CITS4012 – An Ablation Study on NLI Classifier Attention

Group 9

Abstract

This project studies how attention mechanisms affect Natural Language Inference (NLI) on science-specific data. We implemented a vanilla Bi-LSTM, self-attention Bi-LSTM, dual-attention Bi-LSTM, and a Transformer-based model to analyse both attention existence and encoder architecture. Results show that while overall accuracy remains similar, attention improves recall on the *entails* label and highlights key scientific tokens, enhancing interpretability. The Transformer captures long-range dependencies but tends to overfit due to the high OOV rate. Overall, lightweight attention designs improve reasoning without large pretrained models.

1 Introduction

Natural Language Inference (NLI) is the task of determining whether a hypothesis can be logically inferred from a given premise.

We design three BiLSTM-based models to study the impact of attention: a vanilla model without attention, a self-attention model that applies self-sentence attention, and a dual-attention model that incorporates cross-sentence interaction, along with a Transformer-based NLI model for encoder architectural ablation.

2 Methods

In this study, we implemented four different but related NLI models to investigate the effects of architectural ablations, including:

- **Attention Existence Ablation** compares the vanilla Bi-LSTM NLI classifier with its attention-enhanced counterpart to evaluate the impact of attention mechanisms.
- **Encoder Architectural Ablation** compares the dual-attention NLI classifier with its Transformer-based counterpart to assess the impact of replacing the recurrent encoder.

2.1 Model 1: The Vanilla Bi-LSTM NLI Architecture with Dual-Attention

As shown in Figure 1, we implement two similar models for attention existence ablation study.

First, we implement a simplified SNLI classifier as the baseline following Bowman et al. (2015). This model serves as the foundation without any attention mechanism. It is a simplest single-layer Bi-LSTM classifier.

Then, we extend this model by introducing both self-attention within the Bi-LSTM encoders of the premise and hypothesis, and sentence-level cross-attention between them, as our attention existence ablation study, following Li et al. (2024).

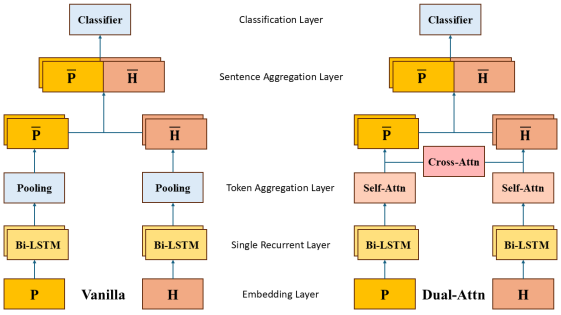

Figure 1: The Vanilla Bi-LSTM NLI Ablation.

2.2 Model 2: The Bi-LSTM Self-Attention Architecture

In this project, we also implemented a **Bidirectional Long Short-Term Memory (Bi-LSTM) model with an attention mechanism** to perform the Natural Language Inference (NLI) task. The objective of this model is to determine the logical relationship between a given *premise* and *hypothesis*, classifying each pair into one of two categories: **entailment** and **neutral**.

The Bi-LSTM encoder reads both the premise and hypothesis sequences in forward and backward directions, capturing contextual dependencies from

both ends of the text. The **attention layer** is then applied to dynamically weight the importance of each token in the sequence, allowing the model to focus on the most relevant words when inferring meaning. The representations of the premise and hypothesis are combined through concatenation, element-wise multiplication, and absolute difference operations to capture both their semantic alignment and contrast.

The combined feature vector is passed through fully connected layers for final **two-class classification**. The model is trained using the cross-entropy loss function and optimized with the Adam optimizer. Performance is evaluated on a separate validation set using **accuracy**, **confusion matrix**, and **classification report** metrics. This Bi-LSTM + Attention architecture effectively balances contextual understanding and interpretability, making it well-suited for reasoning tasks such as NLI.

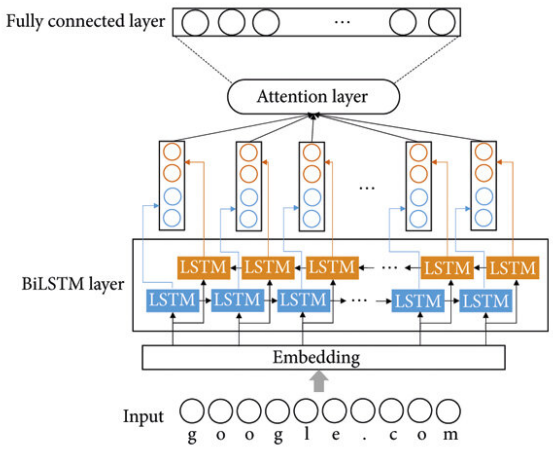

Figure 2: The Bi-LSTM Self-Attention Architecture.

2.3 Model 3: The Transformer-based NLI Classifier

In addition to the Bi-LSTM variants, we implemented a Transformer-based NLI classifier to examine whether non-recurrent self-attention can better capture long-range dependencies in science texts. We replace recurrence with a Transformer encoder to model long-range dependencies in scientific NLI. Premise and hypothesis tokens are mapped to trainable embeddings with sinusoidal positional encodings, then encoded independently by a single-layer multi-head self-attention encoder (residual + layer norm + feed-forward). Each sequence is grouped by mean into a sentence vector. We form the joint representation by concatenating the two vectors with their element-wise absolute difference and product, and feed it to a small MLP with dropout to predict entails vs neutral using cross-entropy and Adam.

Hyperparameters are aligned with prior models for fair comparison (embed/hidden 256, 1 encoder layer, 8 heads, dropout 0.1, same batch size and epochs). Evaluation mirrors earlier sections: accuracy, confusion matrix, and classification report on the validation/test sets.

3 Experiment Setup

3.1 Dataset Description

We are required to train, validate, and test our models using given science-specific Natural Language Inference (NLI) datasets. Each JSON dataset includes three fields: *premise*, *hypothesis*, and *label*.

The NLI task should analyze the input premise and decide whether its correlated hypothesis is entails or neutral (Miquido, 2024).

3.2 Cleaned Noises

In principle, we should only address the noises included in training set.

1. HTML/XML tags with ID pattern
2. Non-linguistic long/pure separators
3. Duplicate consecutive phrases
4. Single-word sentences
5. Duplicated whitespaces
6. Spaces before punctuations, except '!' and '?'
7. Premise with long concatenated sentences

3.3 Identified Noise

These are noises we identified but left unhandled due to either practical compromise or excessive complexity.

1. Instructional prompt words
2. Numbered markers
3. Misplaced label information
4. Metadata prefixes
5. Isolated single symbols

Figure 3: Premise Word Cloud (train.json).

3.4 Word Frequency

We use a word cloud visualization to illustrate the word frequency distribution of the science-specific training set after removing stopwords.

3.5 Word Embedding

To simplify the embedding process, we construct a word-to-index vocabulary directly from the training set.

Compared with training a Word2Vec embedding model, this approach is more lightweight and easier to implement. Each token is assigned a unique index, which effectively corresponds to a one-hot embedding. These indices are then fed into `torch.nn.embedding()` to generate dense embeddings for model training.

In addition, the OOV constraint imposed by the fixed vocabulary serves as a practical indicator for evaluating different model performances, with both the validation and test sets containing about 20% OOV tokens.

3.6 Padding and Batching

Since some premises contain long concatenated sentences forming entire paragraphs, excessive padding would introduce unnecessary computational overhead.

To mitigate this issue, we first identify and split such paragraph-style premises into individual sentences that align with their original hypotheses. Then, each sequence is simply padded to the maximum length in its field, and all padded pairs are grouped into mini-batches before being fed into the model.

3.7 Hyperparameters

To ensure consistency across all models, we define a shared configuration of hyperparameters for the entire experiment. The embedding and hidden dimensions are both set to 256, and each BiLSTM encoder consists of a single recurrent layer. A dropout rate of 0.1 is applied to prevent overfitting.

During training, the Adam optimizer is adopted with a mini-batch size of 128. The initial learning rate is set to 0.001 and kept constant throughout the training for stability. Each model is trained for 20 epochs, with monitored validation loss. The classifier outputs either *neutral* or *entails* corresponding to the dataset feature as we assumed.

4 Results

Most of our models show similar overall prediction accuracy at about 65%, but differ in their ability to predict the *neutral* labels as Table 1 shows.

4.1 Existence Ablation Study between Vanilla and Dual-Attention Models

The extended dual-attention mechanism does not introduce a significant training cost. The vanilla model requires around three minutes for a 20-epoch training, while the enhanced model takes roughly one minute longer, which is acceptable.

The training loss curves in Figure 4 show a similar trend that attention does not accelerate convergence but helps reduce loss fluctuation. While the increased validation loss reflects the persistent OOV issue, we attribute this to the relatively high OOV rate ($\approx 20\%$) rather than model overfitting.

The confusion matrices in Figure 5 reveal an improvement on the *entails* label when dual-attention is introduced. This may be due to the training-set imbalance, where two-thirds of the labels are *neutral*, causing the vanilla model to misclassify true positives as false negatives.

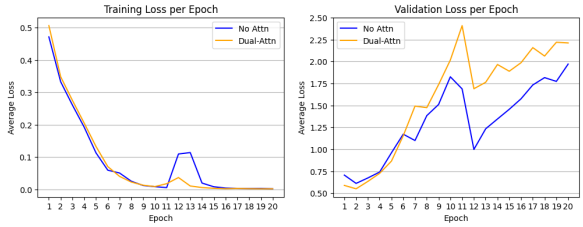

Figure 4: Model 1 Loss Comparison.

4.2 Encoder Architectural Ablation between Vanilla and Transformer models

To investigate the effect of encoder architecture, we replaced the BiLSTM encoder in the vanilla model with a Transformer encoder while keeping the classifier identical.

Model	Precision (Entails)	Recall (Entails)	F1-score (Entails)
Vanilla BiLSTM	0.66	**0.55**	0.60
Dual-Attn BiLSTM	0.63	**0.59**	0.61
Self-Attn BiLSTM	**0.80**	0.67	0.73
Transformer-Based	0.66	**0.74**	0.70

Table 1: Performance comparison of different models on the *entails* label.

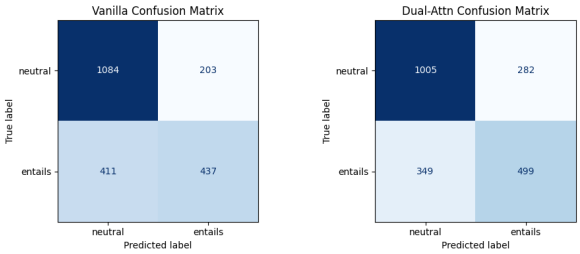

Figure 5: Model 1 Confusion Matrix Comparison.

As shown in Table 1, the Transformer-based model achieves higher recall (0.74) and F1-score (0.70) on the *entails* label compared to the vanilla BiLSTM (recall 0.55, F1 0.60). This indicates that the self-attention mechanism in the Transformer effectively captures long-range dependencies within both premise and hypothesis, improving entailment recognition beyond the recurrent encoder's sequential limitation.

4.3 The Self-Attention Bi-LSTM NLI model

The Bi-LSTM with Attention model achieved a **overall validation accuracy of 75%**, effectively distinguishing between "entails" and "neutral" relationships in the NLI task. It demonstrated balanced performance with an average **F1-score of 0.77**, though slight overfitting was observed after several epochs. Overall, the model successfully captured semantic dependencies between premise and hypothesis sentences, serving as a strong baseline for further enhancement.

4.4 The Transformer Encoder NLI model

The Transformer-based classifier converged quickly in the training set, with the training loss decreasing from about 0.50 to below 0.15, but the validation loss increased and fluctuated after the first few epochs, showing signs of overfitting. This behavior is consistent with the earlier models and is mainly caused by the fixed vocabulary and OOV constraint rather than by the network structure itself. In the test set of 2135 samples, both overall accuracy and the macro F1 were 0.68. For each class: neutral had precision 0.70, recall 0.61, and F1 0.65; entails had precision 0.66, recall 0.74, and F1 0.70. The confusion matrix shows that more neutral cases were misclassified as entails than the other way around, meaning the model tends to predict entailment more easily. Compared with the Bi-LSTM + Attention model, this Transformer performed worse overall, likely due to the lack of pretrained embeddings, the use of only one encoder layer with simple pooling, and the vocabulary limitation. In general, this lightweight self-attention model can still capture long-range dependencies and achieve reasonable recall on the entail class. Future improvements could include the use of pretrained Transformer encoders, improved grouping strategies, and subword tokenization to reduce OOV effects.

5 Qualitative Results

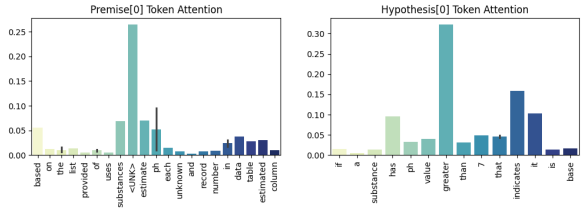

Figure 6: Sample Token Attentions

To qualitatively interpret how the dual-attention model performs entailment reasoning, we visualised the token-level attention weights just after its self-attention layer from a correctly predicted test instance Figure 6. The histogram illustrate that the model successfully highlights science-specific and comparative tokens after the self-attention layer.

In the sample premise, the model places noticeably higher attention on <*UNK*>, *substances*, and *estimate*, indicating its focus on domain-relevant terms describing experimental or quantitative concepts.

In the sample hypothesis, the strongest attention lies on *greater* and *indicates*, suggesting that the model effectively captures comparative and infer-

ential cues that are crucial for recognizing textual entailment.

These attention patterns suggest that the model learns to align semantic evidence between the premise and hypothesis, focusing on scientifically meaningful tokens that support logical inference. This behaviour provides qualitative evidence that the dual-attention mechanism enhances interpretability and mimics human-like reasoning in science-domain NLI.

6 Conclusion

This project explored how attention mechanisms and encoder architectures influence NLI model behaviour in the science domain. Through systematic ablation, we found that incorporating self- and dual-attention improves entailment recognition and interpretability, while a Transformer encoder enhances long-range dependency modeling but risks overfitting under limited data and vocabulary constraints. The key achievement lies in implementing multiple custom architectures from scratch and demonstrating the qualitative benefits of attention without relying on pretrained weights. The main limitation is the relatively small and noisy dataset with a high OOV rate, which restricts generalisation. Future work could explore subword tokenisation, pretrained embeddings, and deeper Transformer layers to further stabilise training and improve accuracy.

Team Contribution

Uni ID	Name	Contribution
24141207	Kaichao Zheng	Data Preprocessing; Vanilla NLI Dual-Attention Existence Ablation
24645175	Ziqi Meng	Self-Attention NLI
23998001	Yanglei Yuan	Transformer NLI

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv:1508.05326 [cs]*.

Peiguang Li, Hongfeng Yu, Wenkai Zhang, Guangluan Xu, and Xian Sun. 2024. Sa-nli: A supervised attention based framework for natural language inference. *Neurocomputing*, 407.

Miquido. 2024. What is natural language inference (nli)?