Applying Natural Language Processing in Answering Multiple-Choice Questions for Assessing Child/Youth and Adults Needs and Strengths (CANS©/ANSA)

Stanford CS224N Custom Project

Bohdan Junior
bohdanjr@stanford.edu

Joshua Adams
adamsdsc@stanford.edu

Kalikant Jha
kalikant@stanford.edu

Abstract

In this paper, we propose an NLP-based Multi-Choice Machine Reading Comprehension (MRC) solution which can fill out the CANS© assessment form by reading vignettes of patients with a semi-supervised approach. The solution comprises (1) a self-supervised encoder based Pre-trained Language Model (PrLM), such as `bert-base-uncased` (Devlin et al., 2018) and its other flavors, along with a fine-tuning process using CANS© dataset, and (2) a matching network, such as DCMN (Dual co-matching network) which was implemented by Zhang et al. (2020) and supposedly capture the relationships among the triplets of passage (P), question (Q), and answering options (O). In our baseline model we implemented and tested part (1) of the solution, achieving 62% during training, and 59% during testing. We also implemented the complete solution (1) + (2), with DCMN and DUMA (Zhu et al., 2022) matching networks. We achieved 73.0% / 60.0% and 74.0% / 72.0% for training/testing sets, respectively. Nevertheless, our results demonstrated that the best results were achieved with the PrLM Albert (Lan et al., 2019) `albert-xxlarge`, without a matching network: 78.0% / 72.0% for training/testing. Based on these results, we performed a qualitative analysis to understand some reasons for such a performance, and other alternatives that can improve our training results.

1 Key Information to include

- TA mentor: David Huang; External collaborator: Joshua Adams

2 Introduction

The World Health Organization (WHO) states that mental health is an essential component of a healthy human life (Gabín, J.; Pérez, A.; Parapar, J, 2021). In this paper, we describe a Natural Language Processing (NLP) solution that can have a positive impact on mental health. The CANS© /ANSA assessment is a comprehensive mental health evaluation tool consisting of a multi-category four-option questionnaire (Lyons, 2009). This assessment is comprised of fifty question sets and serves as an efficient communication mapping tool, with a reliability score above 0.90 with real-world case scenarios answered in a 'realistic manner' mimicking real-responses (Lyons, 2009), where a score of 1.00 means no errors. Each question is relevant for service planning, focused on the child/youth, and indicates if an action is required. Additionally, the exam monitors outcomes, i.e., if question set responses rated '2'or '3' moved to rating '0' or '1' over the treatment time.

The NLP solution here was designed to complete the exam based on machine reading comprehension (MRC) to understand a passage of text and then answer the questions naturally $(P, Q) \rightarrow A$. Natural and accurate responses to questions, in general, have been an active and challenging problem not entirely solved yet. There have been various benchmarking datasets, such as RACE (Lai et al., 2017) and SQuAD (Sun et al., 2019a), as well as models, such as DrQA (Chen et al., 2017), BERT (Devlin et al., 2018), DCMN (Zhang et al., 2020) and DUMA (Zhu et al., 2022) used as attempts to solve the problem.

For a specific MRC task, it can be classified as generative or selective, according to Baradaran et al. (2020). In generative tasks, the model will generate answers not limited to the spans of the passage. In selective tasks, on the other hand, the model will select the best alternative given candidate answers.

The focus of this paper is a selective task, Multi-choice MRC. Table 6 shows one example of the CANS© dataset, whose task is to select the best answer option among four options given a particular tuple of passage and question. The NLP solution designed can complete the four-option CANS© questionnaire, based on a patient vignette, with 70% accuracy. Relevant to this, that training and certification with a reliability of at least 0.70 on a test case vignette is required for certification and compliance in ethical use. A reliability equals to 1.00 means no wrong answer.

The core method for solving the selective MRC problem is based on a two-level hierarchical process, 1) representation encoding, which is done by an encoder PrLM, such as `bert-base-uncased` (Devlin et al., 2018) and its other flavors, and 2) capturing the relationship among the triplet of passage, question and answer, which has to be handled by a matching network such as OCN (Ran et al., 2019a), DCMN (Zhang et al., 2020) and DUMA (Zhu et al., 2022). With the advancements of PrLMs, design of efficient matching networks become more difficult. Table 1 shows that newer PrLMs such as ALBERT (Lan et al., 2019) is efficient even without a matching network.

Table 1: Performance of several models on RACE Dataset sorted by releasing time

	without matching	OCN	DCMN	DUMA
$BERT_{base}$	65.0*	66.8**	67.0*	—
$BERT_{large}$	72.0*	71.7**	75.4*	—
$ALBERT_{xxlarge}$	86.1***	—	85.7***	88.0***

*(Zhang et al., 2020); **(Ran et al., 2019a); ***(Zhu et al., 2022)

We established a baseline model focusing on representation encoding only using different PrLMs. Then reproduced the matching network using as references DCMN (Zhang et al., 2020) and DUMA (Zhu et al., 2022). In the DCMN approach, the matching module calculated attentions scores and representations from encoded passage-question, passage-option, and question-option pairwise relationships bidirectionally for each triplet $\{P, Q, A\}$, totaling six combinations, exploiting the gated mechanism to fuse the representations from two directions.

In the DUMA approach, based on multi-head attention, the matching module calculated attention representations using encoded passage tokens H^P as query and encode question-option H^{QO} as key-value, and then H^{QO} as query and H^P as key-value (bi-directional way, totaling two combinations), to finally fuse both representations. Intuitively two combinations (DUMA) is less complex than six (DCMN), and the use of multi-head (DUMA), i.e., multiple blocks for many encoded words simultaneously seems to indicate a simpler and better approach. It is possible to observe that, when both models are pre-trained with ALBERT, only DUMA surpasses the basic pre-trained model by 2 percentage points(see table 1).

However, in our experiment, we observed that ALBERT surpassed DUMA performance during training, 78% over 74%, and both of them achieved the same performance in testing, i.e., 72.0%. It showed us that matching networks could achieve same results, and eventually better ones by a small amount, but, to our understanding, this depends on the size of the dataset. This result also motivated us to look at other alternatives than matching networks, e.g, perform qualitative and error analysis to further identify potential better ways to improve the quantitative results.

3 Related Work

Reading comprehension can be seen as an important test bed for evaluating how well computer systems understand human language. Lehnert, in 1977, as stated by Manning et al., (2023) that *'the ability to answer questions is the strongest possible demonstration of understanding'* (Manning et al., 2023). Furthering research in challenging computer systems to understand human language, Chen et al. (2017) demonstrated a minimal, highly successful architecture for machine reading comprehension (MRC) and question answering, which became the Stanford Attentive Reader. In this model, the passage's paragraphs token p_i are represented as feature vectors $\tilde{p_i} \in \mathbb{R}^d$, and encoded via a bidirectional LSTM becoming. The question is also encoded, and the model predicts, at paragraph level, the span of tokens (start and end) that is most likely the correct answer. It achieved 79.4% F1 scores on the test set, matching the top performance SQuAD 1.1 leaderboard at the time of this research.

A large-scale deep PrLM, Bert, has led the field of NLP to a new stage. It introduced a bi-directional transformer encoder pre-trained on large amounts of text (Wikipedia + BookCorpus), and it was based on two training objectives: masked language model (MLM) and next sentence prediction (NSP). `bert-base` has 12 layers, 768 hidden-dim and 110M parameters, `bert-large` has 24 layers, 1024 hidden-dim and 330M parameters. Since initial Bert was released, other PrLMs were proposed, such as Robustly Optimized BERT Pre-training approach, RoBERTa (Liu et al., 2019)), which applies many techniques for better training, and Lite Bert ALBERT (Lan et al., 2019)), which proposes to share parameters among the modeling layers. While `roberta-base` has 12 layers, 768 hidden-dim and 123M parameters, `albert-xxlarge` has 12 layers, 4096 hidden-dim and 223M parameters.

BERT was applied to the MRC to solve open questions where the answer was a span of the passage. The question-passage was encoded by the model following the template: *[CLS] Question tokens [SEP] Passage tokens ...*, and a cross-entropy loss function was used to train the model and highlight, based on the probability assigned to each token, and predict the start and end token indexes in the passage to answer the question. In 2018, Google's Bert achieved 91.8% F1 on SQuAD 1.1. leaderboard, surpassing human performance, 91.2%.

Despite the great success of a PrLM as Bert, it has been shown that the ability to solve MRC problems could be further improved with a well-designed matching network. For a selective Multi-Choice MRC task, RACE, a collection of approximately 28,000 passages and 100,000 questions from middle and high school English exams, was used as a benchmark dataset. It is worth noting that RACE is a four-option questionnaire that contains five category types of questions (Sun et al., 2019b), which can increase the challenge of the MRC task: 1) detail (facts and details), 2) inference (reasoning ability), 3) main (main idea or purpose of a document), 4) attitude (author's attitude toward a topic or tone/source of a document), and 5) vocabulary (vocabulary questions). DCMN (Zhang et al., 2020) and DUMA (Zhu et al., 2022) proposed matching networks, using attention concepts, to relate the triplet $\{P, Q, A\}$ obtaining better results, by a small margin, than plain PrLMs (see Table 1). Besides adding a matching module, different authors proposed methods to improve results, e.g., relating and/or excluding answering options. Ran et al. proposed a method to model relationships and interactions among answer options to the benefit of distinguishing them (Ran et al., 2019b). Kim and Fung (2020) integrated a model that learns to select the wrong answer. Zhang et al. (2020) considered interactions among options to select the best one. This last author also proposed to select the more important paragraphs from the passage to improve the matching representations.

The models in DCMN, and especially in DUMA, solved Multi-Choice MRC exams by means of a PrLM fine-tuned with the RACE dataset, and a matching network. Gabín, Pérez, and Perapa applied almost the same reasoning, without the matching network, to complete a multi-choice depression severity questionnaire, made up by twenty-one questions (Gabín, J.; Pérez, A.; Parapar, J, 2021). One extra step was necessary: after fine-tuning with RACE, they repeated with a very small dataset, eRisk 2019 (Losada et al., 2019), made up by social networking posts from twenty users. They achieved similar results of best models on the overall depression level, but not on the individual questions. It exemplifies that a model, with great performance in one dataset, does not automatically generalize well to another one.

It is also important to note that although some models already surpassed human performance on SQuAD leaderboard, in terms of accuracy, it can't be assumed that MRC is solved yet. Ribeiro et al. (2020) concluded that although useful, accuracy on benchmarks is not sufficient for evaluating NLP models. For example, he shows that `bert-large`, although can achieve good accuracy levels on MRC tasks, often fails to properly grasp intensity modifiers and comparisons/superlatives. In the passage-question: 'Anna is worried about the project. Matthew is extremely worried about the project', 'Who is least worried about the project?', the failure rate of the system is 91.3%. Manning (2011) indeed did a salient analysis, concluding that at a certain level improvements in NLP, which can be extended to MRC, do not come from machine learning alone, but from linguistic resources and associated error analysis.

Current related work indicates that a well-chosen PrLM seems to drive the path forward to solve a Multi-Choice MRC task. As some authors advocated the use of a matching network to improve the results, we experimented with it however with small gains. Most importantly, we replicated multiple authors' suggestions in analyzing the problem from the linguistic perspective in the mental health assessment scenario. This was essential for us to understand the successes and errors of our model, as well as to identify alternatives to improve the model's performance.

4 Approach

4.1 Baseline

The experimented PrLM models were: (1) `bert-base-uncased` (Devlin et al., 2018) with $BertForMultipleChoice$, (2) `bio_clinicalBERT` (Alsentzer et al., 2019) with $SequenceClassification$ and (3) `distilbert-base-uncased` (Sanh et al., 2019) with $SequenceClassification$. Bert-base-uncased was first, fine-tuned with CANS© dataset. For pre-processing, each word of the dataset (5.1) was tokenized and mapped to indexes of a vocabulary (around 30,000 words). Texts were lowercased and tokenized using $WordPiece$ technique. The bert-base-uncased model follows a masking procedure in which 15% of the tokens are masked. In 80% of cases, the masked tokens are replaced with [MASK]. In 10% of cases, the masked tokens are substituted with a different random token from the original. The remaining 10% of cases leave the masked tokens unchanged. Then manually converted and truncated to a feature of a Bert expected template: $[CLS]$ passage $[SEP]$ question + option $[SEP]$. This formed the embedding, along with masks (equal to one for all tokens), segment ids (equal to 0 for passage and 1 to question + option), correct answer question, passage, and option lengths. After conversion to torch tensors, they were encoded by the Bert model, and a linear layer was applied to the pooled_output to calculate the logits. These logits were reshaped to the number of options(four), indicating the final predicted answer and allowing the calculation of the cross entropy loss, using AdamW as the optimizer for training. We observed that `bio_clinicalBERT` model gave better results compared to vanilla base BERT because it has better medical language representation. We implemented the baseline code from scratch and located at github (Kalikant Jha, 2023). The used references were: Huggingface, DCMN code (Qzsl123, 2020) and bert_race code (Kegang Xu, 2019). The process is depicted in Fig. 1.

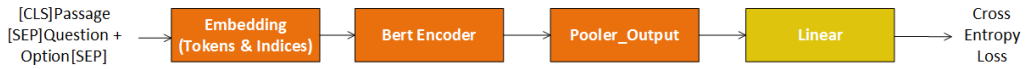

Figure 1: The architecture of baseline model.

We observed that Bert has a max-size-length limitation of 512 tokens. To overcome that, we experimented with tweaking the model a little bit, with our implementation, dividing the passage into two parts. First, we divided passages into two equal sizes, and then we encoded both and combined the pooled_output of both, applying a linear layer trying max and mean combinations before softmax. Other than this approach, we also tried eliminating tokens at the start, and end by truncating the passages to accommodate only 512 words. We achieve better results when truncating the end of passages.

4.2 Dual Co-Match Network (DCMN)

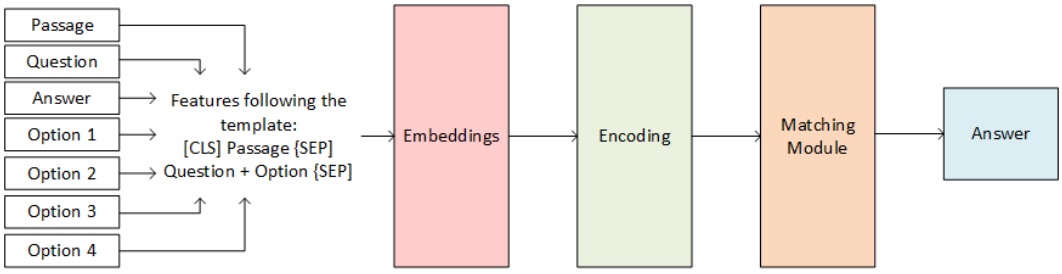

Figure 2: High-level DCMN and DUMA architectures.

We kept the original DCMN architecture, done by Zhang et al. (2020), which is depicted in Fig. 2, and implemented our solution using the original code as a reference (Qzsl123, 2020). We only modified the original code to load our dataset, which resembles the RACE template. Similar to the baseline, the first step was to encode the embeddings using a PrLM, the `base-large-uncased` in our experiment.

Next, the encoded embeddings were fed into a matching module to better relate encoded passage $H^P \in \mathbb{R}^{|P| \times d}$, question $H^Q \in \mathbb{R}^{|Q| \times d}$ and answering options $H^{O_i} \in \mathbb{R}^{|O_i| \times d}$, with $i \in \{1, 2, 3, 4\}$,

the last layer outputs by the encoder (see Fig. Fig. 6). Considering the passage-option relationship, we had two bidirectional alternatives:

$$G^{PO_i} = softmax(H^P W_1 {H^{O_i}}^\top) \in \mathbb{R}^{|P|\times|O_i|} \; and \; G^{O_i P} = softmax(H^{O_i} W_2 {H^P}^\top) \in \mathbb{R}^{|O_i|\times|P|}$$

with learnable parameters $W_1, W_2 \in \mathbb{R}^{d\times d}$, similar to calculating attention scores and distribution using multiplicative attention to relate the two alternatives. Then:

$$E^P = G^{PO_i} H^P \in \mathbb{R}^{|P|\times d}, E^{O_i} = G^{O_i P} H^{O_i} \in \mathbb{R}^{|O_i|\times d}$$

i.e., multiplication of previous *attention distribution* by the original encoded values, and:

$$S^P = ReLU(E^P W_3), S^{O_i} = ReLU(E^{O_i} W_4)$$

application of a ReLU layer to add non-linearity to the model, with $W_3, W_4 \in \mathbb{R}^{d\times d}$ as learnable parameters. $S^{P\_O_i}$ and $S^{O_i\_P}$ applied a row-wise max pooling layer in S_P and S_{O_i}, respectively, to capture most important activation. And:

$$g = \sigma(S^{P\_O_i} W_5 + S^{O_i\_P} W_6 + b)$$

with $W_5, W_6 \in \mathbb{R}^{d\times d}$ and $b \in \mathbb{R}^d$ as learnable parameters. This variable g worked as a gate to weigh the influence of each pair-wise representation. Then the equation to match representations from opposite directions were:

$$M^{P\_O_i} = gS^{P\_O_i} + (1-g)S^{O_i\_P} \in \mathbb{R}^d$$

The same reasoning was applied to passage-question and question-option pairs. Finally, we concatenated $M^{P\_O_i}$, M^{P_Q} and $M^{O_i\_Q}$ into $C_i \in \mathbb{R}^{3d}$ for each option, i.e., $i = \{1, 2, 3, 4\}$, applied a linear layer, making it possible to get a prediction and to use a cross-entropy loss to train the model.

4.3 DUal Multi-head co-Attention (DUMA)

We kept the original DUMA architecture, done by Zhu et al. (2022), similar to DCMN one (see Fig. 2). The differences between the two were the PrLM, `albert-xxlarge` in DUMA, and the matching network. It is important to clarify that we implemented our solution using the original code as a reference (pfZhu, 2022), only modifying it to load our dataset, which resembles the RACE template.

Then the encoded embeddings were fed into the matching module, being $H^P \in \mathbb{R}^{|P|\times d}$ the encoded passage and $H^{QO} \in \mathbb{R}^{|QO|\times d}$ the encoded question-option. This matching module reused the architecture of multi-head attention, and the attention representations were calculated in a bi-directional way, i.e., 1) H^P as *Query*, H^{QO} as *Key* and *Value*; and 2) H^{QO} as *Query*, H^P as *Key* and *Value* (see Fig. 7). The used equation for multi-head attention one MHA_1 is:

$$output_\ell = softmax(\frac{Query(Key)^\top}{\sqrt{d/h}})Value = softmax(\frac{H^P W_\ell^Q (H^{QO} W_\ell^K)^\top}{\sqrt{d/h}}) H^{QO} W_\ell^V$$

with $\ell \in \{1, 2, ..., h\}$. It is important to observe that $W_\ell^Q \in \mathbb{R}^{d\times d}$, $W_\ell^K \in \mathbb{R}^{d\times d}$ and $W_\ell^V \in \mathbb{R}^{d\times d}$ were added by the authors as learnable parameters. The final of all heads were combined by $output = [output_1, ..., output_h]Y$, with $Y \in \mathbb{R}^{d\times d}$. In a similar way, the MHA_2 is calculated, and we can express the final result by:

$$\text{DUMA}(H^P, H^{QO}) = \text{Fuse}(MHA_1, MHA_2)$$

The Fuse function first used mean pooling to pool the sequence outputs of both multi-head attentions, and then aggregated the two pooled outputs by element-wise multiplication. The model took the outputs of DUMA and computed the predicted answering option, and it was possible to use a cross-entropy loss to train the model.

5 Experiments

5.1 Data

For training, we obtained openly accessible data from pre-trained models: `bert-base-uncased`, `bert-large-uncased` (Devlin et al., 2018), `bio_clinicalBERT` (Alsentzer et al., 2019), `distilbert-base-uncased` (Sanh et al., 2019) and `albert-xxlarge` (Lan et al., 2019) for pre-training. Our data for analysis, the CANS© dataset comprised of fully proprietary realistically completed exams: 18 CANS© (15 with 50 questions and 3 with 40 questions) and 6 ANSA (49 questions), totaling 1164 examples. These samples are not patient-specific data, but rather proprietary

developed and answered based on real-world scenarios, so an IRB approval is not needed. It is also important to mention that we added four entire completed questionnaires after baseline runs. We pre-processed and transformed them into JSON/CSV files to feed into PrLM models. Each example has an id, a passage/vignette, a question, an answer, and an answering option (see. Fig. 8). We observed that the data is skewed (fewer counts for option D) hence we performed resampling using the class-balancing technique to mitigate the biased behavior of the model. The distribution of correct answers in training set is $\{A : 597(51.29\%); B : 165(14.17\%); C : 255(21.91\%); D : 147(12.63\%); \}$.

The average number of words in a passage before encoding is $\{P : 621.52\}$, the average number of words in a question is $\{Q : 26.71\}$, and the average number of words in answer options is $\{O : 63.82\}$. The total average number of words in a text is approximately 712.06, which is calculated by adding the average passage size, average question size, and average options size together $\{P + Q + O : 712.06\}$.

5.2 Evaluation method

In our model, we have used two major approaches for evaluation 1) performed train-dev-test split of (0.8-0.1-0.1) with random sampling for training and sequence sampling for testing, and 2) we trained using 20 entire exams from CANS© dataset (see Section 5.1). Dev and test sets were composed of 2 entire exams each, and sequence sampling was chosen because correct positions of item predictions were needed for analysis. For the gold exam set accuracy calculated percentage of correctly answered questions using following formula:

$$accuracy = \frac{\textit{number of correct answers}}{\textit{number of total questions}}$$

We also calculated precision, recall, and F1-score. The precision metric measures the accuracy of the positive predictions made by the model, using the formula: precision $= \frac{TP}{TP+FP}$, where TP are true positives and FP are false positives. The recall metric measures the model's ability to correctly identify positive cases, and is calculated as recall $= \frac{TP}{TP+FN}$, where FN are false negatives. The F1-score is a balanced metric that takes into account both precision and recall and is calculated as F1-score $= 2\frac{precision \cdot recall}{precision+recall}$.

We considered a qualitative evaluation, based on the fact that CANS© can be used to monitor outcomes, i.e., patients can be monitored over time by the results of the assessment, using both item or dimension scores. The dimensions scores can be generated by summing up individual items within six specific domains: Behavioral Emotional Needs, Life Domain Functioning, Risk Behaviors, Cultural Factors, Strenght Domain, and Caregiver Resources and Needs (see Table 7 and Table 5).

5.3 Experimental details

Table 2: Details about experiments until reaching best accuracies

Model	Hyperparameters	GPU	Time	Obs
Baseline	$batch_{SIZE} = 8$, $gradient_{ACC\_STEPS} = 8$, $*\alpha = 5.0 \times 10^{-6}$	7.2 GB	70 min for each run	10 epochs in 8 attempts using Colab
DCMN	$batch_{SIZE} = 4$, $gradient_{ACC\_STEPS} = 2$, $*\alpha = 1.0 \times 10^{-5}$	**23.4 GB	120 min for each run	20 epochs in 12 attempts using Colab
DUMA	$batch_{SIZE} = 2$, $gradient_{ACC\_STEPS} = 1$, $*\alpha = 3.0 \times 10^{-6}$	**36.7 GB	600 min for each run	20 epochs in 5 attempts using Colab
Albert	$batch_{SIZE} = 2$, $gradient_{ACC\_STEPS} = 1$, $*\alpha = 5.0 \times 10^{-6}$	**39.1 GB	600 min for each run	20 epochs in 4 attempts using Colab

$*\alpha \in \{1.0 \times 10^{-4}, 2.0 \times 10^{-5}, 1.0 \times 10^{-5}, 5.0 \times 10^{-6}, 3.0 \times 10^{-6}\}$;
**Not possible to run on AWS due to 24GB GPU memory limit

5.4 Results

The initial expectation was to obtain random answers (around 50% choosing A option - see 5.1), but we achieved 62%/59% (dev/test) in the baseline (see Table 8). It is possible to observe the loss and accuracy learning curves for our baseline model in Fig 9. The final results can be observed in table 3.

We expected DUMA to be better than sole Albert, but Albert achieved 78%-72% accuracy on dev-test sets, being our best model. We believe improved results can be obtained with more computational budget, i.e., more GPU memory. We assume that because during training we observed that in Colab (40 GB) it was possible to double the batch size, whilst in AWS (24 GB) we could not. That gave us higher accuracies in DUMA. If it was possible to double batch size again, we assume better results would be achieved. Not only that, but we also believe that our dataset was small. We indeed observed that it did not generalize well to the test set, as it can be seen in the training curves (Fig. 10).

Table 3: *Final results (see 5.1 for distribution of correct answering options)*

Model	Training	#train	#dev-test*	Accuracy
DCMN `bert-large-uncased`	BertForMultipleChoiceWithMatch	964	100-100	73%-60%
DUMA `albert-xxlarge`	AlbertDUMAForMultipleChoice	964	100-100	74%-72%
ALBERT `albert-xxlarge`	AlbertForMultipleChoice	964	100-100	78%-72%

*For comparison analysis, we utilized the same test set for all models

Table 4: *Precision, Recall and F1-Score for the test results*

	DCMN			ALBERT			DUMA		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
A	0.80	0.77	0.78	0.88	0.80	0.84	0.78	0.92	0.84
B	0.08	0.10	0.09	0.23	0.30	0.26	0.25	0.10	0.14
C	0.35	0.32	0.33	0.57	0.63	0.60	0.64	0.37	0.47
D	0.55	0.60	0.57	0.80	0.80	0.80	0.60	0.80	0.70
Avg	0.61	0.60	0.61	0.74	0.72	0.73	0.68	0.72	0.68

6 Analysis

In order to perform our qualitative analysis, we used the same testing set, containing the same patient's vignettes across all the models.

6.1 Precision, Recall and F1-score

The results in table 4 indicated that there were very few false positive predictions for class A and class D across all models. We inferred that the model was able to correctly assess extremes, i.e., A and D classes, but comparatively did not succeed in predicting in-between values, i.e., B and C classes. More data to train and a solution for long texts could leverage F1-Scores achieved: 0.61 for DCMN, 0.73 for Albert, and 0.68 for DUMA, improving its class-wise predictive skills.

6.2 CANS domains

Table 5: Correct model predictions over CANS domains for the patients belonging to the test set

	DCMN		ALBERT		DUMA	
Domain	Patient 1	Patient 2	Patient 1	Patient 2	Patient 1	Patient 2
Behavioral / Emotional Needs	67%	44%	100%	67%	78%	78%
Caregiver Resources & Needs	80%	90%	60%	40%	50%	90%
Cultural Factors	100%	67%	100%	100%	67%	100%
Life Functioning	45%	45%	45%	27%	18%	55%
Risk Behaviors	13%	13%	38%	50%	50%	13%
Child Strengths	22%	11%	22%	22%	22%	22%

Table 5 indicates that for the *Cultural Factors* category, which had the highest values, we achieved 100% accuracy in most experiments. However, further examination revealed that there was minimal discussion about this topic in the passage, implying that the model may be inclined towards the most biased response (i.e. 0) in the training dataset. We also noticed that the model tended to predict

high ratings (e.g. 3) for suicidal behavior and depression in the *Risk Behaviors* and *Behavioral & Emotional Needs* categories, despite their preferred rating is very low (i.e. 0) and there is no mention in the passage about these topics. Further investigation revealed that the presence of related tokens in either questions, answers, or options has led the model to build some co-relation attention link to the respective class. In contrast to *Cultural Factors* category, the *Child Strengths* consistently received a very low score because the corresponding answers were typically at the end of the passage. As we used BERT as the base model for tokenization and attention mechanism, possibly truncating the token (P+Q+O) beyond 512 counts, resulted in insufficient syntactical or contextual information available for this category to answer the questions.

6.3 Types of Questions

Some questions were challenging to answer accurately because of the linguistic complexity of answers. Adequate inference needed to be applied and few questions were open-ended, such as the inquiry **"How does the extended family communicate with each other?"** To enable the model to respond to such questions, it must comprehend the entire context of the passage. However, due to the truncated passage being encoded into the model, the number of tokens available was insufficient for a comprehensive understanding. We noticed that the attention link between open-ended questions and answer was weak, kindly refer Fig 11 and 12, (Vig, 2019).

6.4 Albert

Sole Albert boosted DCMN performance in 4 p.p. (dev set) - due to a better pre-trained language representation. By dividing the vocabulary embedding matrix into two small ones, and allowing cross-layer parameter sharing, fewer parameters were required and faster training was achieved, which really indeed improved our results.

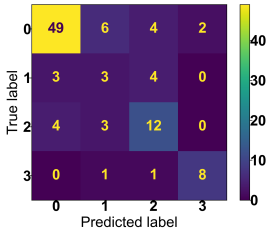

Figure 3: ALBERT

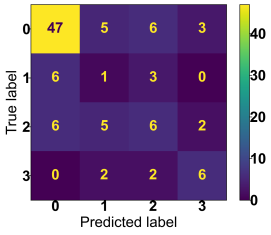

Figure 4: DCMN

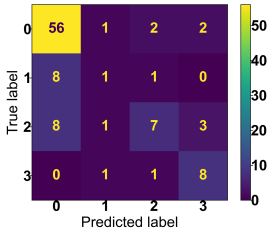

Figure 5: DUMA

7 Conclusion

We created a strong baseline model for a Multi-Choice MCR task from scratch using a PrLM. We tested different alternatives and considered we reached a baseline threshold due to our dataset size. With the aid of Dr. Lyons (2023) and his collaborative team, we received more data to work on the final models. And inspired by DCMN and DUMA matching modules, we tested more robust models, with bigger PrLMs, such as ALBERT, and matching modules. After running such models, we concluded that plain ALBERT was our best model, achieving 78%-72% accuracies in dev-set, respectively. From Fig. 3, it is possible to see the correct answering options, 0-3, in the diagonal. We consider that our architecture, based on BERT, as the foundation for token encoding and attention mechanisms, was fundamental to achieve such results. We must also stress out our disappointment with the matching modules, which should increase the final scores. We believe that such modules can indeed improve the model's accuracy, but with more training data, such as RACE, which is approximately 100 times bigger than ours. Although the gains were small, around 2 percentage points, previous authors demonstrated its value. In our case, these matching modules were able only to achieve same performance as a sole PrLM, i.e., 72% accuracy on test set. After these results with matching models, we invested some time to overcome a known limitation of BERT in handling longer paragraphs containing over 512 tokens, trying to tweak the models with simple approaches in order to achieve good results. It was necessary since our average combination of $\{P, Q, O\}$ triplet sequence tokens length was 712. To our surprise, the longest truncation approach, i.e., truncating the passages, which is a simple solution, gave us better results. Simplicity seems to be related to efficient and elegant solutions. For future work, we can consider increasing the computation budget, and we can also test other PrLM with different strategies, e.g., Chat-GPT, which instead of the Encoder approach of Bert uses a Decoder approach. Actually, some authors, such as Jiang et al. (2021), already applied to Multi-Question. It is worth the search since the most beneficiaries will be children, who will be able to treat with utmost care. Our code is hosted at Kalikant Jha (2023).

References

Emily Alsentzer, John R. Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, and Matthew B. A. McDermott. 2019. Publicly available clinical BERT embeddings. *CoRR*, abs/1904.03323.

Razieh Baradaran, Razieh Ghiasi, and Hossein Amirkhani. 2020. A survey on machine reading comprehension systems. *CoRR*, abs/2001.01582.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *CoRR*, abs/1704.00051.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Gabín, J.; Pérez, A.; Parapar, J. 2021. Multiple-choice question answering models for automatic depression severity estimation. *Eng. Proc*, 7(1):23–.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How Can We Know When Language Models Know? On the Calibration of Language Models for Question Answering. *Transactions of the Association for Computational Linguistics*, 9:962–977.

Joshua Adams Kalikant Jha, Bohdan Jr. 2023. Applying natural language processing in answering multiple-choice questions for assessing child/youth and adults needs and strengths - code. `https://github.com/kalikant/cs224n_project`.

Jungyoun Kim Kegang Xu, Jingjie Tin. 2019. A bert based model for multiple-choice reading comprehension - code. `https://github.com/tosmaster/bert-race/`.

Hyeondey Kim and Pascale Fung. 2020. Learning to classify the wrong answers for multiple choice question answering (student abstract). *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(10):13843–13844.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

David E. Losada, Fabio Crestani, and Javier Parapar. 2019. Overview of erisk 2019 early risk prediction on the internet. In *Experimental IR Meets Multilinguality, Multimodality, and Interaction*, pages 340–357, Cham. Springer International Publishing.

John S. Lyons. 2009. *Communimetrics: A Communication Theory of Measurement in Human Service Settings*, 1st edition. Springer New York, NY.

John S. Lyons. 2023. College of public health. `https://cph.uky.edu/people/john-lyons`.

Chris Manning, John Hewitt, Amelie Byun, and John Cho. 2023. Cs224n: Natural language processing with deep learning. `http://web.stanford.edu/class/cs224n/index.html#coursework`.

Christopher D. Manning. 2011. Part-of-speech tagging from 97linguistics? In *Proceedings of the 12th International Conference on Computational Linguistics and Intelligent Text Processing - Volume Part I*, CICLing'11, page 171–189, Berlin, Heidelberg. Springer-Verlag.

pfZhu. 2022. duma. `https://github.com/pfZhu/duma_code`.

Qzsl123. 2020. dcmn. `https://github.com/Qzsl123/dcmn`.

Qiu Ran, Peng Li, Weiwei Hu, and Jie Zhou. 2019a. Option comparison network for multiple-choice reading comprehension.

Qiu Ran, Peng Li, Weiwei Hu, and Jie Zhou. 2019b. Option comparison network for multiple-choice reading comprehension.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *ArXiv*, abs/1910.01108.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019a. How to fine-tune bert for text classification? In *China National Conference on Chinese Computational Linguistics*.

Kai Sun, Dian Yu, Dong Yu, and Claire Cardie. 2019b. Improving machine reading comprehension with general reading strategies. 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)*, pages 2633–2643, Minneapolis, Minnesota. Association for Computational Linguistics.

Jesse Vig. 2019. A multiscale visualization of attention in the transformer model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 37–42, Florence, Italy. Association for Computational Linguistics.

Shuailiang Zhang, Hai Zhao, Yuwei Wu, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. 2020. DCMN+: dual co-matching network for multi-choice reading comprehension. *The Thirty-Fourth AAAI Conference on Artificial Intelligence*, pages 9563–9570.

Pengfei Zhu, Zhuosheng Zhang, Hai Zhao, and Xiaoguang Li. 2022. Duma: Reading comprehension with transposition thinking. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:269–279.

A Appendix

Table 6: An example of CANS© dataset to assess the presence of limited cognitive capacity or developmental disabilities that challenges the caregiver's ability to parent (DEVELOPMENTAL).

Excerpt of Passage	Mike is a 15 year-old boy who is currently living with his grandparents. He is not in contact with his mother who has a serious substance dependence disorder. The identity of his father is not known.
Question	Does the caregiver have developmental challenges that make parenting/caring for the child/youth difficult?
Answer Options	0. No evidence of caregiver developmental disabilities or challenges. 1. Caregiver has developmental challenges. 2. Caregiver has developmental challenges that interfere with the capacity to parent the child/youth. 3. Caregiver has severe developmental challenges that make it impossible to parent the child/youth at this time.

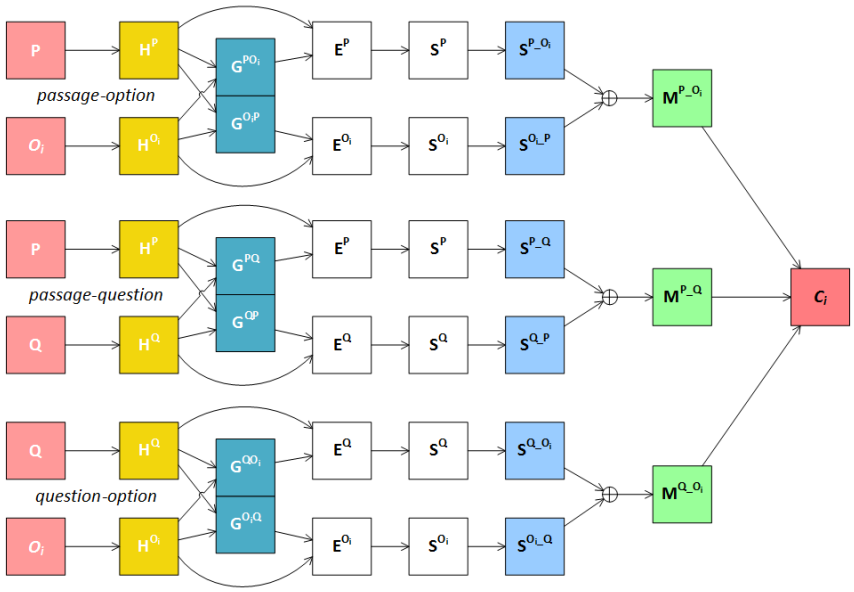

Figure 6: The architecture of the matching module, named dual co-matching network (DCMN).

Table 9: *Baseline Results (see 5.1 for distribution of correct answering options)*

Pre-trained Model	Fine-tuning Model	#train	#dev-test	Accuracy
bert-base-uncased	BertForMultipleChoice	794	100-100	62%-59%
Bio_ClinicalBERT	BertForSequenceClassification	795	199 (test)	57% (test)
distilbert-base-uncased	DistilBertForSequenceClassification	795	199 (test)	33% (test)

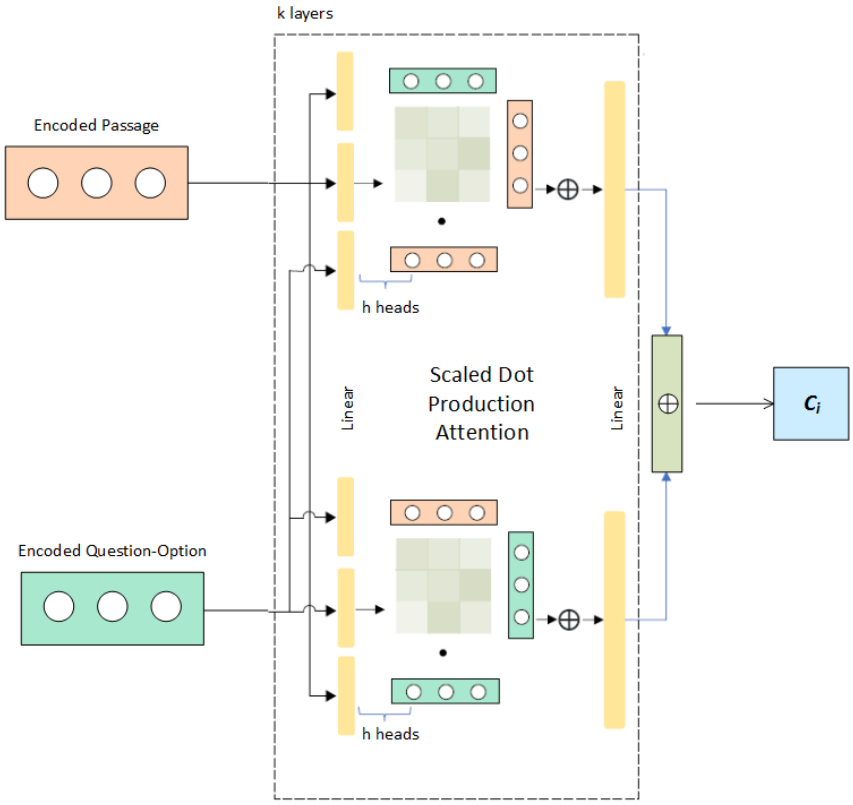

Figure 7: The architecture of the matching module, named dUal multi-head co-Attention (DUMA).

49. DEVELOPMENTAL

This item describes the presence of limited cognitive capacity or developmental disabilities that challenges the caregiver's ability to parent.

Questions to Consider	Ratings and Descriptions
• Does the caregiver have developmental challenges that make parenting/caring for the child/youth difficult?	0 No current need; no need for action or intervention. No evidence of caregiver developmental disabilities or challenges. Caregiver has no developmental needs.
	1 Identified need requires monitoring, watchful waiting, or preventive activities. Caregiver has developmental challenges. The developmental challenges do not currently interfere with parenting.
	2 Action or intervention is required to ensure that the identified need is addressed; need is interfering with functioning. Caregiver has developmental challenges that interfere with the capacity to parent the child/youth.
	3 Problems are dangerous or disabling; requires immediate and/or intensive action. Caregiver has severe developmental challenges that make it impossible to parent the child/youth at this time.

JSON
- answers
  - 0 : "A"
  - 1 : "A"
- options
  - 0
    - 0 : "No evidence of caregiver developmental disabilities or challenges."
    - 1 : "Caregiver has developmental challenges."
    - 2 : "Caregiver has developmental challenges that interfere with the capacity to parent the child/youth."
    - 3 : "Caregiver has severe developmental challenges that make it impossible to parent the child/youth at this time."
  - 1
    - 0 : "No evidence of safety issues."
    - 1 : "Household is safe but concerns exist about the safety of the child/youth due to history or others who might be abusive."
    - 2 : "Child/youth is in some danger from one or more individuals with access to the home."
    - 3 : "Child/youth is in immediate danger from one or more individuals with unsupervised access."
- questions
  - 0 : "Does the caregiver have developmental challenges that make parenting/caring for the child/youth difficult?"
  - 1 : "Is the caregiver able to protect the child/youth from harm in the home? Are there individuals living in the home or visiting the home that may be abusive to the child/youth?"
- article : "Mike is a 15 year-old boy who is currently living with his grandparents. He is not in contact with his mother who has a serious substance dependence disorder. The identity
- id : "cans015.txt"

Figure 8: On top it is an example of CANS© question, with four alternatives. On the bottom, it is possible to observe how the json format of input data is. Adapted with permission from Dr. Lyons (2023)

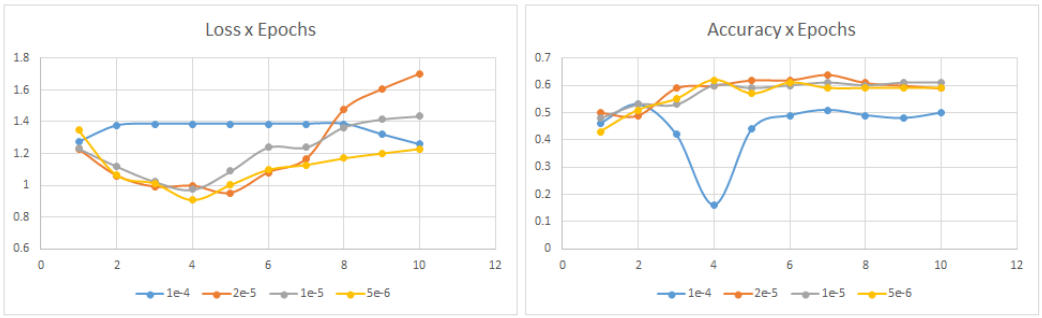

Figure 9: Best baseline model. After each epoch, the dev_loss was calculated (left) and the accuracy measured, both using a dev set, composed of two entire completed questionnaires.

Table 7: Domains of CANS©

Domain	CANS© Items
BEHAVIORAL EMOTIONAL NEEDS (9)	Psychosis, Impulsivity/Hyperactivity, Depression, Anxiety, Oppositional, Conduct, Anger Control, Substance Use, Adjustment to Trauma
LIFE DOMAIN FUNCTIONING (11)	Family Functioning, Living Situation, Social Functioning, Developmental/Intellectual, School Behavior, School Achievement, School Attendance, Medical/Physical, Sexual Development, Sleep
RISK BEHAVIORS (8)	Suicide Risk, Non-Suicidal Self-Injurious Behavior, Other Self-Harm, Danger to Others, Sexual Aggression, Delinquent Behavior, Runaway, Intentional Misbehavior
CULTURAL FACTORS (3)	Language, Traditions and Rituals, Cultural Stress
STRENGTHS DOMAIN (9)	Family Strengths, Interpersonal, Educational Setting, Talents/Interests, Spiritual/Religious, Cultural Identity, Community Life, Natural Supports, Resiliency
CAREGIVER RESOURCES AND NEEDS (10)	Supervision, Involvement with Care, Knowledge, Social Resources, Residential Stability, Medical/Physical, Mental Health, Substance Use, Developmental, Safety

Table 8: *Baseline Results (see 5.1 for distribution of correct answering options)*

Pre-trained Model	Fine-tuning Model	#train	#dev-test	Accuracy
bert-base-uncased	BertForMultipleChoice	794	100-100	62%-59%
Bio_ClinicalBERT	BertForSequenceClassification	795	199 (test)	57% (test)
distilbert-base-uncased	DistilBertForSequenceClassification	795	199 (test)	33% (test)

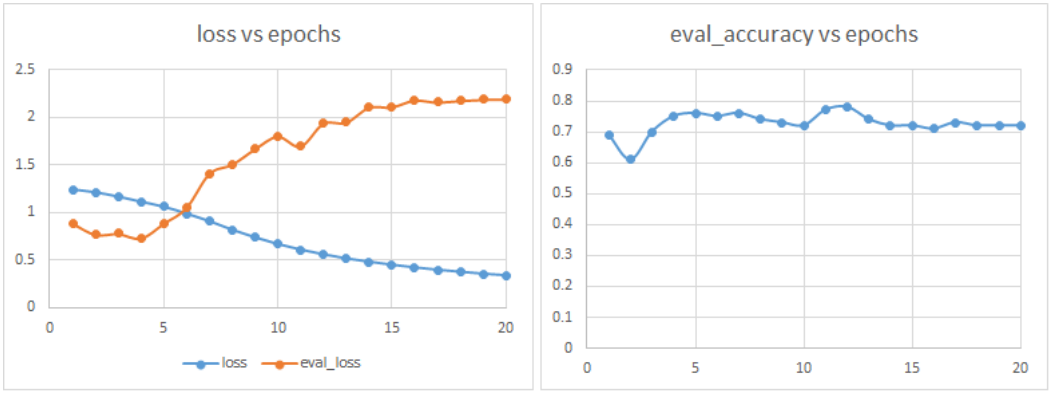

Figure 10: Final training for best model, Albert.

Figure 11: Patient 1

Figure 12: Patient 2