Read Between the Lines

Malak Khan, Miguel Betrán Menz, João Fernandes Bárbara

University of Amsterdam

https://github.com/JoaoBarbara5/nlp-logic-model

Background

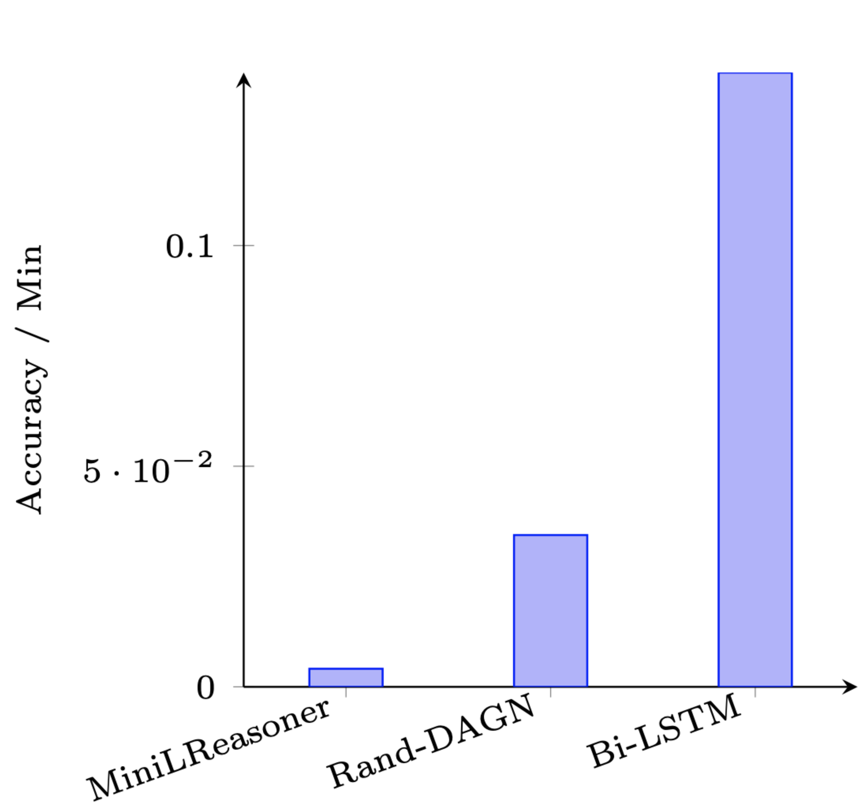
We address a reading comprehension task: pick the correct option among four given a *context* and *question*. Given **4638** training and **500** test samples in the **ReClor** format [1], we use **RoBERTa-Large** [2] fine-tuned on ReClor (context+question+option) as the reference baseline. We remove any ReClor questions overlapping with our assignment test set to avoid data leakage.

RQ & Hypotheses

Can any of the following architectures match performance of the reference model on our test set?

- **Method A (MiniLReasoner):** A minimal LReasoner [3] style model: a BERT [4] encoder produces a pooled `[CLS]` representation for each of 4 options which is scored by a single linear layer (one logit per option). Trained with multiple-choice cross-entropy; by default the BERT weights are **randomly initialized** (BERT config). **Total Parameters:** $\approx 109,480,000$.
- **Method B (Rand-DAGN):** A Discourse-Aware Graph Network (DAGN) [5] with **randomly initialized embeddings** (dim 300) trained end-to-end with cross-entropy loss. **Total Parameters:** $\approx 42,700,000$.
- **Method C (Bi-LSTM with Attention + Contextual Embeddings)):** Bidirectional Long Short-Term Memory with Attention Mechanism [6]. Iterations with both **randomly initialized embeddings** (dim 100) and **Word2Vec contextual embeddings** trained solely on the available training set [7]. End-2-end training with cross-entropy loss. **Total Parameters:** $\approx 1,680,000$.

We hypothesize that without contextual embeddings pretrained on large corpora and with a very small training set, models will fail to generalize well.



Experimental Setup

- **Method A (MiniLReasoner):** For each multiple choice option, we encode `context` paired with `question + [SEP] + option` (max length 256) and score the BERT pooled output using a single linear layer (one score per option). Training uses AdamW [8] with linear warmup scheduling [9] and gradient clipping.
- **Method B (Rand-DAGN):** A Discourse-Aware Graph Network [5] using [10] delimiters and punctuation marks. The pretrained backbone is replaced with: random embeddings ($d = 300$) as in Word2Vec/GloVe [11, 12] + Layer Normalization [13] + Projection to 1024 dim as in [2]. Tested with and without LayerNorm, with embedding variance = 1, and embedding variance = 0.2. Rest of the model is cited from [14].
- **Method C (Bi-LSTM w/ Att):** The input sequence is set as in A. Tokens are mapped to dense vectors using an embedding layer, with trainable weights. These weights are randomized in the first iteration and initialized with corpus-trained Word2Vec in the second. The model processes the input in both forward and backward directions to capture long-range dependencies [6]. An Attention mechanism computes a learnable weight for each time step, aggregating the LSTM hidden states into a context vector. This vector is passed through a linear classifier generate a scalar validity score.

Experimental Results

Performance comparisons on CV and held-out Test sets:

- **Method A Results:** Achieved a mean CV accuracy of **41.6%** and a final Test accuracy of **49.6%**.
- **Method B Results:** Training only converges **with** LayerNorm and with embedding variance = **0.2**. Else suffers from exploding logits. Performance equal for LR values 1e-4, 5e-6.
- **Method C Results:** Grid search identified the optimal config as *Emb Dim 100, Hid Dim 128, Drop 0.5, LR 0.001, Batch size 16* and the model achieved a final CV accuracy of **30.83%** and a Test accuracy of **34.8%**.
  With the additional **corpus-trained Word2Vec** the Test Accuracy was exatly the same **34.8%**.

Results Summary:

Model Architecture	Mean CV Acc.	Test Acc.
RoBERTa-L (ReClor)	–	**0.590**
Method A (MiniLReasoner)	0.416	**0.496**
Method B (Rand-DAGN)	–	0.344
Method C (Bi-LSTM)	0.308	0.348
Method C* (Bi-LSTM + Word2Vec)	-	0.348

Table 1:Compared cross-validation and test accuracies for each model

Model A, the most complex, presents the best results, predicting the correct answer close to 50% of the time. Model B and Model C perform slightly better than random, around 34% and 35% respectively; for Model C, both the initialized and non-initialized embeddings seem to have the same performance on the test set.

Analysis

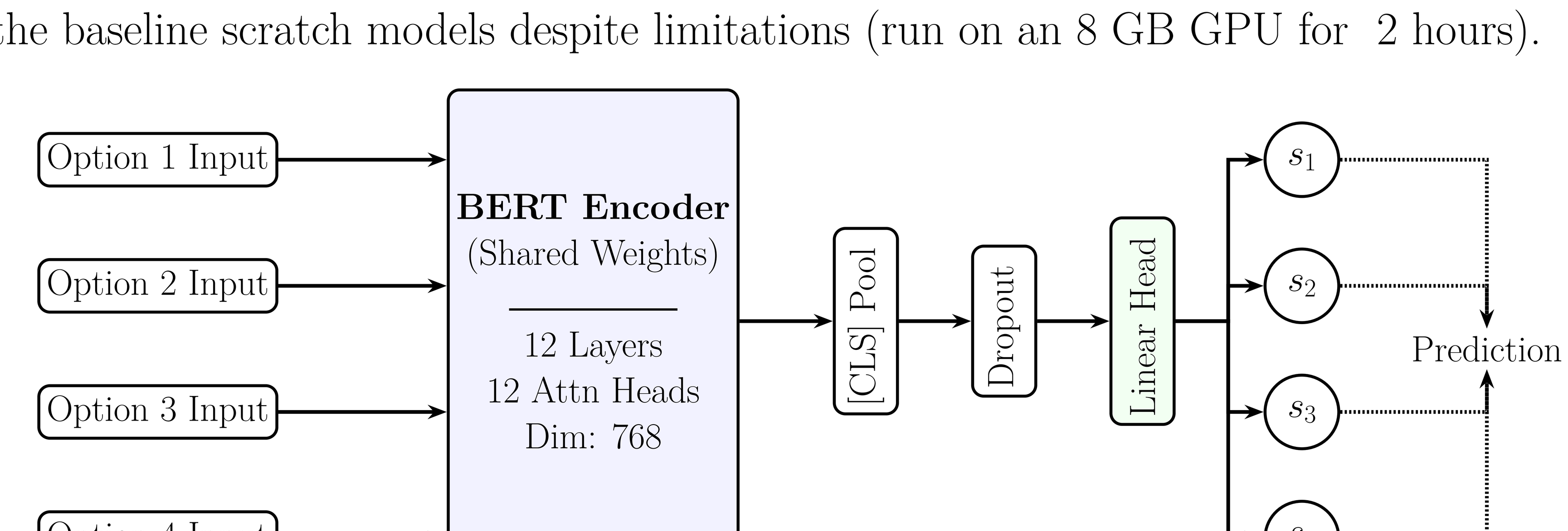
- The baseline dominates because it starts with contextual embeddings pretrained on huge corpora [2], whereas MiniLReasoner's random initialization forces it to learn basic grammar and logic simultaneously from a tiny dataset.
- MiniLReasoner succeeds because its standard Transformer architecture is optimizationally stable with AdamW [8] and learning rate 'warmup' [9], while the complex Graph Network of Method B suffers from 'exploding logits' and fails to converge without rigid constraints.
- MiniLReasoner's cross-encoder attention allows immediate, simultaneous comparison of context and option words, avoiding the 'memory bottleneck' inherent in the sequential processing in Bi-LSTM-based methods.
- In the Bi-LSTM w/ Attention, the introduction of domain-specific embeddings increased overfitting with training accuracies around 90% but similar test results. This happens because the initialization of a strong context embedder based on the training data does not help the model learn patterns that are generalizable to unseen data outside its learned vocabulary but rather it helps fit the existing data better.

Winning Model: MiniLReasoner

Architecture: The winning model, `MiniLReasoner`, uses a BERT encoder [4] with a single linear head. For each example with C choices, inputs are tokenized as **CLS**, context, **SEP**, question, **SEP**, option, **SEP**], flattened to a batch of size $B \times C$, and passed through BERT. The pooled `[CLS]` output is dropout-regularized and projected to a scalar score per option; the resulting logits (reshaped to (B, C)) are trained with multiple-choice cross-entropy.

Parameterization: Under the default `bert-base-uncased` configuration (12 Transformer layers, hidden size 768, intermediate size 3072, vocab size 30,522, and learned positional & token type embeddings) [4], the encoder contains approximately 109.48 million parameters. The classification head adds $768 \times 1 + 1 = 769$ parameters.

Initialization and Training: By default, `MiniLReasoner` loads a BERT config and randomly initializes weights. Training uses AdamW [8] with linear warmup [9] and gradient clipping, with 5-fold stratified CV. The model had a CV dev accuracy averaging approximately **41.6%** (per-folds: 0.4397, 0.4332, 0.3588, 0.4261, 0.4196), outperforming the baseline scratch models despite limitations (run on an 8 GB GPU for 2 hours).



References

[1] Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. ReClor: A Reading Comprehension Dataset Requiring Logical Reasoning, August 2020. arXiv:2002.04326 [cs].

[2] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach, July 2019. arXiv:1907.11692 [cs].

[3] Siyuan Wang, Wanjun Zhong, Duyu Tang, Zhongyu Wei, Zhihao Fan, Daxin Jiang, Ming Zhou, and Nan Duan. Logic-Driven Context Extension and Data Augmentation for Logical Reasoning of Text, May 2021. arXiv:2105.03659 [cs].

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *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)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[5] Yinya Huang, Meng Fang, Yu Cao, Liwei Wang, and Xiaodan Liang. DAGN: Discourse-Aware Graph Network for Logical Reasoning, April 2021. arXiv:2103.14349 [cs].

[6] Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. Attention-based bidirectional long short-term memory networks for relation classification. In Katrin Erk and Noah A. Smith, editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 207–212, Berlin, Germany, August 2016. Association for Computational Linguistics.

[7] Vincent Major, Alisa Surkis, and Yindalon Aphinyanaphongs. Utility of general and specific word embeddings for classifying translational stages of research, 2018.

[8] Ilya Loshchilov and Frank Hutter. Decoupled Weight Decay Regularization, January 2019. arXiv:1711.05101 [cs].

[9] Dayal Singh Kalra and Maissam Barkeshli. Why warmup the learning rate? underlying mechanisms and improvements, 2024.

[10] Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind Joshi, and Bonnie Webber. The Penn Discourse TreeBank 2.0.

[11] Jeffrey Pennington, Richard Socher, and Christopher Manning. GloVe: Global Vectors for Word Representation. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar, October 2014. Association for Computational Linguistics.

[12] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space, September 2013. arXiv:1301.3781 [cs].

[13] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer Normalization, July 2016. arXiv:1607.06450 [stat].

[14] DAGN: Official implementation for the NAACL'21 short paper DAGN: Discourse-Aware Graph Network for Logical Reasoning.