Item Difficulty Modeling Using Fine-Tuned Small and Large Language Models

Ming Li, Hong Jiao, Tianyi Zhou, Nan Zhang, Sydney Peters, Robert W Lissitz
University of Maryland
minglii@umd.edu, hjiao@umd.edu

Abstract

This study investigates methods for item difficulty modeling in large-scale assessments using both small and large language models. We introduce novel data augmentation strategies, including on-the-fly augmentation and distribution balancing, that surpass benchmark performances, demonstrating their effectiveness in mitigating data imbalance and improving model performance. Our results showed that fine-tuned small language models such as BERT and RoBERTa yielded lower root mean squared error than the first-place winning model in the BEA 2024 Shared Task competition, whereas domain-specific models like BioClinicalBERT and PubMedBERT did not provide significant improvements due to distributional gaps. Majority voting among small language models enhanced prediction accuracy, reinforcing the benefits of ensemble learning. Large language models (LLMs), such as GPT-4, exhibited strong generalization capabilities but struggled with item difficulty prediction, likely due to limited training data and the absence of explicit difficulty-related context. Chain-of-thought prompting and rationale generation approaches were explored but did not yield substantial improvements, suggesting that additional training data or more sophisticated reasoning techniques may be necessary. Embedding-based methods, particularly using NV-Embed-v2, showed promise but did not outperform our best augmentation strategies, indicating that capturing nuanced difficulty-related features remains a challenge.

1 Introduction

Standardized tests rely on a detailed analysis of item attributes to ensure psychometric quality of items and test forms. A key attribute is the difficulty level of each item, which is related to the likelihood that an examinee will answer an item correctly. By producing items across a wide difficulty spectrum, it is expected the same measurement precision can be achieved at different ability levels. Moreover, while items that are more challenging typically result in longer response times, the duration of responses can also shed light on examinees' engagement and cognitive strategies, thereby enhancing the validity of the test outcomes. In addition, having a comprehensive understanding of item characteristics is critical for implementing advanced testing methods such as automated item generation, automated item selection in test form assembly, computerized adaptive testing, and individualized assessments (Baylari and Montazer, 2009; Wauters et al., 2012; Kubiszyn and Borich, 2024)

Typically, estimation of item difficulty and the response time required to answer items are derived from item response data gathered during field testing. However, field testing demands a large sample of examinees, which in turn drives up test administration costs (Bejar, 1983; Impara and Plake, 1998). As a result, researchers have explored alternative methods to predict item characteristics without resorting to actual test administration. One strategy involves soliciting difficulty estimates from domain experts and professionals involved in test development, yet this method has not consistently yielded reliable or satisfactory results (Wauters et al., 2012; Attali et al., 2014).

Another research avenue focuses on predicting item attributes based solely on the textual content of the items, including source passages, item stems, and response options (Hsu et al., 2018; Yaneva et al., 2019). This approach leverages text-mining techniques to extract both superficial features (e.g., word counts) and more complex features (e.g., semantic similarities between sentences), which are then used in sophisticated statistical models for prediction. In our study, we employed cutting-edge language models (LMs) for the development of predictive models aimed at estimating these item characteristics. This paper provides a comprehen-

Proceedings of the Artificial Intelligence in Measurement and Education Conference (AIME-Con) – Volume 3: Coordinated Session Papers, pages 48–55

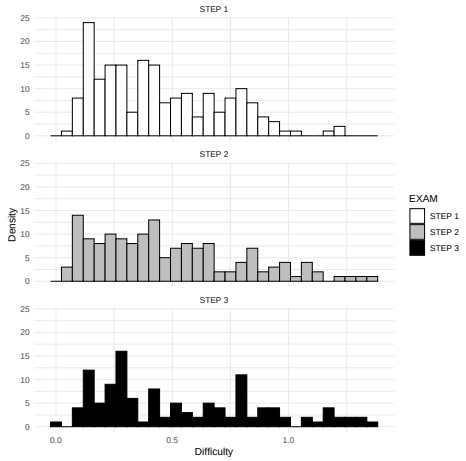

Figure 1: The Item Difficulty Distributions of USMLE Steps 1, 2, and 3 Training Datasets.

sive account of the methodologies implemented and the results obtained from our best-performing models for predicting item difficulty demonstrated using an empirical dataset.

2 Methods

2.1 Datasets

Building on this line of research, this study used data from the National Board of Medical Examiners (NBME) initiated BEA 2024 Shared Task [1] to automate the prediction of item difficulty and response time. The released dataset included 667 items that were previously used and have since been retired from the United States Medical Licensing Examination® (USMLE®)—a series of high-stakes exams [2] that inform medical licensure decisions in the United States. These items, drawn from USMLE Steps 1, 2 Clinical Knowledge (CK), and 3, span a diverse range of topics relevant to medical practice. During the BEA 2024 Shared Task, participating research teams were challenged to leverage NLP techniques using 466 items to develop models to predict item difficulty.

Subsequently, the models developed from the initial phase were applied to a second dataset containing 201 items. This testing set shared the same structural characteristics as the first, except that the values for item difficulty and response time were initially concealed. These values were disclosed only after the BEA 2024 submission deadline, thereby facilitating a fair evaluation of the model's performance in predicting outcomes.

[1] https://sig-edu.org/sharedtask/2024

[2] https://www.usmle.org/step-exams

Figure 1 presents the item difficulty distributions for Steps 1, 2, and 3 USMLE in the training data. The larger values indicate more difficult items. The item difficulty for each Step exam is not evenly distributed. The data imbalance issue is severely critical for this task as the majority of the data lies in the low-difficult range, and only a small number of items is difficult items. This data sparsity in some item difficulty ranges may cause non-representation issues when the item difficulty modeling is developed.

2.2 Models and Methods for Item Difficulty Prediction

This study explored a variety of different methods ranging (i) from small language models (SLM) to large language models (LLMs); (ii) from embedding-based methods to auto-regressive methods; and (iii) from finetune-based methods to inference-only methods. In addition, LLMs with different fine-tuning and prompting techniques were explored. These explorations thoroughly covered the most widely accepted methods off the shelf, which can serve as detailed guidance for future endeavors to other datasets.

2.2.1 SLMs: BERT and its Variants

This study started experimentation with directly fine-tuning small language models for difficulty prediction. We treat this task as a regression task that directly predicts the difficulty value for each item. The models incorporated are mostly encoder-only language models but also some models with encoder-decoder structures, including BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019), DistillBERT (Sanh et al., 2020),deBERTa (He et al., 2021),ELECTRA (Clark et al., 2020), ConvBERT (Jiang et al., 2020), T5 (Raffel et al., 2023), BioClinicalBERT (Alsentzer et al., 2019), and PubMedBERT (Gu et al., 2021). These models intend to set the baseline for comparison. Further, data augmentation was implemented to enhance the prediction accuracy.

2.2.2 Ensemble of SLMs with Majority Voting

Usually, ensemble models are expected to perform better than single-base models. Thus, we explored a commonly used majority voting method to generate more robust results from single SLMs. For the regression task, we used majority voting, which is the average predicted value from different models that participate in the voting process. Compared

with single-model predictions, majority voting is expected to be more robust since the training process is always affected by randomness, and the voting may alleviate the effects of randomness.

2.2.3 SLMs with Data Augmentation

The NLPAug (Ma, 2019) package is utilized for implementing data augmentation. Two types of data augmentation strategies were explored in this study. The two types of data augmentation strategies include: *(i) Augmentation on the fly*, and *(ii) Augmentation with distribution balancing*.

Augmentation-on-the-Fly

In this strategy, we randomly augment original training samples every time it is sent to the model for training. Under this circumstance, all the samples seen by the model are a random augmented version of the original sample, which means the model will not see any identical samples during the epochs of training. This strategy is mostly widely used in the machine learning community as it prevents the model from overfitting to the samples.

Augmentation with Distribution Balancing

This strategy is more complicated as it is specially designed for this task. As shown in Figure 1, the data imbalance issue is severely critical for this task. The majority of the data lies in the low-difficult range, and only a small set of data has high difficulties, e.g., above 0.8. Thus, this imbalance issue causes most of the methods to fail for the prediction, even for our Augmentation-on-the-Fly strategy, as it does not change the frequencies of each sample trained by the model.

Thus, to deal with this issue, we were motivated to balance the sample sizes across the whole distribution, i.e., generate more data for the difficulty levels with lower density and fix them during training. As a regression task, it is naturally difficult to make the data more balanced as they are not as discrete as classification tasks. So, in order to solve it, we first separate all the data samples into 20 bins by fixed intervals and then merge the adjacent bins such that there are at least 2 samples in each bin. This preliminary process converts the consecutive values into discrete bins. Then, we randomly sample 1 instance from each bin to form the validation set. The remaining instances form the training set. This separation ensures that the validation set is balanced enough for fair evaluation. In the remaining bins of the training set, we then randomly augment the existing training samples into a predefined count, i.e., 8 in our experiment. Under this circumstance, the sample counts across all the bins, i.e., the whole distribution, become largely balanced. Then, during training, we fix these samples and do not do augmentation during training. This strategy largely alleviates the distribution imbalance issue and prevents the model from overfitting to high-frequent samples.

Ensemble of Two Data Augmentation Strategies

Both strategies have their own merits. Thus, we further implement an ensemble strategy. For each given instance, each of the above two models generates its own prediction, and then these two predicted values are averaged to simulate the ensemble of the two strategies.

2.2.4 SLMs with LLM Rationales

The training dataset contains only the questions, options, and answers; however, the goal of this study is to predict the difficulties of these items. Thus, there exists a critical gap between the input (item text) and the target (item difficulty). If the input does not contain any information regarding the difficulty, obviously, it will be difficult for SLMs to predict the item's difficulties.

Given the strong reasoning capabilities of LLMs and the potential insights that the chain-of-thought (CoT) prompting technique may provide in reasoning, we hypothesize that incorporating additional rationales that specifically analyze the difficulty of the given items will benefit SLMs in capturing the representative key features in item difficulty modeling. Thus, motivated by the success of CoT (Wei et al., 2023), we employed GPT-4 (OpenAI et al., 2024) to generate a detailed analysis of the item difficulties of different instances, which we refer to as rationales. Then, we concatenate these rationales with the original item text for training the SLMs. The SLMs experimented with are BERT, T5, and Longformer (Beltagy et al., 2020).

2.2.5 BERT with Step-Wise Data Augmentation

Another critical issue of the existing training dataset is its imbalanced nature across exams in the three steps. As shown in Figure 1, the number of step 1 exam items is larger than that for the step 2 and step 3 exam items. To solve this issue, a step-wise data augmentation strategy was implemented using the Python package: NLPAug(Ma, 2019) to augment data in steps 2 and 3 exams for training the BERT model as it was the best-performing base model. Thus, the proposed step-wise data augmen-

tation strategy yielded more augmented data points for step 2 and step 3 exams, while fewer augmented data points for step 1 exam. Further, this step-wise data augmentation method was applied to augment data for the BERT model augmented with the LLM rationales already.

2.2.6 LLMs Finetuning and In-context Learning

All the above methods are based on small language models. In addition, we explored how LLMs performed on this task. Finetuning on LLMs is typically the most commonly used technique when we need LLMs to handle a new task. However, most of the modern LLMs follow the decoder-only structure, which predicts each token in an auto-regressive manner. The modern decoder-only LLMs are more capable of text generation rather than regression tasks, especially when the task is not previously learned during the pretraining phase.

The first category of method we explored was the finetune-based method. Since we only have hundreds of training samples, which is typically not enough for LLMs, we select Phi3 (Abdin et al., 2024) as our base LLM and utilize the full finetuning and LoRA finetuning strategies (Hu et al., 2021). In addition, In-context Learning (ICL) (Brown et al., 2020) is another widely used method for LLM prediction. ICL is less affected by the sparsity of the training data. Thus, we also explored using ICL. One of the biggest advantages of ICL is that it does not require training, thus, it can be used in any LLM, even for closed-source LLMs like GPT-4.

2.2.7 LLMs Embeddings

Previous explorations on LLMs employed the auto-regressive manner of decoder-only structures, which might not be able to well capture the distribution from training data. Typically, embeddings are more effective for regression tasks. Thus, we further explored using LLM embeddings for the prediction, specifically, we utilized the most current state-of-the-art embedding model NV-Embed-v2 (Lee et al., 2024) as the encoder for item difficulty modeling. Further, we trained several additional layers for the difficulty prediction. We also explored combining the benefits of the auto-regressive manner and the benefits of the embedding method together by first generating rationales that specifically analyze the difficulty of the given items, then the SOTA embedding LLMs are utilized to capture the overall distribution of the training data.

2.3 Evaluation of Model Performances

This study used the root mean squared error (RMSE) to evaluate the model performance. This is the evaluation criterion used in the competition. To use the results from the competition as a reference to evaluate the performance of the models and the methods we proposed in this study, we computed RMSE for each model and method explored.

3 Results

3.1 SLMs: BERT, Its Variants, and the Ensemble Models

The performance of the fine-tuned SLMs and the ensemble models is summarized in Table 1. As noted, BERT and Roberta have the top performances, with BERT yielding the smallest RMSE. Contrary to our expectation, utilizing BERT trained with medical-related data (BioClinicalBERT and PubMedBert) does not show an evident improvement in model performance in predicting item difficulty, which might be caused by the class imbalance in the potential distribution gaps. These two models might have a better general understanding of medical-related knowledge, but this knowledge still has a gap in understanding and reasoning item difficulty, which is a basic concept in the psychometric analysis of test items.

The ensembled BERT with a Majority Voting strategy (with RMSE of 0.2981) also exceeds the first place on the leaderboard, showing the effectiveness of this strategy. However, BioClinicalBERT and PubMedBert do not benefit from the ensemble strategy. It is reasonable that the performances of these models are originally not good, and voting by multiple not high-performing models may not necessarily further increase the performance.

3.2 SLMs with Data Augmentation

Our best-performing model is BERT with an ensemble of two types of data augmentation strategies. Both data augmentation strategies (Augmentation-on-the-fly and Augmentation with distribution balancing result in excellent performances that exceeded the first place (RMSE: 0.299) on the leaderboard. Augmentation-on-the-fly yielded RMSE of 0.2975 while augmentation with distribution balancing yields 0.2985 of RMSE, which also exceeds the first place on the leaderboard. Further, the en-

Table 1: Performances for Fine-Tuned SLMs: BERT and its Variants and SLMs with Majority Voting

Model	RMSE
BERT	0.2990
RoBERTa	0.2997
DistilBERT	0.3022
DeBERTa	0.3060
ELECTRA	0.3026
ConvBERT	0.3015
T5	0.3023
BioClinicalBERT	0.3043
PubMedBert	0.3067
BERT (Majority Voting)	0.2981
BioClinicalBERT (Majority Voting)	0.3052
PubMedBert (Majority Voting)	0.3086

Table 2: Performance of the BERT Models with Different Data Augmentation Strategies and the Top Performing Models in the Leaderboard.

Rank	Studied Methods/Team Name	RMSE
Ours	Ensemble of Two Strategies	0.2926
Ours	Augmentation-on-the-Fly	0.2975
Ours	Augmentation with Balancing	0.2985
1	electra	0.299
2	UPN-ICC (run1)	0.303
3	Roberta	0.304
4	RandomForest	0.305
5	ENSEMBLE	0.305
6	Predictions	0.305
7	FEAT	0.305
8	ROBERTA	0.306

semble of these two strategies further leads to an extraordinary performance of 0.2926 in terms of RMSE, which also exceeds the first place on the leaderboard by a really large margin. The performances are further compared in Table 2.

3.3 SLMs with LLM rationales

We used GPT-4 to generate detailed rationales for the difficulties of the items. We concatenated the generated rationales with the original item text for training BERT, T5, and Longformer. The model performances are summarized in Table 3. The models did not perform as effectively as expected. This might be related to the sparsity of the training data in each step exam. When the sample size of the training data is relatively small for each step exam, even the generated rationales may not be able to capture the key item characteristics that distinguish them in terms of item difficulty, though the sample size for each step exam has been increased.

Table 3: Performances for SLMs with LLM Rationales

Model	RMSE
BERT + GPT4 rationales	0.3029
T5 + GPT4 rationales	0.3047
Longformer + GPT4 rationales	0.3050

Table 4: Performances for SLMs with Step-wise Data Augmentation

Model	RMSE
BERT + Step	0.3009
BERT + GPT4 rationales + Step	0.3000

3.4 BERT with Step-Wise Data Augmentation

With the step-wise data augmentation strategy, more synthetic data points were generated to increase more item samples for step 2 and step 3 exams, while slightly more items for the step 1 exam. Step-wise data augmentation was applied to both the BERT model and the BERT model with rationals as data augmentation. The performances of these two models are presented in Table 4. The step-wise data augmentation did not improve the performances of the BERT model, and the BERT model with rationales as augmented data was not better than the first-place model on the leaderboard, both with a slightly larger RMSE of 0.3. This finding indicated that the class imbalance issue in item difficulty distribution is more severe than the class imbalance across the exams in different steps.

3.5 LLMs Finetuning and In-context Learning

We fine-tuned Phi3 as our base LLM and utilized the full finetuning and LoRA finetuning methods. Although the LoRA finetuning method is typically useful for low-resource situations, the tremendous distribution gap between the LLM itself and the learning target causes the LLM to hardly learn anything. On the other hand, when utilizing full finetuning, the LLM is able to partially learn the distribution of the learning target and thus predict item difficulty in the testing dataset with a reasonable value. However, the sparsity of the training samples

Table 5: Performances for LLMs In-context Learning.

Model	RMSE
Phi3 with full finetuning	0.3816
Phi3 with Lora finetuning	0.7632
GPT4	0.3556
GPT4 (ICL)	0.3553

Table 6: Performances for LLMs Embeddings

Model	RMSE
NV-Embed-v2	0.3065
NV-Embed-v2 + GPT4 rationales	0.3023

largely affects its performance, yielding an RMSE of 0.3816, the worst among the models explored in this study except Phi3 with Lora finetuning.

The performances of GPT-4 for item difficulty prediction, with or without using ICL are presented in Table 5. The performances for GPT-4 and GPT-4 with ICL were both worse than the first place on the leaderboard with RMSE larger than 0.355. These results not only show that ICL is not effective on this task but also indicate the difficulty of this task, even the powerful GPT-4 can not yield promising performance.

3.6 LLM Embeddings

As embeddings are effective for regression tasks, we utilized the embedding model NV-Embed-v2 as the encoder and trained several additional layers for the difficulty prediction. Further, NV-Embed-v2 was enhanced by the rationales generated by GPT-4. The model performances are summarized in Table 6. The performances of these two approaches did not beat the first place in the leaderboard with RMSE larger than 0.302. Again, this indicates the difficulty of this task due to the small sample size of training data and the class imbalance issue when item difficulty is represented on a continuous scale.

4 Discussion and Conclusion

In this study, we explored different language models as well as different data augmentation methods for item difficulty modeling for large-scale standardized assessments, leveraging both SLMs and LLMs. Our results demonstrated that the application of data augmentation techniques, particularly our proposed method combining both on-the-fly data augmentation and distribution balancing data augmentation, achieved a slightly lower RMSE of 0.2926. This performance surpasses the first-place winning model in the BEA 2024 Shared Task competition leaderboard (RMSE = 0.299), indicating that our ensemble approach slightly outperforms all other reported models on the leaderboard for this dataset. This finding highlights the effectiveness of data augmentation in improving model performance and mitigating the challenges posed by data imbalance sparsity in some regions of the item difficulty scale.

Our comparative analysis of different modeling approaches revealed several key insights. Firstly, while fine-tuning SLMs such as BERT and RoBERTa yielded smaller RMSE, the introduction of domain-specific models such as BioClinicalBERT and PubMedBERT did not significantly improve model performance, likely due to distributional gaps between medical literature and test item difficulty prediction. Moreover, majority voting among multiple models provided additional robustness, further confirming the benefits of ensemble learning techniques in regression tasks.

The integration of LLMs introduced additional challenges. While models such as GPT-4 exhibited strong generalization capabilities in other NLP tasks, their performance in item difficulty prediction was limited. This outcome suggests that the scarcity of training data and the absence of explicit difficulty-related context in the input might hinder the effectiveness of LLMs in this task. Our attempts to bridge this gap using chain-of-thought prompting and rationale generation did not yield substantial improvements, likely due to insufficient training data to fully capture the key item characteristics along the item difficulty scale and ultimately exploit the advantages of LLM-based reasoning.

Note that even though the difference between the RMSE of our best-performing model, an ensemble of BERT models with two data augmentation strategies, and that of the first-place model in the competition was only 0.0064, the impact of such a difference might be meaningful for high-stakes testing programs. In large-scale standardized assessments for high-stakes decisions like the USMLE, small numerical improvements in predictive metrics such as RMSE may translate into practically meaningful impacts. More accurate item difficulty predictions may lead to improved item selection, test assembly, and better-informed decisions about examinees. Future studies may explore the impact of such slight improvement in item difficulty prediction on improvements at the overall test level.

References

Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, Alon Benhaim, Misha Bilenko, Johan Bjorck, Sébastien Bubeck, Martin Cai, Qin Cai, Vishrav Chaudhary, Dong Chen, Dongdong Chen, and 110 others. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *Preprint*, arXiv:2404.14219.

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. *Preprint*, arXiv:1904.03323.

Yigal Attali, Luis Saldivia, Carol Jackson, Fred Schuppan, and Wilbur Wanamaker. 2014. Estimating item difficulty with comparative judgments. *ETS Research Report Series*, 2014(2):1–8.

Ahmad Baylari and Gh A Montazer. 2009. Design a personalized e-learning system based on item response theory and artificial neural network approach. *Expert Systems with Applications*, 36(4):8013–8021.

Isaac I Bejar. 1983. Subject matter experts' assessment of item statistics. *Applied Psychological Measurement*, 7(3):303–310.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. *Preprint*, arXiv:2005.14165.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2021. Domain-specific language model pretraining for biomedical natural language processing. *ACM Transactions on Computing for Healthcare*, 3(1):1–23.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. *Preprint*, arXiv:2006.03654.

Fu-Yuan Hsu, Hahn-Ming Lee, Tao-Hsing Chang, and Yao-Ting Sung. 2018. Automated estimation of item difficulty for multiple-choice tests: An application of word embedding techniques. *Information Processing & Management*, 54(6):969–984.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *Preprint*, arXiv:2106.09685.

James C Impara and Barbara S Plake. 1998. Teachers' ability to estimate item difficulty: A test of the assumptions in the angoff standard setting method. *Journal of Educational Measurement*, 35(1):69–81.

Zi-Hang Jiang, Weihao Yu, Daquan Zhou, Yunpeng Chen, Jiashi Feng, and Shuicheng Yan. 2020. Convbert: Improving bert with span-based dynamic convolution. *Advances in Neural Information Processing Systems*, 33:12837–12848.

Tom Kubiszyn and Gary D Borich. 2024. *Educational testing and measurement*. John Wiley & Sons.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2024. Nv-embed: Improved techniques for training llms as generalist embedding models. *arXiv preprint arXiv:2405.17428*.

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. *Preprint*, arXiv:1907.11692.

Edward Ma. 2019. Nlp augmentation. https://github.com/makcedward/nlpaug.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer. *Preprint*, arXiv:1910.10683.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *Preprint*, arXiv:1910.01108.

Kelly Wauters, Piet Desmet, and Wim Van Den Noortgate. 2012. Item difficulty estimation: An auspicious collaboration between data and judgment. *Computers & Education*, 58(4):1183–1193.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Victoria Yaneva, Peter Baldwin, Janet Mee, and 1 others. 2019. Predicting the difficulty of multiple choice questions in a high-stakes medical exam. In *Proceedings of the fourteenth workshop on innovative use of NLP for building educational applications*, pages 11–20.