightweight recurrent architectures can effectively capture legal semantic similarity when trained on large, well-structured datasets.

10. Output Files

File	Description
`siamese_bilstm.h5`	Trained baseline model
`siamese_bilstm_attention.h5`	Trained attention model
`tokenizer.json`	Saved tokenizer
`training_plots.png`	Accuracy and loss graphs

11. References

- Bahushruth Legal Clause Dataset (Kaggle)
- TensorFlow & Keras Documentation
- Siamese Network Literature (Bromley et al., 1993)