BloomNet: A Robust Transformer based model for Bloom's Learning Outcome Classification

Abdul Waheed$^{\alpha}$**, Muskan Goyal**$^{\alpha}$**, Nimisha Mittal**$^{\alpha}$**, Deepak Gupta**$^{\alpha}$**, Ashish Khanna**$^{\alpha}$**, Moolchand Sharma**$^{\alpha}$

$^{\alpha}$ Maharaja Agrasen Institute of Technology, New Delhi, India.

{abdulwaheed1513, goyalmuskan1508, nimishamittal1999}@gmail.com

{deepakgupta, ashishkhanna, moolchand}@mait.ac.in

Abstract

Bloom's taxonomy is a common paradigm for categorizing educational learning objectives into three learning levels: cognitive, affective, and psychomotor. For the optimization of educational programs, it is crucial to design course learning outcomes (CLOs) according to the different cognitive levels of Bloom's Taxonomy. Usually, administrators of the institutions manually complete the tedious work of mapping CLOs and examination questions to Bloom's taxonomy levels. To address this issue, we propose a transformer based model named BloomNet that captures linguistic as well semantic information to classify the course learning outcomes (CLOs) . We compare BloomNet with diverse set of basic as well as strong baselines and we observe that our model performs better than all the experimented baselines. Further, we also test the generalisation capability of BloomNet by evaluating it on different distributions which our model does not encounter during training and we observe that our model is less susceptible to distribution shift compared to the other considered models. We support our findings by performing extensive result analysis. In ablation study we observe that on explicitly encapsulating the linguistic information along with semantic information improves the model's IID (independent and identically distributed) performance as well as OOD (out-of-distribution) generalization capability. The open-sourced codebase including data can be found here: https://github.com/macabdul9/BloomNet.

1 Introduction

One of the most difficult challenges faced by the science educators is preparing a curriculum that facilitates the learning process in a structured, planned, and productive manner. It is the goal of the scientific curriculum to educate students who can investigate, question, participate in collaborative projects, and effectively communicate. The expected improvements for the students are articulated in a curriculum as learning outcomes (Zorluoğlu et al., 2019). Learning outcomes are used to track, measure, and evaluate the standards and quality of education received by the students at educational institutions (Attia, 2021). In terms of these learning outcomes, we may also identify the level of any student. Various measurement and evaluation studies are thus incorporated to determine the level of individual learning outcomes.

Exam evaluation is critical for determining how well students understand the course material. Therefore, the objectivity and scientific relevance of the questions developed for exams must be questioned in order to guarantee that students' learning outcomes are tracked and judged effectively. One of the relevant scientific techniques for analyzing this is the Bloom's Taxonomy (Anderson et al., 2000), which is well-known among the educators around the world. The examinations should take account of the difficulty levels, which correspond to the basic objectives and course outcomes in conventional ways like the Bloom's taxonomy.

Dr. Benjamin Bloom, an Educational Psychologist, developed the Bloom's Taxonomy in 1965. Its goal was to encourage high-order thinking, such as analyzing and examining instead of rote memorization of information (Adesoji, 2018). The Bloom's taxonomy is divided into three categories: cognitive (mental skills), affective (emotional areas or attitude), and psychomotor (physical skills). Our study focuses on the cognitive domain, which involves knowledge and intellectual skill development. Researchers have recently demonstrated a growing interest in automatic assessment based on cognitive domains in Bloom's Taxonomy. (Abduljabbar and Omar, 2015; Mohammed and Omar, 2018; Yahya, 2019). The majority of previous research focused on question classification from a specific domain, while Bloom's taxonomy across the multi-domain region is lacking ways for classifying questions (Sangodiah et al., 2017). This work therefore seeks to establish a question classification method based on the cognitive domain of Bloom's taxonomy. The Hierarchical order of levels in cognitive domain is: Knowledge, Comprehension, Application, Analysis, Synthesis and Evaluation. The first three levels are categorised as lower level of thinking, whilst the latter three levels are considered as high level of thinking.

The primary aim of this study is to assess the utility and efficacy of Bloom's Taxonomy as a framework for establishing course learning outcomes, optimizing curriculum, and evaluating various educational programs. In this paper, we propose BloomNet, a novel transformer-based model that incorporates both linguistic and semantic information for the classification of bloom's course learning outcomes. We also examine the generalisation capability of BloomNet on new distributions because train and test distributions are usually not distributed identically. The evaluation datasets rarely represent the entire distribution and the test distribution often drifts over time (Quionero-Candela et al., 2009), resulting in train-test discrepancies. Due to these discrepancies, models can face unexpected conditions at the test time. Therefore, models should be able to detect and generalise to out-of-distribution (OOD) examples.

In most NLP evaluations, the train and test samples are assumed to be independent and identically distributed (IID). Large pretrained transformer models can achieve high performance on a variety of tasks in the IID scenario (Wang et al., 2018). However, high IID accuracy does not always imply OOD robustness. Furthermore, because pretrained Transformers rely largely on false cues and annotation artifacts (Gururangan et al., 2018; Cai et al., 2017) that OOD instances are less likely to feature, their OOD robustness is unknown. Hence, we examine the robustness of BloomNet and other experimented models such as CNNs, LSTMs, pretrained transformers, and more.

The contributions of our research can be summarized as follows:

1. We propose a transformer-based model, BloomNet, that can distinguish between six different cognitive levels of Bloom's taxonomy (Knowledge, Comprehension, Application, Analysis, Synthesis and Evaluation).

2. We implement, train and evaluate multiple models to perform comparative analysis.

3. We evaluate experimented models for OOD generalization and we observe that pretrained transformers (RoBERTa, DistilRoBERTa (Liu et al., 2019; Sanh et al., 2019)) along with proposed model have better generalization capability compared to other models.

4. We perform ablation study to asses the contribution of various components in proposed model.

The following is the final exhibition. Section 2 and Section 3 describes the previous work and methodology respectively. Section 4 delves into the experiments and results. The conclusion and possible future directions are discussed in Section 5.

2 Related Work

Text classification is an important NLP research area with numerous applications. A number of scholars have concentrated on automatic text classification. In recent years, classification of exam questions for the cognitive domain of Bloom's taxonomy has received a lot of attention. Previous works have used different features and methods for text classification. Some of these works are discussed in this section.

In (Chang and Chung, 2009), an online examination system is created that supports automatic Bloom's taxonomy analysis for the test questions. The researchers introduce fourteen keywords for the analysis on questions. Each keyword is associated with a specific cognition level. The experiment is conducted on 288 test items and a 75% accuracy is achieved for the "Knowledge" cognition level.

A. Swart and M. Daneti (Swart and Daneti, 2019) analyzed the learning outcomes for Electronic fundamental module (of two universities) using Bloom's Taxonomy. To identify the proportion of each cognition level, the verbs of each learning outcome are connected to certain specific verbs in Bloom's taxonomy. This reflected the balance between theory and practice for the cognitive development of electrical engineering students. The consistency of the findings of the two universities demonstrated that students could blend theory and practice because they had around 40 percent of higher level cognitive outcomes.

Likewise, (Mohammed and Omar, 2020) classified exam questions for the cognitive domain of Bloom's Taxonomy using TFPOS-IDF and pretrained word2vec. To classify the questions, the extracted features are fed to three distinct classifiers i.e. logistic regression, K-nearest neighbour, and Support Vector Machine. For the experiment, they employ two datasets, one with 141 questions and the other with 600 questions. The first dataset results in a weighted average of 71.1%, 82.3% and 83.7% while the second achieves a weighted average of 85.4%, 89.4% and 89.7%.

Adidah Lajis et al. proposed (Lajis et al., 2018) a framework for assessing students' programming skills. Bloom's taxonomy cognitive domain serves as the foundation for the framework. According to the findings, Bloom's taxonomy could be used as a basis for grading students. It said that the students would be judged based on their ability using Bloom's taxonomy. The authors also suggested that taxonomy be used as an evaluation framework rather than learning.

Based on their domain knowledge, teachers and accreditation organizations manually classify course learning outcomes (CLOs) and questions on distinct levels of cognitive domain. This is time-consuming and usually leads in errors due to human bias. As a result, this technique must be automated. Several scholars have sought to automate this process through the use of natural language processing and machine learning techniques (Haris and Omar, 2012; Jayakodi et al., 2015; Osadi et al., 2017; Kowsari et al., 2019). Deep learning has recently exhibited impressive results when compared to traditional machine learning methods, particularly in the field of text classification (Minaee et al., 2020).

For text classification tasks, several neural models that automatically represent text as embedding have been developed, such as CNNs, RNNs, graph neural networks, and a variety of attention models such as hierarchical attention networks, self-attention networks, and so on. The majority of previous efforts on Bloom's taxonomy have either used traditional machine learning approaches or representative deep neural models such as RNNs, LSTMs, and so on. In this research, we propose a transformer-based approach for performing text classification as per cognitive domains. Transformers, (Vaswani et al., 2017) provide significantly better parallelization than RNNs, allowing for efficient (pre-)training of very large language models and an enhanced performance rate.

3 Methodology

In this section we discuss the methodology part of our research. Our model is inspired by (Gupta et al., 2021) and (Yang et al., 2016b). In BloomNet , we encapsulate the linguistic information along with generic input representation and we also explicitly model word level attention. In following sections we describe each component of our model (shown in Figure 1) in detail.

Notation: We denote current input as set of tokens $x \in X = \{t_0, t_1, t_2, ...t_n\}$ where n is the number of tokens in input. We define a model as $f_{\texttt{model}} : x \longrightarrow h$ where $h \in \mathbb{R}^d$. We define our final classfeir as $f_c : x \longrightarrow C$ where C is the softmax output and its size is equal to number of classes in our data.

3.1 Representation Model

Representation model or language encoder is main component of BloomNet which gives contextualized embeddings (Devlin et al., 2019; Pennington et al., 2014) for the text input. and for this we use pretrained RoBERTa (Liu et al., 2019) model from huggingface model hub repository (Wolf et al., 2020). The reason we use RoBERTa instead of its other widely used counterparts such as BERT (Devlin et al., 2019), DistilBERT (Sanh et al., 2019) is that it has seen much more data during its pretraining compare to its predecessor which results in increased robustness for subpopulation as well distribution shift. We feed tokenized input to RoBERTa model and we use `CLS` token as input representation. It can be represented as :

$$h_{\text{rep}} = f_{\text{rep}}(x) \tag{1}$$

3.2 Linguistic Encapsulation

Work by (Gupta et al., 2021) shows that explicit encapsulation of linguistic information increases the performance of the model for claim detection task, inspired by the same we also explicitly encapsulate linguistic information in modelling of BloomNet. We use POS (Part-Of-Speech) and NER (Named Entity Recognition) information coming from a trained model for POS and NER tasks respectively. We freeze the POS and NER model during training so that it's weights do not change and hence it

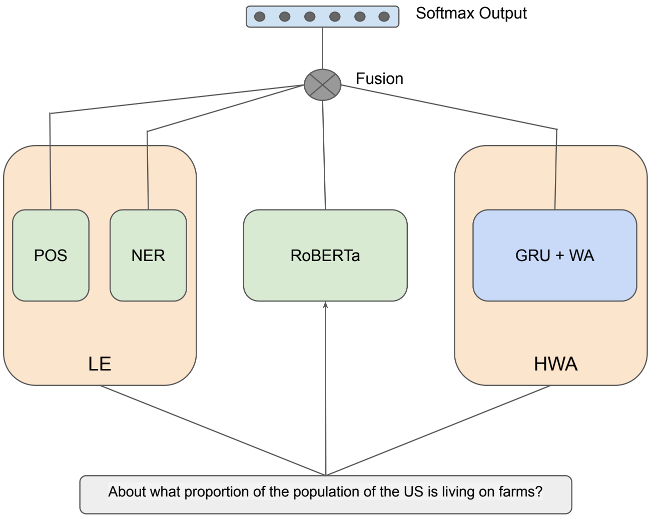
Softmax Output
Fusion
POS
NER
RoBERTa
GRU + WA
LE
HWA
About what proportion of the population of the US is living on farms?

Figure 1: A high level architectrue digram of the proposed model BloomNet . POS (Part of Speech Module. NER(Named Entity Recognition) Module. HWA (Hierarchical Word Attention) Module. LE (Linguistic Encapsulation).

carries linguistic information. We use `CLS` token representation of the model and we define it as as follows:

$$h_{\text{POS}} = f_{\text{POS}}(x) \tag{2}$$

$$h_{\text{NER}} = f_{\text{NER}}(x) \tag{3}$$

3.3 Hierarchical Word Attention and Classification

Inspired by (Yang et al., 2016b) we use word level attention to get the better dense representation of the input. For this we use GRU Cho et al. (2014) and apply word level attention on its output. As result we get a vector from this module as input representation and we use this along with other information for classification. We denote this as follows :

$$h_{\text{HWA}} = f_{\text{HWA}}(x) \tag{4}$$

Finally, we get four different representation coming from different components and we fuse these information using concatenation and feed this to a linear classification model. We write the concatenation as:

$$H = h_{\text{Rep}} \oplus h_{\text{POS}} \oplus h_{\text{NER}} \oplus h_{\text{HWA}} \tag{5}$$

Classification module can be represented as:

$$C = f_{\text{c}}(H) \tag{6}$$

4 Experiments and Results

4.1 Dataset

We use two open domain datasets to evaluate the proposed approach. First dataset was proposed in (Yahya et al., 2012) which comprises 600 open-ended questions. The second dataset was compiled from a variety of websites, publications, and previous research (Haris and Omar, 2015). It contains 141 open-ended questions. The datasets are annotated and classified into six categories (Knowledge, Comprehension, Application, Analysis, Synthesis, and Evaluation). Table 1 illustrates the label distribution for both datasets. The questions in these two datasets come from a variety of fields of study, including chemical, literature, biological, artistic, and computer science, among others.

4.2 Baselines

In this section, we describe the various baseline models that we used for comparative analysis. These models are arranged in the order of their performance.

4.2.1 VDCNN

Very Deep CNN (VDCNN) (Schwenk et al., 2017) learns a hierarchical representation of a sentence with the help of a deep stack of convolutions and max-pooling of size 3 and by operating at the char-

Cognitive Level	Dataset 1	Dataset 2
Knowledge Level	100	26
Comprehension Level	100	23
Application Level	100	15
Analysis Level	100	23
Synthesis Level	100	30
Evaluation Level	100	24
Total	**600**	**141**

Table 1: Number of questions in each cognitive level

acter level representation of the text. VDCNNs are substantially deeper than convolutional neural networks published previously. This is the first CNN model to present the "advantage of depth" in the field of NLP.

4.2.2 LSTM

Text is viewed as a sequence of words in RNN-based models, which are designed to capture word dependencies and text structures for text classification. RNNs (Jain and Medsker, 1999) can memorise the local structure of a word sequence, but they struggle with long-range dependencies. Long-Short Term Memory (LSTM) (Sari et al., 2020) is the most popular variant of RNN, that is created to capture long term dependencies. Vanilla RNNs suffer from gradient vanishing problem and LSTMs resolve this issue by using a memory cell that remember values across arbitrary time periods.

4.2.3 HAN

Hierarchical Attention Networks (HAN) (Yang et al., 2016a) collects relevant tokens from sentences and aggregate their representation with the help of an attention mechanism. The same approach is used to retrieve relevant sentence vectors that is used in the classification task.

4.2.4 CNN

RNNs are taught to detect patterns over time, while CNNs (Kim, 2014) are taught to recognise patterns over space. RNNs work for NLP tasks like POS tagging or QA that need understanding of long-range semantics, but CNNs are good for recognising local and position-invariant patterns (LeCun et al., 1998). These patterns could be key phrases expressing a specific emotion or a topic. As a result, CNNs have become one of the most common text classification model.

4.2.5 RCNN

In contrast to CNN, Recurrent CNN (Girshick et al., 2014) comprises of bi-directional recurrent structure that captures greater contextual data from word representations. This is followed by a max pooling layer which is responsible for extracting key features for text classification.

4.2.6 Seq2Seq-Attention

Deep learning models known as sequence-to-sequence (Bahdanau et al., 2015) models have been deployed in tasks such as machine translation, text summarization, and image captioning. Seq2Seq comprises of encoder, decoder and attention layer where encoder is responsible for compiling data in the form of vector. Further this context is parsed to the decoder that produces desired output sequence. The primary idea behind the attention mechanism is to avoid learning a single vector representation for each sentence and instead be attentive to specific input vectors based on the attention weights.

4.2.7 Self-Attention

Self-attention is a type of attention that allows us to learn the relationship between words in a sentence. Various NLP tasks and Transformers (Vaswani et al., 2017) use self-attention. Despite the fact that CNNs are less sequential than RNNs, the computing cost of capturing relationships between words in a phrase increases with the length of the sentence, much like RNNs. Transformers get around this constraint by using self-attention to compute a "attention score" for each word in a sentence or document in parallel, modelling the influence each word has on the others.

4.2.8 TF-IDF Random Forest

Random Forest (RF) models (Xue and Li, 2015) are made up of a collection of decision trees that were trained on random feature subsets. This model's predictions are obtained via a majority vote of all forest tree projections. In addition, RF classifiers are simple to apply to text classification of high-dimensional noisy data. Furthermore, TF-IDF (Term Frequency Inverse Document Frequency) (Sammut and Webb, 2010) is a commonly used approach for converting text to a number representation that may be employed by a machine algorithm.

TfidfVectorizer weights word counts based on how frequently they appear in the sentence.

4.2.9 DistilRoBERTa

DistilRoBERTa has been distilled from RoBERTa-base model (Liu et al., 2019; Sanh et al., 2019) that contains around half number of parameters as BERT model. It is based on same training process as that of DistilBERT. Moreover, DistilRoBERTa maintains 95 percent of BERT's performance on the GLUE language understanding benchmark (Wang et al., 2018).

4.2.10 RoBERTa

RoBERTa (Robustly Optimized BERT Pretraining Accuracy) model (Liu et al., 2019) is a more robust version of BERT that is trained with a lot more data. It is based on fine-tuning the hyper-parameters that has improved the results and performance of the model significantly. To boost performance of BERT, RoBERTa also modified its training procedure and architecture. These modifications include removing next sentence prediction and dynamically changing the masking pattern during pre-training.

4.3 Experimental Setup

We use HuggingFace Transformers (Wolf et al., 2020), PyTorch (Paszke et al., 2019), PyTorch-Lightning (Falcon, 2019) and Scikit-learn (Pedregosa et al., 2011) for model implementation, training and evaluation. We train all of the models with KFold (k=5) cross validation and report `mean` and `std` values across the folds. We observe that the text in the dataset is relatively short, thus we use `maximum sequence lengths = 128`. For LSTM-like models, we employ `hidden size = 768`, `number of layers = 4`, and `dropout = 0.10` throughout the experiments. We use `Adam` (Kingma and Ba, 2015) as the optimizer and `cross-entropy` as the objective function. We use different `learning rates` for different models depending on how they are initialized, and for `BloomNet` we use `learning rate = 2e-5` and train all models for `50 epochs` with `batch size = 32` and `early stopping` to prevent overfitting. We do not change the shared hyper-parameters across the models so that the comparison is as fair as possible. We do not conduct comprehensive hyper-parameter searches due to computational constraints.

4.4 Results

We evaluate following baselines to compare the performance of our proposed model, BloomNet: VDCNN (Schwenk et al., 2017), LSTM (Sari et al., 2020), HAN(Yang et al., 2016a), CNN(Kim, 2014), RCNN (Girshick et al., 2014), Seq2Seq-Attention (Bahdanau et al., 2015), Self-Attention (Vaswani et al., 2017), Random Forest (Xue and Li, 2015), DistilRoBERTa (Liu et al., 2019), and RoBERTa (Liu et al., 2019). We find that BloomNet outperforms all the considered baselines on the two datasets, demonstrating its higher performance for text classification. Table 2 reports the performance of BloomNet and all the baselines.

The model is trained on Dataset1 and evaluated for both the datasets. Dataset1 is used for evaluating BloomNet's IID performance while Dataset2 is used to test BloomNet's generalisation capabilities (OOD performance) by assessing it on new distributions that our model does not encounter during the training process. We observe that in comparison to the baseline models, BloomNet is less vulnerable to distribution shift.

4.4.1 Comparative Analysis

As seen in Table 2 BloomNet outperforms the baseline models and achieves 87.50 ± 1.88 and 70.40 ± 2.52 and Macro-F1 score 87.23 ± 2.47 and 67.10 ± 2.43 on Dataset1(IID) and Dataset2(OOD) respectively.

In addition, we also made some very in-depth observations while evaluating the baselines. Surprisingly, the TF-IDF (Sammut and Webb, 2010) encoded text with random forest performs better than several strong baselines like LSTM, HAN, CNN, and RCNN. It is the third best performing baseline model that achieves 70.66 ± 2.52 and 62.12 ± 1.38 accuracy and Macro-F1 70.50 ± 2.75 and 58.04 ± 1.73 on Dataset1(IID) and Dataset2(OOD) respectively.

We also observe that Attention based models like Seq2Seq-Attention and Self-Attention show better classification performance than vanilla mod-

[1] Very Deep Convolutional Networks for Text Classification(VDCNN)

[2] Long Short-Term Memory (LSTM)

[3] Hierarchical Attention Networks (HAN)

[4] Convolutional Neural Network (CNN)

[5] Recurrent Convolutional Neural Network (RCNN)

[6] Sequential to Sequential Model with Attention

[7] Term Frequency - Inverse Document Frequency(TF-IDF)

[8] Distilled from RoBERTa model

[9] Robustly Optimized BERT Pre-training Approach

Model OV	Dataset1 (IID)		Dataset2(OOD)	
	Accuracy	**Macro-F1**	**Accuracy**	**Macro-F1**
VDCNN[1]	32.00 ± 6.78	31.70 ± 6.71	28.79 ± 3.82	26.54 ± 4.12
LSTM[2]	58.50 ± 3.99	59.27 ± 3.55	47.09 ± 4.05	45.47 ± 2.71
HAN[3]	59.64 ± 3.72	58.90 ± 4.16	54.69 ± 3.39	50.61 ± 3.12
CNN[4]	60.67 ± 1.11	60.57 ± 1.36	49.79 ± 2.17	48.17 ± 2.00
RCNN[5]	66.33 ± 3.01	65.90 ± 3.51	54.04 ± 3.57	51.05 ± 3.09
Seq2Seq-Attention[6]	64.00 ± 3.09	63.79 ± 3.50	52.91 ± 2.22	50.92 ± 2.11
Self-Attention	70.17 ± 3.55	69.92 ± 3.80	55.46 ± 2.07	52.75 ± 1.81
Random Forest TF-IDF[7]	70.66 ± 2.52	70.50 ± 2.75	62.12 ± 1.38	58.04 ± 1.73
DistilRoBERTa[8]	80.50 ± 3.23	80.21 ± 3.49	67.80 ± 1.59	63.94 ± 1.48
RoBERTa[9]	82.00 ± 2.01	81.67 ± 2.20	68.65 ± 2.74	65.65 ± 2.82
BloomNet	**87.50 ± 1.88**	**87.23 ± 2.47**	**70.40 ± 2.52**	**67.10 ± 2.43**

Table 2: Mean and Standard deviation of the results obtained over 5 folds. BloomNet performs significantly better ($p < 0.004$) than the RoBERTa. **Bold** shows best performance. All models are trained and evaluated on Dataset1 hence IID, and OOD evaluation is performed on Dataset2.

els (like VDCNN, CNN, LSTM, and RCNN). Further, we investigate BERT-based models DistilRoBERTa and RoBERTa (which are pre-trained Transformers) that achieve superior performanc over all the other considered baselines. RoBERTa is the best performing model with accuracy of 82.00 ± 2.01 and 68.65 ± 2.74 and Macro-F1 score 81.67 ± 2.20 and 65.65 ± 2.82 on Dataset1(IID) and Dataset2(OOD) respectively.

4.4.2 Out-of-distribution Generalisation

We evaluate models on new data which is not seen during training to evaluate the OOD robustness. We observe that OOD and IID performance is linearly correlated. The models that do not perform well on IID data such as VDCNN, LSTM, etc also perform poor on OOD data. Pretrained transformers have been proven robust to distribution shift (Hendrycks et al., 2020; Ramesh Kashyap et al., 2021) but in our case we notice significant performance drop (20%) between IID and OOD data across all the pretrained transformer based models in our experiment which is same for other models as well. We hypothesise that this might be caused by large discrepancy between IID and OOD data.

4.5 Ablation Study

Our proposed model BloomNet has three main component: 1. Representation model or Language Encoder 2. Linguistic Encapsulation Module and 3. Hierarchical Word Attention Model. We conduct an ablation study to assess the contribution of different components in our model. First we remove the word attention module from BloomNet and train it like other models with same configuration. We observe the BloomNet without word attention yields ≈ 84 and ≈ 65 accuracy for IID and OOD data respectively. Then we remove the linguistic encapsulation block and train the model like previously. BloomNet without linguistic encapsulation yeilds similar IID performance (≈ 84 accuracy) but performs better on OOD data. If we remove the both components word attention as well as linguistic encapsulation BloomNet is same as RoBERTa (Liu et al., 2019) baseline. The result of ablation is stated in the table 3.

Component	Accuracy	Macro-F1
RoBERTa	82.00	81.67
+ WA	84.11	84.10
+ POS-NER	84.64	84.48
+ WA + POS-NER	**87.50**	**87.23**

Table 3: Ablation results for BloomNet . Mean of IID Accuracy and Macro-F1 is reported. Linguistic Encapsulation along with Word level attention yields significantly better ($p < 0.004$) results. WA: Word Attention. POS: Part-of-speech. NER: Named-Entity Recognition.

5 Discussion

Limitations: We propose a novel transformer based model named BloomNet which has three language encoder (we use RoBERTa), two for linguistic encoding named as POS Encoder and

NER Encoder, and one generic encoder. Due to three large transformer based language encoder proposed model is compute and memory heavy hence it becomes very cumbersome to deploy it in production. To asses the generalization capability of models we evaluate them on a different distribution which they do not see during training. We do not quantify the shift between IID and OOD and we restrict ourselves to only evaluation as investing the cause of performance drop on OOD data is beyond scope of this study. The datasets used in our work is relatively small having 600 and 141 samples respectively in both Dataset1 and Dataset2. Although we do cross validation and report mean and standard-deviation but we expect change in performance on bigger dataset. For same reason we do not train models on Dataset2.

Ethical Considerations: We are well aware of the societal implication of deploying large language models it could have unintended bias against marginalized groups and model itself plays significant role in amplifying those biases. We do not see any immediate misuse of our work, but more research in this area could lead to the development of systems such as automated scoring, which can have a disproportionately detrimental impact on marginalized groups.

6 Conclusion and Future Work

We propose a novel transformer-based model, BloomNet, that captures the linguistic and semantic information to classify the course learning outcomes according to the different cognitive domains of Bloom's Taxonomy. BloomNet outperforms the considered baseline models analyzed in this study in terms of performance and generalization capability. Interestingly, we observe that carefully processed text with TF-IDF encoding outperforms numerous strong baselines like CNN, RNN, and attention based models. We also observe that pretrained Transformers generalize to OOD examples surprisingly well. Overall, we use a state-of-the-art Natural Language Processing (NLP) model for a relatively new task, and we believe it opens up new research directions for NLP in the education domain. We believe that, similar to previous domain-oriented NLP studies, such as NLP4Health, NLP4Programming, LegalNLP, and so on, NLP4Education has the potential to improve existing systems for the mutual benefit of the community and society in general. This is a novel task employing the state-of-the-art Natural Language Processing(NLP) system into education which is relatively new and we believe that it will open a new direction for NLP research.

References

D. Abduljabbar and N. Omar. 2015. Exam questions classification based on bloom's taxonomy cognitive level using classifiers combination. Journal of theoretical and applied information technology, 78:447–455.

F. Adesoji. 2018. Bloom taxonomy of educational objectives and the modification of cognitive levels. Advances in Social Sciences Research Journal, 5.

L. Anderson, D. Krathwohl, and B. Bloom. 2000. A taxonomy for learning, teaching, and assessing: A revision of bloom's taxonomy of educational objectives.

A. S. Attia. 2021. Bloom's taxonomy as a tool to optimize course learning outcomes and assessments in architecture programs.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. CoRR, abs/1409.0473.

Zheng Cai, Lifu Tu, and Kevin Gimpel. 2017. Pay attention to the ending: Strong neural baselines for the roc story cloze task. In ACL.

Wen-Chih Chang and Ming-Shun Chung. 2009. Automatic applying bloom's taxonomy to classify and analysis the cognition level of english question items. 2009 Joint Conferences on Pervasive Computing (JCPC), pages 727–734.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1724–1734, Doha, Qatar. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

et al. Falcon, WA. 2019. Pytorch lightning. GitHub. Note: https://github.com/PyTorchLightning/pytorch-lightning, 3.

Ross B. Girshick, Jeff Donahue, Trevor Darrell, and J. Malik. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. 2014 IEEE Conference on Computer Vision and Pattern Recognition, pages 580–587.

Shreya Gupta, Parantak Singh, Megha Sundriyal, Md. Shad Akhtar, and Tanmoy Chakraborty. 2021. LESA: Linguistic encapsulation and semantic amalgamation based generalised claim detection from online content. In Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, pages 3178–3188, Online. Association for Computational Linguistics.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In NAACL-HLT.

S. S. Haris and N. Omar. 2012. A rule-based approach in bloom's taxonomy question classification through natural language processing. 2012 7th International Conference on Computing and Convergence Technology (ICCCT), pages 410–414.

S. S. Haris and N. Omar. 2015. Bloom's taxonomy question categorization using rules and n-gram approach. Journal of theoretical and applied information technology, 76:401–407.

Dan Hendrycks, Xiaoyuan Liu, Eric Wallace, Adam Dziedzic, Rishabh Krishnan, and Dawn Song. 2020. Pretrained transformers improve out-of-distribution robustness. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 2744–2751, Online. Association for Computational Linguistics.

L. C. Jain and L. R. Medsker. 1999. Recurrent Neural Networks: Design and Applications, 1st edition. CRC Press, Inc., USA.

K. Jayakodi, M. Bandara, and I. Perera. 2015. An automatic classifier for exam questions in engineering: A process for bloom's taxonomy. 2015 IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE), pages 195–202.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In EMNLP.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. CoRR, abs/1412.6980.

Kamran Kowsari, K. Meimandi, Mojtaba Heidarysafa, Sanjana Mendu, Laura E. Barnes, and D. Brown. 2019. Text classification algorithms: A survey. Inf., 10:150.

Adidah Lajis, H. Nasir, and N. A. Aziz. 2018. Proposed assessment framework based on bloom taxonomy cognitive competency: Introduction to programming. Proceedings of the 2018 7th International Conference on Software and Computer Applications.

Y. LeCun, L. Bottou, Yoshua Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. ArXiv, abs/1907.11692.

Shervin Minaee, Nal Kalchbrenner, E. Cambria, Narjes Nikzad, M. Chenaghlu, and Jianfeng Gao. 2020. Deep learning–based text classification. ACM Computing Surveys (CSUR), 54:1 – 40.

Manal Mohammed and N. Omar. 2018. Question classification based on bloom's taxonomy using enhanced tf-idf. International Journal on Advanced Science, Engineering and Information Technology, 8:1679–1685.

Manal Mohammed and N. Omar. 2020. Question classification based on bloom's taxonomy cognitive domain using modified tf-idf and word2vec. PLoS ONE, 15.

K. A. Osadi, Mgnas Fernando, and W. V. Welgama. 2017. Ensemble classifier based approach for classification of examination questions into bloom's taxonomy cognitive levels. International Journal of Computer Applications, 162:1–6.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In Advances in Neural Information Processing Systems, volume 32. Curran Associates, Inc.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in python. Journal of Machine Learning Research, 12(85):2825–2830.

Jeffrey Pennington, R. Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In EMNLP.

Joaquin Quionero-Candela, Masashi Sugiyama, Anton Schwaighofer, and Neil Lawrence. 2009. Dataset shift in machine learning.

Abhinav Ramesh Kashyap, Laiba Mehnaz, Bhavitvya Malik, Abdul Waheed, Devamanyu Hazarika, Min-Yen Kan, and Rajiv Ratn Shah. 2021. Analyzing the domain robustness of pretrained language models, layer by layer. In Proceedings of the Second Workshop on Domain Adaptation for NLP, pages 222–244, Kyiv, Ukraine. Association for Computational Linguistics.

Claude Sammut and Geoffrey I. Webb, editors. 2010. TF–IDF, pages 986–987. Springer US, Boston, MA.

A. Sangodiah, Rohiza Ahmad, and W. Ahmad. 2017. Taxonomy based features in question classification using support vector machine 1.

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.

Winda Kurnia Sari, Dian Palupi Rini, and R. F. Malik. 2020. Text classification using long short-term memory with glove features.

Holger Schwenk, Loïc Barrault, Alexis Conneau, and Y. LeCun. 2017. Very deep convolutional networks for text classification. In EACL.

A. Swart and M. Daneti. 2019. Analyzing learning outcomes for electronic fundamentals using bloom's taxonomy. 2019 IEEE Global Engineering Education Conference (EDUCON), pages 39–44.

Ashish Vaswani, Noam M. Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. ArXiv, abs/1706.03762.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. In BlackboxNLP@EMNLP.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 38–45, Online. Association for Computational Linguistics.

Dashen Xue and Fengxin Li. 2015. Research of text categorization model based on random forests. 2015 IEEE International Conference on Computational Intelligence & Communication Technology, pages 173–176.

Anwar Ali Yahya. 2019. Swarm intelligence-based approach for educational data classification. J. King Saud Univ. Comput. Inf. Sci., 31:35–51.

Anwar Ali Yahya, Z. Toukal, and A. Osman. 2012. Bloom's taxonomy-based classification for item bank questions using support vector machines. In Modern Advances in Intelligent Systems and Tools.

Zichao Yang, Diyi Yang, Chris Dyer, X. He, Alex Smola, and E. Hovy. 2016a. Hierarchical attention networks for document classification. In HLT-NAACL.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016b. Hierarchical attention networks for document classification. In Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

S. L. Zorluoğlu, Kübra Elif Bağrıyanık, and Ayşe Şahintürk. 2019. Analyze of the science and technology course teog questions based on the revised bloom taxonomy and their relation between the learning outcomes of the curriculum. The International Journal of Progressive Education, 15:104–117.