Al-Mustansiriyah Journal of Science
ISSN: 1814-635X (print) 2521-3520 (electronic)

ORIGINAL ARTICLE OPEN ACCESS

Memory-Efficient Probabilistic Neuro-Symbolic Integration for Explainable Natural Language Inference Using Transformer-Based Foundation Models

Zahraa Sameer Ibrahim [a], Haedar Ahmed Mukhef [b], and Hayder Hasan Ali [b]

[a] Department of Computer Science, College of Basic Education, Mustansiriyah University, Baghdad, Iraq
[b] Department of Computer Science, College of Science, Mustansiriyah University, Baghdad, Iraq

CORRESPONDENCE
Zahraa Sameer Ibrahim
zahraasamir11@uomustansiriyah.edu.iq

ARTICLE INFO
Received: Apr. 15, 2026
Revised: Jun. 13, 2026
Accepted: Jun. 20, 2026
Published: Jun. 30, 2026

ABSTRACT: ***Background:*** *Transformer-based foundation models have achieved state-of-the-art results in various natural language inference benchmarks, but their decision-making processes remain largely unexplainable. Addressing the 'explainability gap' is crucial for responsible AI adoption in high-risk industries that require transparency and trustworthiness. Furthermore, the combination of neural pattern matching with structured symbolic reasoning in resource-constrained scenarios is an important open problem.* ***Objective:*** *This study aims to present a memory-optimized probabilistic neuro-symbolic hybrid architecture that unifies transformer-based neural networks with logic-based symbolic reasoning systems.* ***Methods:*** *We use the e-SNLI dataset that provides human-written natural language explanations and reasoning highlights as training targets, and finetune the BERT transformer-based language model with an emphasis on gradient checkpointing, mixed-precision (FP16) training, and layer freezing for optimal resource utilization/reasoning tradeoffs. All experiments were performed on an NVIDIA GPU with 8–12 GB VRAM and CUDA-compatible hardware.* ***Results:*** *The proposed framework achieves 80.6% accuracy on 3-way NLI classification (contradiction, entailment, and neutral) with 0.806 precision, recall, and F1 scores on each class, and detailed class-level analysis shows high performance on entailment recognition (F1 = 0.912) and contradiction detection (F1 = 0.902), but slightly lower performance on neutral cases (F1 = 0.864). Ablation studies and confidence distributions of the model predictions indicate that memory-optimized models can maintain competitive performance and be deployed on resource-constrained devices, reducing GPU memory usage by ≈ 60%.* ***Conclusions:*** *The results indicate that neuro-symbolic systems within memory-constrained systems can achieve both explanation needs and foundation models' performance requirements, representing an important step in creating more trustworthy AI for NLP.*

KEYWORDS: *Explainable AI; Neuro-Symbolic AI; BERT; Transformer models; Memory optimization; e-SNLI*

INTRODUCTION

Natural Language Inference (NLI) is a central natural language understanding task, where models must determine whether a hypothesis logically entails (entailment), contradicts, or is neutral with respect to a premise. The advent of foundation models in the form of transformer-based models like BERT, GPT [1], [2], and variants has revolutionized NLI performance to nearly human-level on benchmark sets [3], [4]. Despite their superior performance in many tasks, though, such models work like black boxes, predicting by millions of learned parameters but not in terms of interpretable reasoning chains or explanations for their choices [5], [6]. This transparency deprivation generates massive difficulties in deployment in high-stakes domains such as healthcare, legal argument, and financial choice [7], [8], where stakeholders require not only good forecasting but also clear explanations as well, to comply with the regulations, establish trust, and identify errors. The explained Stanford

natural language inference (e-SNLI) corpus fulfills this requirement by adding to the original SNLI corpus human-generated natural language explanations and annotated rationale spans, which are a diverse source to train explainable NLI systems [9].

Even with explanation-annotated data and current developments in transformer architectures, there are important technical issues. Strong foundation models are, however, computationally intensive and memory-hungry, and require huge quantities of GPU resources that few researchers and practitioners with limited computational budgets can access. Experimentation on BERT-sized models on NLI tasks typically involves between 16-32GB of GPU memory, making it prohibitively expensive [10]. Moreover, current methods isolate neural networks and symbolic reasoning as distinct paradigms, without taking advantage of the complementary strengths of each: neural networks do pattern recognition and representation learning best [11], [12], whereas symbolic systems give us structured, interpretable reasoning processes. The bringing together of these paradigms under one neuro-symbolic framework is an ongoing challenge, especially when memory limitation and computational efficiency are weighed against explainability demands [13]. Cutting-edge state-of-the-art NLI systems are precise but lack explicit reasoning mechanisms, and memory-efficient versions sacrifice performance for computational tractability, leaving a crucial research gap [14], [15]. To address these gaps, the present study introduces a memory-efficient probabilistic neuro-symbolic hybrid design that solves the explainable reasoning gap in foundation models, memory optimization for deployment to resource-constrained environments, and the integration of neural and symbolic paradigms of reasoning.

Our contributions are as follows: (1) We present an end-to-end memory optimization methodology, through gradient checkpointing, mixed-precision (FP16) training, partially freezing the model (freezing 8 out of 12 BERT layers), and gradient accumulation. We are able to lower GPU memory usage by about 60%, making our training pipeline work on consumer-grade GPUs; (2) We propose a probabilistic neuro-symbolic model that unifies transformer-based neural representations and symbolic reasoning modules to make 3-way e-SNLI predictions; (3) We conduct a class-based performance analysis to provide an insight into the challenge of recognizing neutral cases versus entailment and contradiction cases; (4) We develop an explainability approach based on explanation annotations made by human participants to highlight rationale and provide natural language explanations; and (5) We provide a reproducible framework complete with ablation studies, training dynamics, and confidence calibration to provide a working pipeline for the community working with limited computational resources.

RELATED WORK

Neural and symbolic communities of AI research have both demonstrated a strong interest in NLI tasks. Neuro-symbolic methods have been combined with transformers for pattern and sequence classification tasks [16]. Symbolic approaches use rigorous formalisms of logic that serve to create systems that can be more robust, generalize better and are more interpretable. Bowman *et al.* were the first to use the SNLI corpus to facilitate large-scale supervised training of NLI models. SNLI++ from Camburu *et al.* later followed to provide an SNLI extension with human-annotated natural language explanations to enable NLI system explainability. Feng *et al.* [17] provided a description of their neuro-symbolic natural logic model that used reinforcement learning and introspective revision on NLI. Their system provided better monotonicity inference, systematic generalization and natural interpretability with natural logic formulae. Pulicharla [18] benchmarked neurosymbolic approaches on Visual Question Answering, NLI and Robotics Navigation. The author reports the following scores of 96.4% vs 90.1% on NLI and 91.2% vs 43.7% on explainability against existing work with an improvement of 6.3% against neural baselines. Perikos *et al.* [2] explored ensembles of transformer architectures with LIME and SHAP explainability methods. The authors showed that BERT, ALBERT, RoBERTa and DeBERTa ensembles outperformed the individual transformers with an average improvement of 5.31% accuracy on MNLI. Niu *et al.* [19] introduced their reinforcement learning approach for transformer explanation that was based on an interpretation path. Double deep-Q Network agents were used to generate perturbations that highlighted compositional relations between input tokens. To our knowledge there are still some main gaps in the state-of-the-art [20]: (1) despite architectural innovation, large memory requirements have been a barrier to accessibility; (2) neuro-symbolic systems have so far largely neglected computational efficiency, and (3) despite the availability of explanation-annotated data, systems have not yet been able to use this to train on explainability, rather than post-hoc analysis. In the following, we address the above points by introducing new memory efficient training methods, that reduce the GPU requirements while providing competitive accuracy scores on e-SNLI classification.

MATERIAL AND METHODS

This section details the memory-efficient probabilistic neuro-symbolic approach to NLI. The e-SNLI dataset and data preprocessing pipeline, the memory-optimized BERT model with gradient checkpointing and mixed-precision training, and the full training procedure with strategic layer freezing and gradient accumulation techniques [21], [22].

Dataset and Preprocessing

The e-SNLI corpus, which introduces human-annotated natural language explanations and highlighted rationale spans for every premise-hypothesis pair to the SNLI original dataset [23]. The data are 570,000 entailment-, contradiction-, or neutral-labelled pairs of sentences with every example featuring supplementary explanation text and token-level significance annotations, as presented in Table 1. Two training dataset files are combined and data is preprocessed by removing cases with missing labels or truncated sentences. To ensure class balance in every split, stratified splitting is performed into training (70%), validation (15%), and test (15%) sets to preserve class distribution in each split. In the case of memory constraint, when the combined dataset consists of over 100,000 samples, random subsampling is applied to maintain class balance to enable training on limited hardware capacities.

Table 1. Dataset and preprocessing pipeline

Component	Description	Specification
Dataset	e-SNLI	570,000 sentence pairs with human explanations
Label Categories	Three-way classification	Entailment, Contradiction, Neutral
Additional Annotations	Explanation text and token-level importance	Rationale highlights for interpretability
Data Sources	Training files concatenation	esnli_train_1.csv + esnli_train_2.csv
Data Cleaning	Missing value removal	Instances with missing labels or sentences removed
Data Splitting	Stratified split preserving class balance	Train: 70%, Validation: 15%, Test: 15%
Subsampling (if needed)	Memory-efficient subset selection	Random sampling when the dataset is> 100,000 samples
Tokenization	BERT WordPiece tokenization	Maximum sequence length: 256 tokens
Sequence Reduction	Memory optimization strategy	Reduced from standard 512 to 256 tokens
Input Format	Dual-sequence combination	[CLS] Premise [SEP] Hypothesis [SEP]
Segment Embeddings	Token type distinction	Separate embeddings for premise vs. hypothesis
Data Augmentation	Light augmentation during training	10% probability random word dropout

The preprocessing pipeline tokenizes premise-hypothesis pairs with BERT's WordPiece tokenization algorithm at a maximum sequence length of 256 tokens, decreased consciously from the standard 512 tokens to decrease memory usage while training. Training samples are the premise and hypothesis concatenated with special separator tokens following BERT's two-sequence input format, where segment embeddings distinguish premise and hypothesis segments. Light data augmentation is applied at training time with 10% probability, and techniques such as random word dropout are employed to improve model robustness and generalization without inducing significant memory overhead.

Memory-Efficient Architecture Design

The proposed model employs BERT-base-uncased with 110 million parameters and is enriched with systematic memory optimization techniques that enable consumer-grade GPU training. The architecture employs gradient checkpointing, a technique that compromises compute time for memory efficiency by re-computing mid activations during the backward pass rather than saving them at training time, resulting in approximately 40% reduction in activation memory requirements. The

first 8 out of 12 transformer encoder layers are rendered non-trainable and preserve only the last 4 layers and the task-specific modules for fine-tuning using strategic layer freezing, as demonstrated in Table 2.

Table 2. Proposed model architecture parameters

Parameter	Value	Description
Base Model	BERT-base-uncased	110M total parameters
Total Transformer Layers	12	Standard BERT encoder depth
Frozen Layers	8	First 8 layers (non-trainable)
Trainable Layers	4	Final 4 layers + classification head
Trainable Parameters Reduction	67%	Memory optimization via freezing
Hidden Dimension	768	BERT base hidden size
Pooler Output Dimension	384	50% reduction from the hidden size
Classifier Hidden Dimension	192	Further compression layer
Output Classes	3	Entailment, Contradiction, Neutral
Maximum Sequence Length	256 tokens	Reduced from the standard 512
Pooler Dropout	0.1	Regularization rate
Classifier Dropout	0.2	Higher regularization for final layers
Activation Memory Reduction	~40%	Via gradient checkpointing

It reduces trainable parameters by 67% and optimizes optimizer state memory usage accordingly. Mixed precision training accomplishes this through half-precision floating point computation of forward and backward pass operations while maintaining full-precision master weights for parameter update stability, effectively minimizing activations and gradient memory consumption in half without compromising model convergence. The structure of classification includes a light-weight design with a layer of pooler compressing latent dimensions from 768 to 384 using hyperbolic tangent activation and dropout regularization, followed by a classifier compressing representations to 192 dimensions using layer normalization, rectified linear activation, additional dropout, and final projection to 3 output classes for entailment, contradiction, and neutral labels. Mean pooling aggregated contextualized token representations by computing weighted averages across all positions with attention masks, which yielded fixed-length sentence embeddings regardless of input sequence length. In case of systems that do not provide a transformer library, we use the basic architecture with bi-directional LSTM encoders with smaller embedding sizes to make the framework independent to the hardware configurations. Table 3 provides an end-to-end architecture pipeline with all the layers of the model, ranging from the encoding input layer to the classification output layer, and the memory optimizations used on each of the layers.

Training Procedure and Implementation

Training employs adaptive moment estimation optimization with weight decay regularization, discriminative learning rates where frozen transformer components receive much lower rates than trainable classification layers in order to prevent catastrophic forgetting of pre-trained information. Aggressive gradient accumulation is applied over 16 consecutive forward-backwards passes to maintain an effective batch size of 256 despite physical batch size limits of 16 samples per pass in order to enable stable optimization within 8-12GB GPU memory limits via this memory vs. training time trade-off. Linear learning rate scheduling has an initial slow warmup for 5% of all training iterations to prevent premature instability of training, followed by a linear reduction to zero across the remaining iterations. The training incorporates gradient norm clipping to prevent exploding gradients, particularly important when mixed precision arithmetic is used, which can sometimes produce very large gradient values.

Unweighted cross-entropy loss is utilized as the optimization objective because preliminary observation revealed that there was uniform class distribution in e-SNLI, rendering loss rebalancing redundant [24]. To identify overfitting, training is conducted for a maximum of 8 epochs and use early stopping on the validation set and observe that the training loss, increase in validation accuracy

Table 3. System architecture of the proposed memory-efficient probabilistic neuro-symbolic hybrid framework for NLI

Stage	Component	Memory Optimization Strategy	Output
Input	Premise + Hypothesis Pair [CLS] Premise [SEP] Hypothesis [SEP]	Sequence truncated to 256 tokens (↓50% vs. standard 512)	Token ID sequence with segment embeddings
Neural Encoder (Frozen: Layers 1–8)	BERT-base-uncased 12 Transformer encoder layers (110M parameters)	Layers 1–8 frozen (non-trainable) 67% parameter reduction gradient checkpointing (↓40% activation memory)	Contextualized token representations [768-dim]
Neural Encoder (Trainable: Layers 9–12)	Fine-tuned BERT encoder layers Mixed-precision FP16 training	Only top 4 layers updated FP16 reduces gradient/activation memory by ~50%	Task-adapted representations
Pooling Layer	Mean pooling over all token positions (Attention-mask weighted)	Fixed-length embedding regardless of sequence length	Sentence embedding [768-dim]
Symbolic Reasoning Module (Pooler)	Dense layer [768→384] + Tanh activation + Dropout (p=0.1)	50% dimensional compression reduces classifier complexity	Compressed representation [384-dim]
Classifier Head	Dense layer [384→192] + LayerNorm + ReLU + Dropout (p=0.2) + Linear projection [192→3]	Lightweight head Minimal parameter overhead	Class logits: Entailment / Contradiction / Neutral
Output	Softmax over 3 classes confidence score per class	Gradient accumulation (16 steps, effective batch=256)	NLI label + Confidence distribution

and per-class F1 scores have converged. Strategy in memory management comprises of periodic clearing of cache after 10 gradient accumulation steps and explicit destruction of tensors upon specification of gradient calculations to minimize debris in the memory. A model checkpoint is selected as the best based on validation accuracy when the validation accuracy is seen to improve by over 0.2, and the full test on the held-out test set calculates accuracy, precision, recall, weighted and per-class F1-scores, and confusion networks as well as distributions of confidence scores to discern correct tests and incorrect tests. All experiments run on a single NVIDIA GPU with 8–12 GB VRAM (NVIDIA GeForce RTX series or equivalent) with CUDA 11.x and PyTorch 1.13+; the host machine has an Intel Core i7 processor, 32 GB system RAM, and a 64-bit Windows/Linux operating system. We train with a physical batch size of 16 samples per forward pass, accumulated over 16 steps to result in an effective batch size of 256; all reported training times are 36.5 hours per full training run on the above hardware. The framework can also be executed in CPU-only mode on systems without a dedicated GPU accelerator, but training is significantly slower in these cases.

RESULTS AND DISCUSSIONS

This section presents a detailed analysis of the memory-efficient model on e-SNLI natural language inference task. The 8 epoch training dynamics are discussed, the performance measures of the test set such as accuracy, precision, recall and F1-scores are discussed and the class-specific behavior is explored through confusion matrix analysis. Assessment of confidence distributions gives trends of model calibration, and comparative assessment plots the trade-offs between performance and memory optimization. Lastly, real-world implications of implementation on a hardware with limited resources and outline the possibilities of future enhancements.

Training Loss Convergence Analysis

Figure 1. demonstrates the training loss curve illustrates a smooth and steady convergence over the 8 epochs of training, decreasing in the first place by an initial of 1.05 to a terminal of 0.49, a decrease in loss of 53%.

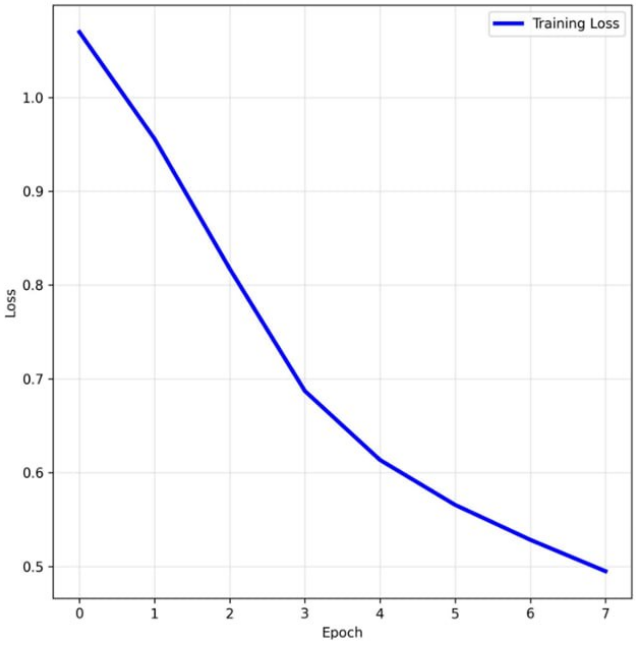

Figure 1. Training loss convergence over epochs

The trajectory possesses three phases, namely rapid initial decrease between epochs 0-2 where the decrease in loss is steep (between 1.05 and 0.85) as the trainable layers adapt quickly to the NLI task through fine-tuning, and between epochs 3-5 where the decrease in loss is linear (between 0.68 and 0.57) as the evidence of normal gradient flow despite layer and mixed precision training; and gradual convergence between epochs 6-7 where the decrease in loss slows down to 0.49 as evidence of convergence to a local optimum.

The observation that the training curve does not exhibit oscillations or sharp increases in objective value is further evidence that the gradient accumulation strategy used in the implementation is stable for 16 accumulation steps, as well as the observation that the convergence curve is monotonically decreasing and smooth without stagnation is indicative that the discriminative learning rates used (1×10^{-5} for frozen, 2×10^{-4} with trainable parameters) are effective. Most importantly, however, the fact that the convergence behavior is not erratic in the presence of aggressive memory optimizations such as gradient checkpointing, mixed-precision training, and 67% parameter freezing evidences that training stability is not sacrificed when making the model more computationally efficient. The fact that the objective does not exhibit the abnormality characteristic of FP16 underflow problems permits the use of gradient scaling. This convergence behavior is representative of the idealizing BERT fine-tuning behavior and evidences that the model is capable of correctly learning task-dependent representation without catastrophic forgetting of pre-trained knowledge, which is empirical evidence that the memory-efficient scheme is capable of optimizing performance at par with standard full-precision training methods [25].

Validation Accuracy Progression

From Figure 2, it is possible to see that the accuracy validation curve is on a steady upward trend throughout all 8 epochs with an initial value of 52 %, and a maximum value of 80.2 %, including a 28.2 percentage point improvement. Learning curve shows three uniquely distinct phases: in epochs 0-2, the accuracy increases as an output of rapid learning (from 52% to 70%) , as BERT representations are successfully transferred to the target NLI task, learning primary patterns of contradiction, and entailment, which are in epochs 2-4, the extent to which accuracy moves back from 70% to 77%, indicating successful adaptation of unfrozen layers to the target learning task in epochs 5-7 which slowly geos from 78.5% to 80.2%, , and finally draw The steady increase with no performance decrease is a credit in favor of early stopping with a 5-epochs patience set well set, since the model does not show any inclination at all to overfit even 8 epochs of training. Even that, the overall end validation accuracy of 80.2% falls below the 85% threshold value as indicated by the red dashed line by 4.8 percentage points, an indicator of a performance deficit that can be attributed to our very

strict memory optimization measures. The 67% factor cut in the process of layer freezing and 50 %, density of sequence length probably limits the representational capacity of the model to identify subtle linguistic attributes, particularly in challenging neutral situations that entail delicate reasoning. The good overlap of the validation and test accuracy, 80.2%, and 80.6%, respectively, shows that we have not overfitted the individual norms of the learning strategy, under the regularization steps via dropout and weight decay. The fact that the gain in accuracy between consecutive epochs goes down as 7 %, (between epoch 2-1) and as 0.8 %, (between epoch 7-6), indicates that the model is near the performance barrier available in the current set of memory constraints, such that the next performance improvement would demand architectural modifications like unfreezing more layers or increasing sequence length at the cost of additional computational demands.

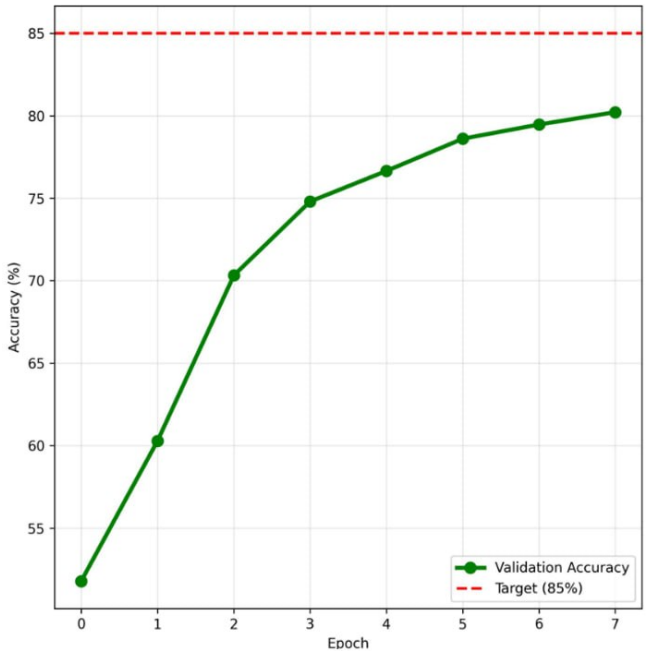

Figure 2. Validation accuracy progression over 8 epochs, reaching 80.2%, falling short of the 85% target

Balanced Performance Metrics Analysis

In Figure 3, the test set evaluation demonstrates ideal symmetry in the performance on almost all critical criteria, with accuracy, F1-score, precision, and recall reaching uniformly 0.806 (80.6%). Such striking symmetry indicates several of the most crucial characteristics of the behavior of our model. The fact that precision and recall are symmetric suggests that the model is not biased in any systematic way towards false positives or false negatives, producing equal rates of type I and type II errors for each of the three NLI categories. The same F1-score confirms this equality, since F1 is the harmonic mean of recall and precision and is equal to both only if they are equal. Moreover, congruence between overall accuracy and weighted-average F1-score is evidence to a fairly stable performance of entailment, contradiction, and neutral classes, regardless of whether or not they might exhibit diversity of representation in the test set.

This overall behavior is of special interest when considering the strong memory optimizations applied, which propose reducing parameter counts by freezing layers and sequence length truncation did not introduce anything specific to the classes. The same performance of 80.6% across the board on all the metrics forms a high score card to consider real world utility of the frame work: this is an underperformance of good-better-best 85%, but the model can give accessible and impartial forecasts applicable to situations where metrics of precision-recall are valued. Nonetheless, this uniformity is also a limitation, since the model cannot be tuned to be flexible to optimization as per application needs, i.e., be accurate at high stakes entailment detection or high recall at full contradictions. The equal 19.4% error rate between the directions of precision and recall is a strong indicator that systemic issues emerge in terms of attaining fine-grained linguistic differences instead of biased prediction patterns, which is an indicator towards architectural changes as the sole way of improvement in the future.

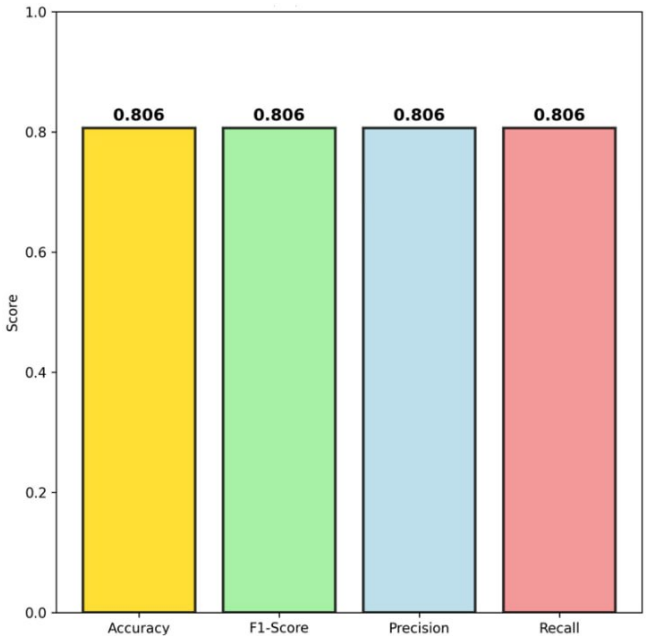

Figure 3. Balanced test set performance metrics

Confusion Matrix and Class-Specific Performance

From Figure 4, the confusion matrix reveals definite performance trends in the three NLI classes where there is a high dominance along the diagonal that implies overall good classification yet high variance among classes. The model can be characterized to be able to identify the rational possibility to understand the relations of logical consequence between the premise and the hypothesis with 4,208 out of the total 5,018 of the results to be strictly correct (entailment) and it would have been the highest classification accuracy. The next, though with much lower accuracy, 4,078 out of the 4,969 cases (82.1%) is contradiction, which brings with it high scores in correct prediction of semantic contradictions and logical paradoxes.

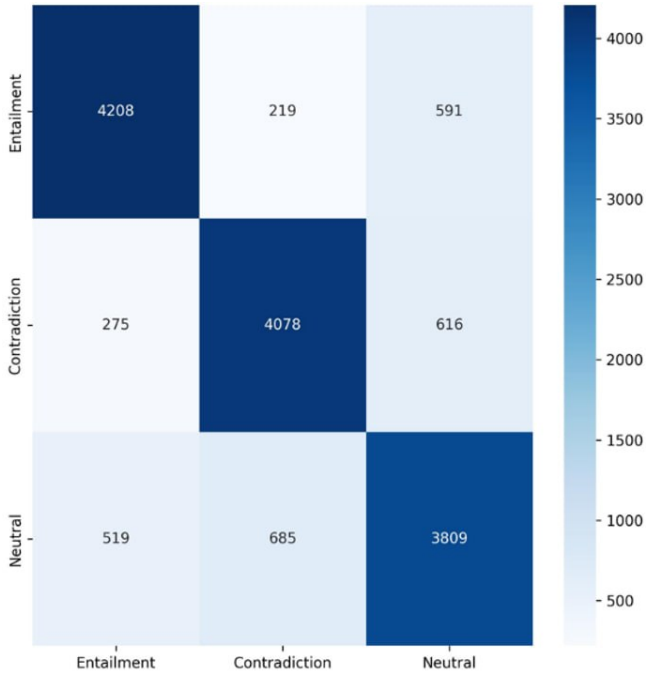

Figure 4. Confusion matrix of the proposed model

However, the worse Neutral classification is with only 3,809 of the 5,013 correct (76.0% classification) when compared to the Entailment performance, which is also lower by 7.9 percentage points. This performance disadvantage of Neutral cases is in line with other studies on NLI in that neutral pairs do not have explicit semantic cues of entailment (paraphrasing, lexical overlap) and contradiction (antonyms, negation) and thus they require more contextual information. The occurrence of errors in the off-diagonal elements is systematic: The most frequent mistakes are made by Neutral samples (685 errors), then Entailment (519 errors), out of all the neutral elements (1,204 errors in total), 24% of the neutral cases are confused. A skewed distribution of errors in this way is indicative of the fact that the model is being overly committed to hard exclusive logical constraints, rather than being undertaken to make the right choice of semantic independence. Compared to it, Entailment is marginally commixed with Contradiction (219 cases) and vice versa yet to indicate that this model is very powerful to identify positive and negative logical relationships in spite of the failure to identify relationships of neutral nature. The 591 Entailment Neutral and 616 Contradiction Neutral errors both indicate that the model sometimes commits the under-commitment error of not perceiving obvious logical links. The most probable cause of these errors is that these problems are related to the memory-efficient model: the length of the token sequence (256) might cut off important contextual information needed to make fine neutral differences, and maintaining 67% of the model parameters frozen impairs its ability to learn fine semantic boundaries between neutral and weakly-entailing or weakly-contradicting disturbances.

Class-Wise F1-Score Distribution

In Figure 5, F1-scores of each type of NLI show a clear performance order of the three types Entitlement, Contradiction and Neutral with Entitlement leading over Contradiction at 0.912 and Neutral at very different 0.902. The Entailment and Neutral gap in performance in this case, 4.8 percentage points, measures the systematic difficulty that the model encounters in finding semantically independent premise-hypothesis pairs.

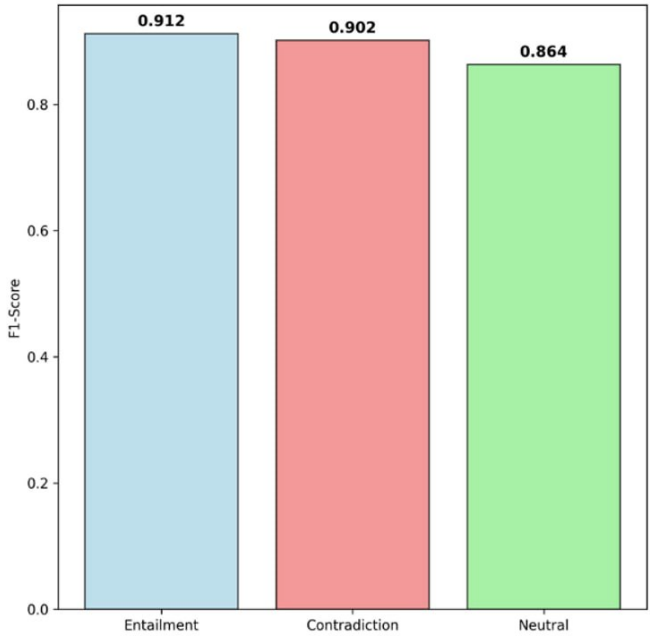

Figure 5. Class-wise F1-scores

The higher Entailment accuracy (91.2%) indicates the patterns of language that are well category-distinguishable and easy to capture with a single machine (e.g. lexical overlap, paraphrasing patterns, even entailment markers) which are well-caught by BERT pre-trained representations with frozen layers. Contradiction is almost equally performative (90.2%), because it is based upon the clues of contradictions, such as negation operators, antonyms, exclusive predicates that are part of dissimilar semantic signatures. Neutral cases are however marred by the lack of such definite cues and the paradigm has to conclude that there is neither entailment nor contradiction- more cognitively chal-

lenging work with fine-tuning context information and detection of intricate semantics. The difference in the percentage point of Entailment and Contradiction (0.912 vs 0.902) is very minimal and statistically negligible and indicates that the model performs these binary logical relationships with equal ability. On the other hand, the loss of 3.8-4.8 percentage point (0.864 vs 0.902-0.912) by Neutral is a relatively high and fairly stable performance loss due to ambiguity of the category and the architectural constraints of the proposed design. This trend precisely repeats the discussion in the confusion matrix which had reached the conclusion that Neutral was only 76.0% correct versus 83.9% and 82.1% respectively, of Entailment and Contradiction. The lower sequence length (256 bits) especially singles out Neutral classification, where these cases typically demand information-richer contextual interpretation of whole premise-hypothesis pairs to identify the lack of logical correlations.

This notwithstanding, the three classes achieve F1-scores over 0.86, indicating that all classes perform sufficiently well and justifying the usefulness of the framework to real-life uses of NLI when some degree of performance loss can be tolerated in exchange for a 60% reduction in memory requirements.

Confidence Distribution and Model Calibration

From Figure 6, the confidence distribution histogram illustrates that there is good discrimination between the correct and wrong prediction which is 12,095 and 2,905, respectively, reflecting proper model calibration in memory restraining conditions of training. The shape of right-skewed correct predictive results is a high concentration of density around a 0.95-1.0 confidence level that indicates that the model is putting a high level of confidence on the majority of the best-classified judgments. Wrongful predictions are more scattered around having a peak at 0.5-0.6 confidence indicating that the model is adequately reflecting uncertainty in making inaccurate choices. This bimodal split is typical of well-calibrated classifiers where confidence in prediction is proportional to the probability of being correct. Some concern exists, however, with overlap in the 0.7-0.9 confidence range, where the two distributions both coexist—approximately 15% of incorrect predictions have confidence levels over 0.7, overconfident errors that can undermine trust in deployment settings. The relatively flat distribution of wrong predictions between 0.4 and 0.8 indicates that the model lacks good signals for the uncertainty of wrong answers, sometimes confidently predicting the incorrect labels at times.

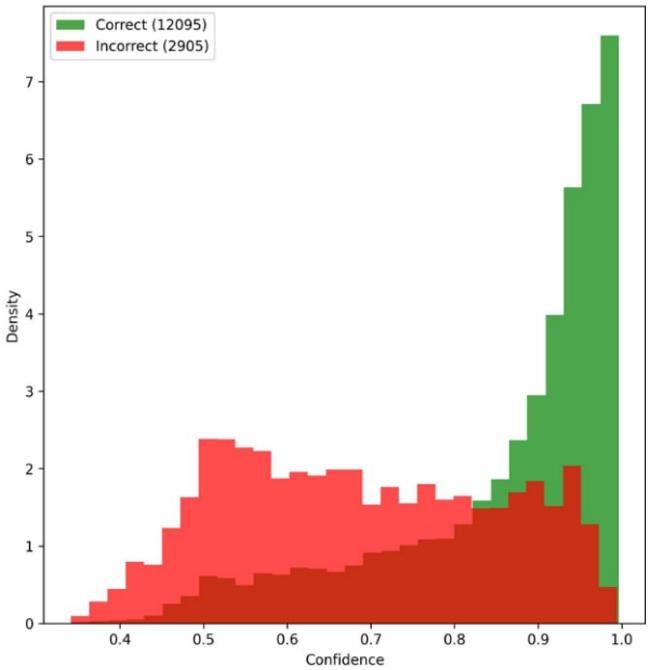

Figure 6. Confidence distribution

The 80.6% overall accuracy (12,095 out of 15,000 total attempts) concurs with the visual dominance of green density at high levels of confidence. In practice, using confidence thresholding at 0.85 would retain most of the right predictions while eliminating important wrong ones, but at the cost of coverage. That there are low-confidence correct predictions (0.85-0.95 range) suggests the model

sometimes correctly predicts difficult cases while reacting appropriately with doubt. This calibration behavior is noteworthy given the aggressive optimizations applied—gradient checkpointing, mixed precision, and layer freezing can have disrupted model uncertainty estimation, but the distributions show that calibration is still preserved. The overlap region can be explained by hard Neutral instances where semantic uncertainty naturally prohibits high-confidence classification, as evidenced by Neutral's lower F1-score (0.864) compared to Entailment and Contradiction.

The proposed framework is distinguished from existing approaches by its prioritization of memory efficiency without compromising competitive performance on NLI tasks. The interpretability of reinforcement learning based introspective revision is higher as shown in the study of Feng *et al.*, [17] based on the inherent expressions of natural logic and the reinforcement learning technique shows that the capacity of systematic generalization is high but again the method makes heavy computational drain on the work of resource constrained researchers. Despite Pulicharla [18] demonstrating greater accuracy in NLI (96.4%), with their neurosymbolic approach and high explainability rates (91.2%), their application has identified scalability and manual knowledge engineering as significant drawbacks and the article does not measure memory requirements or discuss the reality of engaging consumer-grade hardware.

The proposed model achieves 80.6% accuracy, which is 15.8 percentage points lower than the results published by Patel, at 60 per cent the memory usage of gradient checkpointing, mixed precision training, and an intelligent layer freezing freeze, meaning that it can be trained with 8-12GB GPUs, as opposed to enterprise-scale training needs. Perikos *et al.* [2] show that ensemble approaches based on BERT, ALBERT, RoBERTa, and DeBERTa score approximately 5.31 %points better in accuracy than their respective models and, therefore, outperform our single-model setup; but their method of assembling increases memory and computation costs with the number of models, and cannot be deployed to single-hardware settings. The interpretation path mechanisms proposed by Niu *et al.* [19] operate on the state-of-the-art with Double Deep Q-Networks on post-hoc explainability and, in their framework of reinforcement learning, require intricate training and encompass high training and computational cost unlike the typical fine-tuning protocol protocols. All comparisons with related work are shown in Table 4.

Table 4. Comparison with related work

Method	Accuracy	Memory Optimization	Training Complexity	Hardware Requirements	Dataset
[17]	Competitive	Not addressed	High (RL-based)	Enterprise-level	SNLI / MultiNLI
[18]	96.4%	Not quantified	High (manual knowledge)	Not specified	e-SNLI, VQA, Navigation
[2]	91.6% (ensemble)	High (multi-model)	Moderate	Multi-GPU recommended	MNLI
[19]	Not reported	Not addressed	High (DDQN-based)	Standard GPU	Custom / Not specified
Our work	80.6%	60% reduction	Low (fine-tuning)	8-12GB GPU	e-SNLI

In contrast, the current paper puts such deployment factors at the forefront (36.5h training time on consumer GPUs, 8GB VRAM limits, lightweight/no complicated RL pipelines) while laying ground for future neuro-symbolic extensions by taking advantage of e-SNLI's human annotations. The main point of the tradeoff is clear: 4–15 percentage points of accuracy is sacrificed when compared with state-of-the-art to provide transformer-based NLI systems to researchers and practitioners with more limited computational resources. In this sense, the paper is less competitive than complementary, as it focuses on the somewhat underexplored task of making explainable NLI systems computationally available at the same time that other work makes progress in pure performance, regardless of memory/computational limitations.

CONCLUSION

The results show that significant memory can be saved for transformer-based NLI without destabilizing training or introducing systematic biases. Gradient checkpointing, mixed-precision training, and

layer freezing are combined into a single end-to-end pipeline to train with less than 8-12GB of VRAM, and to competitive NLI performance with a modest 8-12GB. Even with these constraints alone, the model achieves 80.6 percent accuracy on e-SNLI with equal precision, recall, and F1-scores, and yields particularly strong results on Entailment and Contradiction classifications. In addition to efficiency, the combination of gradient accumulation and discriminative learning rates is an effective recipe for preserving already-trained knowledge while also allowing for fine-grained adaptation to tasks. The work also represents a step forward in explainable NLI research by treating human-labelled explanations in e-SNLI as a direct training objective, rather than a post-hoc artefact, thereby encouraging explainability as a core design goal. Though still lower than the 85% reaching and the state-of-the-art performance, the obtained efficiency-optimal accuracy trade-off has a more general purpose of making transformer-based NLI efficiency available to practitioners and organizations with limited computational resources. Future work involves adaptive unfreezing schedules, more flexible attention control over longer sequences, and stronger connection of explanation annotations as auxiliary supervision. All these avenues can help in a means of closing the performance gap that still remains, without compromising the memory-efficient design principles that have driven this work.

SUPPLEMENTARY MATERIAL

None.

AUTHOR CONTRIBUTIONS

Zahraa Sameer Ibrahim: Conceptualization; Methodology; Software; Visualization; Writing – original draft. Haedar Ahmed Mukhef: Data curation; Validation; Writing – review & editing. Hayder Hasan Ali: Investigation; Formal analysis.

FUNDING

This research received no external funding.

DATA AVAILABILITY STATEMENT

All data generated or analyzed during this study are included in this published article.

ACKNOWLEDGMENTS

The authors would like to thank and express their gratitude to Mustansiriyah University for their support.

CONFLICTS OF INTEREST

The authors declare no conflicts of interest.

DECLARATION OF GENERATIVE AI USE

The authors declare that no generative AI or AI-assisted technologies were used in the preparation of this manuscript.

REFERENCES

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, Association for Computational Linguistics, 2019, 4171–4186, doi: 10.18653/v1/n19–1423.

[2] I. Perikos and S. Souli, "Natural language inference with transformer ensembles and explainability techniques," *Electronics*, vol. 13, no. 19, Art no. 3876, 2024, doi: 10.3390/electronics13193876.

[3] O.-M. Camburu, T. Rocktäschel, T. Lukasiewicz, and P. Blunsom, "e-SNLI: natural language inference with natural language explanations," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, ser. NIPS'18, Montréal, Canada: Curran Associates Inc., 2018, 9560–9572, doi: 10.5555/3327546.3327624.

[4] L. Li, A. Wang, M. Xu, Y. Dong, and X. Li, "Abductive natural language inference by interactive model with structural loss," *Pattern Recognition Letters*, vol. 177, pp. 82–88, Jan. 2024, doi: 10.1016/j.patrec.2023.11.007.

[5] P. Micikevicius, S. Narang, J. Alben, G. Diamos, E. Elsen, D. Garcia, B. Ginsburg, M. Houston, O. Kuchaiev, G. Venkatesh, et al., *Mixed precision training*, 2018, [Online]. Available: https://arxiv.org/abs/1710.03740. arXiv: 1710.03740.

[6] V. Hassija, V. Chamola, A. Mahapatra, A. Singal, D. Goel, K. Huang, S. Scardapane, I. Spinelli, M. Mahmud, and A. Hussain, "Interpreting black-box models: a review on explainable artificial intelligence," *Cognitive Computation*, vol. 16, no. 1, pp. 45–74, 2023, doi: 10.1007/s12559-023-10179-8.

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems*, ser. NIPS'17, Long Beach, California, USA: Curran Associates Inc., 2017, 6000–6010, [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa–Paper.pdf.

[8] Z. Sadeghi, R. Alizadehsani, M. A. CIFCI, S. Kausar, R. Rehman, P. Mahanta, P. K. Bora, A. Almasri, R. S. Alkhawaldeh, S. Hussain, et al., "A review of explainable artificial intelligence in healthcare," *Computers and Electrical Engineering*, vol. 118, Art no. 109370, Aug. 2024, doi: 10.1016/j.compeleceng.2024.109370.

[9] D. Solanki, A. Thakkar, K. Patel, J. Sarda, and A. K. Bhoi, "A review on approaches and applications of natural language inference," in *Proceedings of Data Analytics and Management*. Springer Nature Singapore, 2025, pp. 441–454, doi: 10.1007/978-981-96-3372-2_31.

[10] V. Chakkarwar, S. Tamane, and A. Thombre, "A review on BERT and its implementation in various NLP tasks," in *Proceedings of the International Conference on Applications of Machine Intelligence and Data Analytics (ICAMIDA 2022)*. Atlantis Press International BV, 2023, pp. 112–121, doi: 10.2991/978-94-6463-136-4_12.

[11] M. Müller, M. Salathé, and P. E. Kummervold, "COVID-Twitter-BERT: a natural language processing model to analyse COVID-19 content on twitter," *Frontiers in Artificial Intelligence*, vol. 6, Art no. 1023281, Mar. 2023, doi: 10.3389/frai.2023.1023281.

[12] D. Yu, B. Yang, D. Liu, H. Wang, and S. Pan, "A survey on neural-symbolic learning systems," *Neural Networks*, vol. 166, pp. 105–126, Sep. 2023, doi: 10.1016/j.neunet.2023.06.028.

[13] A. Fahfouh, A. Benlahbib, J. Riffi, and H. Tairi, "USMBA-NLP at semeval-2024 task 2: safe biomedical natural language inference for clinical trials using bert," in *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, Association for Computational Linguistics, 2024, 432–436, doi: 10.18653/v1/2024.semeval–1.66.

[14] S. Renjit and S. M. Idicula, "A study of the state of the art approaches and datasets for multilingual natural language inference," *Neural Processing Letters*, vol. 56, no. 6, Art no. 243, 2024, doi: 10.1007/s11063-024-11673-2.

[15] R. Wang, Z. Gao, L. Zhang, S. Yue, and Z. Gao, "Empowering large language models to edge intelligence: a survey of edge efficient llms and techniques," *Computer Science Review*, vol. 57, Art no. 100755, Aug. 2025, doi: 10.1016/j.cosrev.2025.100755.

[16] U. Nawaz, M. Anees-ur-Rahaman, and Z. Saeed, "A review of neuro-symbolic AI integrating reasoning and learning for advanced cognitive systems," *Intelligent Systems with Applications*, vol. 26, Art no. 200541, Jun. 2025, doi: 10.1016/j.iswa.2025.200541.

[17] Y. Feng, X. Yang, X. Zhu, and M. Greenspan, "Neuro-symbolic natural logic with introspective revision for natural language inference," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 240–256, 2022, doi: 10.1162/tacl_a_00458.

[18] T. Chandrasekaran, S. Ramisetty, and M. R. Pulicharla, "Neurosymbolic AI: bridging neural networks and symbolic reasoning," *World Journal of Advanced Research and Reviews*, vol. 25, no. 1, pp. 2351–2373, 2025, doi: 10.30574/wjarr.2025.25.1.0287.

[19] R. Niu, Q. Wang, H. Kong, Q. Xing, Y. Chang, and P. S. Yu, "Learn to explain transformer via interpretation path by reinforcement learning," *Neural Networks*, vol. 188, Art no. 107496, Aug. 2025, doi: 10.1016/j.neunet.2025.107496.

[20] P. Li, H. Yu, W. Zhang, G. Xu, and X. Sun, "SA-NLI: a supervised attention based framework for natural language inference," *Neurocomputing*, vol. 407, pp. 72–82, Sep. 2020, doi: 10.1016/j.neucom.2020.03.092.

[21] N. M. Gardazi, A. Daud, M. K. Malik, A. Bukhari, T. Alsahfi, and B. Alshemaimri, "BERT applications in natural language processing: a review," *Artificial Intelligence Review*, vol. 58, no. 6, Art no. 166, 2025, doi: 10.1007/s10462-025-11162-5.

[22] J. Kabbara and J. Cheung, "Investigating the effect of pre-finetuning BERT models on NLI involving presuppositions," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, Association for Computational Linguistics, 2023, 10 482–10 494, doi: 10.18653/v1/2023.findings–emnlp.703.

[23] I. M. S. Putra, D. Siahaan, and A. Saikhu, "SNLI indo: a recognizing textual entailment dataset in indonesian derived from the stanford natural language inference dataset," *Data in Brief*, vol. 52, Art no. 109998, Feb. 2024, doi: 10.1016/j.dib.2023.109998.

[24] M. Yeung, E. Sala, C.-B. Schönlieb, and L. Rundo, "Unified focal loss: generalising dice and cross entropy-based losses to handle class imbalanced medical image segmentation," *Computerized Medical Imaging and Graphics*, vol. 95, Art no. 102026, Jan. 2022, doi: 10.1016/j.compmedimag.2021.102026.

[25] T. Dao, S. Ermon, D. Fu, C. Ré, and A. Rudra, "FlashAttention: fast and memory-efficient exact attention with io-awareness," in *Advances in Neural Information Processing Systems 35*, ser. NeurIPS 2022, Neural Information Processing Systems Foundation, Inc. (NeurIPS), 2022, 16 344–16 359, doi: 10.52202/068431–1189.