

A2-CS452: Legal Clause Similarity Detection

Student Name: Nabeeha Fazail

Fast ID: i211761

1. Objective

The goal of this project is to develop and evaluate NLP models capable of identifying semantic similarity between legal clauses.

Given two clauses, the task is to determine whether they express the same or related legal meaning, even if phrased differently.

This problem is crucial for contract review, case law retrieval, and legal document comparison, where understanding similar or redundant clauses helps automate legal analysis.

2. Dataset Details

Source: Kaggle – Legal Clause Dataset

Link: <https://www.kaggle.com/datasets/bahushruth/legalclaudedataset>

Dataset Summary

Description	Count
Total Clauses	150,881
Unique Clauses after Cleaning	148,195
Training Clauses	118,451
Testing Clauses	29,744
Training Pairs	78,800
Testing Pairs	78,800
Sequence Length (MAX_LEN)	120 tokens

Each data point contains:

- **clause1, clause2** → two legal clauses
- **label (0 or 1)** → 1 if semantically similar, 0 otherwise

3. Data Preprocessing

Comprehensive deep preprocessing was applied to ensure text consistency:

- Lowercasing
- Removing punctuation, stopwords, and non-alphanumeric symbols
- Tokenization using Keras Tokenizer
- Sequence padding (max length = 120)
- Dataset split: 80% train / 20% test

Example pairs:

Lab	Clause 1 (excerpt)	Clause 2 (excerpt)
1	“indemnification contribution provisions set forth hereof contravene chilean law public policy”	“chevron agrees indemnify hold harmless underwriter person controls underwriter...”
0	“assignability employee may assign agreement third party without consent company”	“investment company act lender represents warrants qualified purchaser...”

4. Model 1 – BiLSTM Baseline

Architecture Summary

Layer	Output Shape	Parameter
Embedding	(None, 120, 128)	4,349,952
BiLSTM (shared)	(None, 128)	$98,816 \times 2$
Dense (ReLU)	(None, 128)	32,896
Dropout (0.3)	-	0
Dense (Sigmoid)	(None, 1)	129
Total Parameters	4,580,609	17.47 MB

Training Settings

- **Optimizer:** Adam ($\text{lr}=0.001$)
- **Batch Size:** 64
- **Epochs:** 10
- **Loss Function:** Binary Crossentropy

Training Graphs

- Loss steadily decreased from 0.28 → 0.02
- Validation accuracy improved from 0.95 → 0.98

Performance

Metric	Score
Accuracy	0.9876
Precision	0.9800
Recall	0.9956

F1-Score 0.9877

ROC-AU 0.9963

C

Sample Predictions

True Label	Pred	Observation
1	1	Correct – “noncompetition” clauses
0	0	Correct – unrelated indemnification vs headings
1	1	Correct – “partial invalidity” clauses
1	1	Correct – “compensation” clauses

5. Model 2 – Attention-based Encoder

Architecture Summary

Layer	Output Shape	Parameters
Embedding	(None, 120, 128)	4,349,952
LSTM × 2	(None, 120, 64)	49,408 × 2
AttentionLayer × 2	(None, 64)	184 × 2
Dense (128, ReLU)	(None, 128)	16,512
Dropout (0.3)	-	0
Dense (Sigmoid)	(None, 1)	129
Total Parameters	4,465,777	17.04 MB

Training Settings

- **Optimizer:** Adam
- **Batch Size:** 64
- **Epochs:** 10
- **Loss Function:** Binary Crossentropy

Training Graphs

- Validation accuracy plateaued at ~0.95
- Model converged slower than BiLSTM but remained stable

Performance

Metric	Score
Accuracy	0.9513
Precision	0.9189
Recall	0.9899
F1-Score	0.9531
ROC-AU	0.9688
C	

Sample Predictions

True Label	Pred	Observation
1	1	Correct – “noncompetition” clauses
0	0	Correct – unrelated reinstatement/subordination
1	1	Correct – “partial invalidity” clauses

1

1

Correct – “compensation” clauses

6. Comparative Performance

Model	Accuracy	Precision	Recall	F1	ROC-AUC
BiLSTM	0.9876	0.9800	0.9956	0.9877	0.9963
Attention Encoder	0.9513	0.9189	0.9899	0.9531	0.9688

7. Analysis & Discussion

- **BiLSTM Strengths:**

Captures long-term dependencies and clause sequence structure effectively.
Achieved higher accuracy and recall excellent for catching all true similar clauses.

- **Attention Encoder Strengths:**

Better interpretability attention highlights key legal terms (“indemnify”, “terminate”, etc.).

Slightly less accurate but provides more transparent semantic matching.

- **Limitations:**

No pre-trained embeddings (per assignment restriction), so vocabulary sparsity limits performance.

Some near-synonymous legal terms (“terminate” vs “rescind”) not captured well.

- **Metric Selection:**

F1-Score is ideal as it balances precision and recall, both important when missing or wrongly marking clauses could cause legal risk.

ROC-AUC supports robustness evaluation over multiple thresholds.

8. Conclusion

Both baseline architectures successfully modeled semantic similarity in legal clauses without any pre-trained transformer.

- BiLSTM outperformed attention in quantitative metrics.
- Attention Encoder provided richer interpretability and semantic focus.

Future work could incorporate:

- Pre-trained legal embeddings (e.g., Legal-BERT)
- Cross-category contrastive training
- Explainable attention visualization for clause comparison

9. References

- Bahushruth Legal Clause Dataset (Kaggle)
- TensorFlow / Keras API Documentation
- GoogleCloudPlatform keras-idiomatic-programmer Guidelines