Probing Effects of Contextual Bias on Number Magnitude Estimation

Xuehao Du[1,3], Ping Ji[1*], Wei Qin[2,3], Lei Wang[4], and Yunshi Lan[5]
[1] Hefei University, Hefei, 230601, China
[e-mail: duxhya@gmail.com, jiping@hfuu.edu.cn]
[2] Hefei University of Technology, Hefei, 230009, China
[e-mail: qinwei.hfut@gmail.com]
[3] Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei, 230088, China
[4] Singapore Management University, Singapore, 188065, Singapore
[e-mail: lei.wang.2019@phdcs.smu.edu.sg]
[5] East China Normal University, Shanghai, 200062, China
[e-mail: yslan@dase.ecnu.edu.cn]
[*]Corresponding author: Ping Ji

Received May 8, 2023; revised October 19, 2023; accepted August 18, 2024; published September 30, 2024

Abstract

The semantic understanding of numbers requires association with context. However, powerful neural networks overfit spurious correlations between context and numbers in training corpus can lead to the occurrence of contextual bias, which may affect the network's accurate estimation of number magnitude when making inferences in real-world data. To investigate the resilience of current methodologies against contextual bias, we introduce a novel out-of-distribution (OOD) numerical question-answering (QA) dataset that features specific correlations between context and numbers in the training data, which are not present in the OOD test data. We evaluate the robustness of different numerical encoding and decoding methods when confronted with contextual bias on this dataset. Our findings indicate that encoding methods incorporating more detailed digit information exhibit greater resilience against contextual bias. Inspired by this finding, we propose a digit-aware position embedding strategy, and the experimental results demonstrate that this strategy is highly effective in improving the robustness of neural networks against contextual bias.

Keywords: Natural language processing, Question answering, Out of distribution, Contextual bias, Number magnitude estimation.

1. Introduction

Understanding numbers in text demands contextual consideration, as demonstrated by the multiple meanings of "7" in "Tom woke up at 7 a.m. and bought 7 apples". Language models like BERT [1] and T5 [2] use contextual encoding to comprehend the semantics of numbers, but this approach may lead to inaccurate estimations of number magnitude. Language models, like humans, may misjudge number magnitude due to contextual factors. For instance, when asked which is heavier between 1000g of iron and 1000g of cotton, untrained children often overestimate the weight of iron and underestimate the weight of cotton because the context of iron usually involves larger numbers than the context of cotton. When the context surrounding a number interferes with accurate magnitude estimation, we refer to this phenomenon as contextual bias. Our paper demonstrates that deep learning methods using contextual encoding are susceptible to this bias, resulting in inaccurate estimations of number magnitude.

Bias is extensively prevalent in deep learning [3, 4]. Contextual bias can arise when a powerful network overfits spurious correlations that are commonly present in the training corpus due to corpus biases. As a result, the network may perform poorly on new data that does not exhibit the same biases as the training corpus. Specifically, prior research implies that deep learning models can capture the correlation between context and number magnitude. For instance, [5] discovered that pre-trained language models could produce appropriate noun scalar sizes based on contextual inputs. This is because neural networks have powerful fitting capabilities that enable them to memorize all correlations and minimize training errors [6, 7, 8]. Furthermore, [9, 10] revealed the prevalent biases in the existing training corpus, including reporting bias and polysemy bias. As depicted in **Fig. 1**, the English corpus primarily collected from the Northern Hemisphere Internet creates a correlation between July and high temperatures. This correlation can lead to erroneous predictions when neural networks with strong fitting ability are trained on such a biased corpus and applied to real-world data from the Southern Hemisphere.

To assess the resilience of current methodologies against contextual bias, we introduce an out-of-distribution numerical comprehension dataset that features specific correlations between contextual cues and numerical values in the training data, which are not present in the test data. We select number-related question-answering as the benchmark task due to its complexity, involving multiple types of inquiries. To answer such questions, models must possess a genuine comprehension of numbers, as well as the capacity to learn numerical reasoning through textual descriptions. In the proposed dataset, through our deliberate design, there is a correlation between context and numbers in the training set, while in the test set, this correlation is distorted. When models overfit spurious correlations between context and numbers to minimize the training error, their reliance on these correlations can lead to poor performance when making inferences.

Using our proposed dataset, we assess the robustness of various numerical encoding and decoding methods when confronted with contextual bias. All methods are described in the evaluation experiment section below. We observe that numerical encoding methods, which include more detailed digit information, exhibit greater resilience against contextual bias. For instance, the digit-aware encoding method outperforms other encoding methods in out-of-distribution scenarios, yielding superior outcomes.

Based on our findings, we propose a digit-aware position embedding strategy, which can be integrated into language models. In specific, we add an additional vector encoding for digit positions on top of the original position embedding in the language model. This vector can more explicitly provide the magnitude meaning of each digit in each number's place value.

The experiments demonstrate that employing the digit-aware position embedding strategy can enhance the robustness of models against contextual bias.

Below, we summarize the contributions of this paper.

(1) We introduce a novel out-of-distribution number-related question-answering task that can be utilized to assess the resilience of models against contextual bias.

(2) We assess the resilience of various numerical encoding and decoding methods against contextual bias and make a surprising observation that the encoding methods that incorporate more digit information exhibit greater robustness.

(3) Building on this observation, we propose a digit-aware position embedding strategy that offers explicit magnitude meaning for each digit. Our experiments confirm that the digit-aware position embedding strategy is highly effective in enhancing the model's robustness against contextual bias.

The remainder of this paper is structured as follows. Section 2 provides an overview of the related work. Section 3 introduces the dataset constructed for this study in detail. Section 4 presents the experiments conducted to evaluate different encoding and decoding methods, along with the experimental outcomes and analysis. Section 5 introduces our proposed numerical position embedding strategy. Finally, Section 6 concludes the paper and provides a glimpse into future directions.

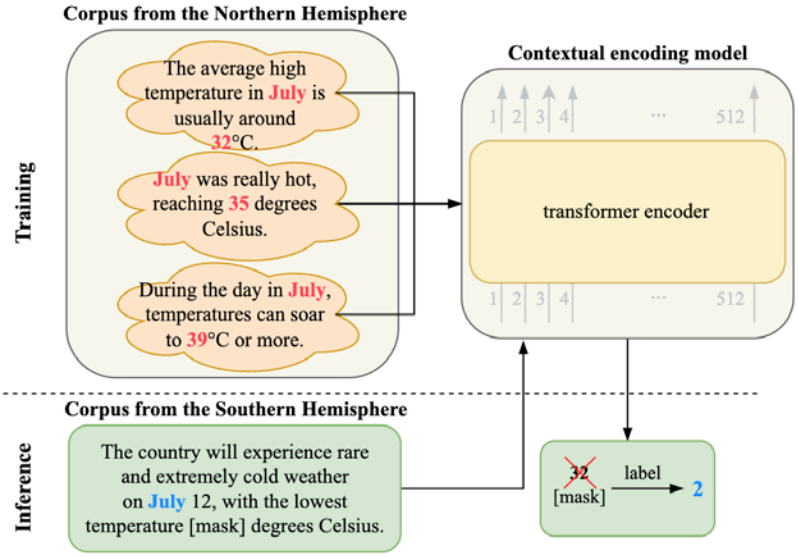


Fig. 1. A visual explanation of contextual bias. The model trained with the northern hemisphere corpus will be influenced by "July" when making inferences on the southern hemisphere corpus and incorrectly predicts the masked position as a large number (like 32).

2. Related Work

2.1 Numerical Encoding

Numbers, like words, are present in almost all documents and often provide much hidden information. However, relatively little attention has been paid to numbers in text, and many systems encode numbers as unknown tokens or ignore them [11, 12]. Several works have explored different methods of string-based numerical encoding, encoding numbers from different granularity levels [1, 13, 14], using scientific notation to emphasize information

about the exponents and mantissa of numbers [5, 15], and explicitly enumerating the semantics of digit positions [16, 17]. Other works have explored methods of real-based numerical encoding [18, 19, 20]. However, we focus on the former in this work. This is because the real-based encoding methods only model numbers through various numerical computations; the string-based encoding method encodes numbers in their surface form, and the encoding object is the numbers themselves, which is more consistent with the original purpose of our exploration.

2.2 Impact of Context

Context is a crucial element in machine learning as it can provide valuable information that contributes to better comprehension and utilization of data. Several studies have indicated that contextual information is beneficial in improving the performance of neural networks on various tasks, such as hate speech detection [21, 22, 23], depression detection [24], and entity disambiguation [25]. However, the introduction of context does not always guarantee performance improvements. For instance, over-reliance on context to solve classification problems can lead to model flaws [26], longer contexts may hinder attention learning and result in poor model performance [27], and context may amplify perceptible toxicity [28]. In contrast to these works, our study focuses on investigating the impact of context on neural networks' ability to estimate the magnitude of numbers.

2.3 Numerical Question-answering

The reading comprehension task has seen significant progress in the last few years, so much that the state-of-the-art models in many related datasets have surpassed human performance [29, 30, 31]. Nevertheless, existing models still struggle with more complex numerical reasoning tasks [32, 33]. [34] proposed a numerical question-answering dataset. The only supervision provided is for question-answering pairs, and the model must learn to reason numerically, while learning to read and comprehend. However, the numerical operations in this dataset involve only straightforward addition and subtraction. Most current models do not actually understand numbers, relying only on symbolic prediction and span extraction to answer questions [35, 36, 37]. In contrast, our dataset has more types of numerical operations and an additional out-of-distribution test set that can be used to probe whether the models actually understand numbers and perform correct numerical reasoning.

3. Dataset

In this section, we provide a comprehensive overview of the dataset and elaborate on the intricacies of its construction process.

3.1 Dataset Overview

We proposed a unique number-related question-answering dataset that demonstrates a specific correlation between context and numbers within the training set, but this correlation is disrupted in the OOD test set, as shown in **Fig. 2**. In the training and IID test sets, there is a correlation between the contextual word plane and the numbers, i.e., the price of the plane tends to be considerable numbers. However, in the OOD test set, the numbers around the plane are significantly smaller, as the plane here represents the “toy plane” rather than the “real plane” in the training set. This intricate design allows us to investigate the robustness of the model against contextual bias particularly when the correlations between context and numbers in real-

world data differ from those in the training corpus. **Table 1** provides the percentages of the different parts of our dataset. Following, we will present our dataset in two aspects to give a comprehensive understanding.

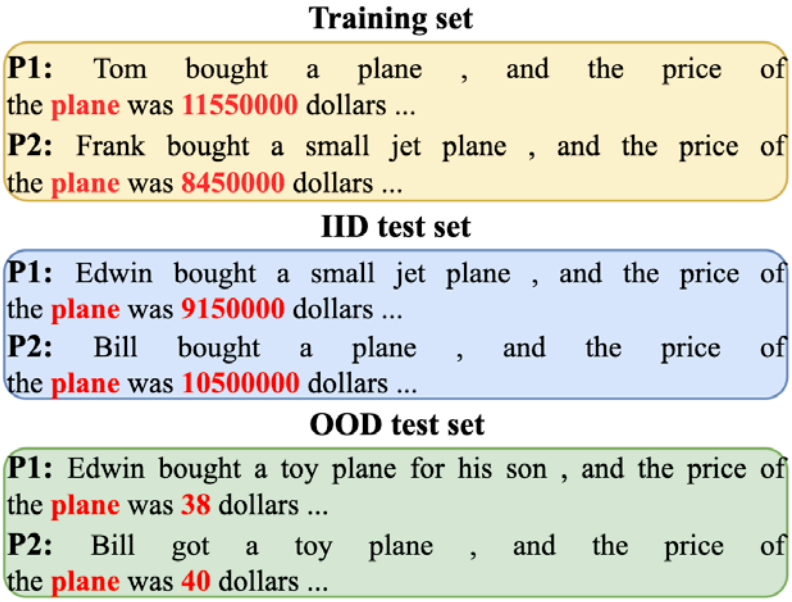


Fig. 2. The uniqueness of the OOD test set. In the training and IID test sets, the word "plane" is always associated with large numbers, but this correlation is disrupted in the OOD test set.

Table 1. The percentages of different parts of the dataset.

Type of data	Percent	Number
Training	76.9	73920
Validation	7.7	7392
IID test	7.7	7392
OOD test	7.7	7392

3.1.1 Topic and Style

The main content of the dataset we constructed in this work is the passage descriptions and questions about the different dimensions of the entity (e.g., price, length, mass, and speed). Each passage in the dataset consists of several sentences that describe the values of these dimensions of entities, along with the corresponding degree of variation. Additionally, each passage is accompanied by multiple questions that require the model to comprehend the numerical values and perform accurate numerical reasoning by connecting the questions to the information provided in the passage. For instance, **Table 2** presents an example of the price dimension in the dataset. The passage outlines the original price of the jacket, as well as any subsequent price changes. The corresponding questions inquire about the current price of the jacket and the extent of the price change.

Table 2. An example of the price dimension. The passage describes the price of the entity's price, and the corresponding two problems examine numerical reasoning capability.

Passage: Cary bought a jacket, and the price of the jacket was 172.53 dollars. Now the price of the jacket has changed to 112% of the previous price.
Question1: How much is the jacket now? Answer1: 193.23 Type: multiplication **Question2**: How much has the price of the jacket changed? Answer2: 20.7 Type: hybrid compute

3.1.2 Dataset Diversity

Our dataset exhibits a high degree of diversity while boasting a large volume. It comprises a total of 154 unique entities distributed across four distinct dimensions. Furthermore, our dataset encompasses various types of problems, including addition, subtraction, multiplication, division, and mixed operations. **Fig. 3** shows the percentage of different question types in the training set. [16] had generated a numerical question-answering dataset called textual data for model pre-training. **Table 3** provides a comparison of our dataset with textual data in terms of diversity. It can be seen that our dataset covers a broader range of fields, features a more significant number of entities, and includes more complex numerical operations. In contrast, textual data is limited to two fields (history and the National Football League) and a few dozen entities. Moreover, it only involves two simple operations: addition and subtraction.

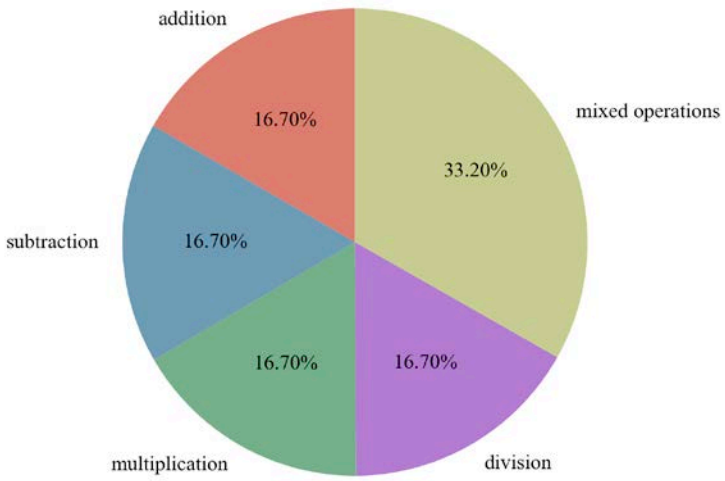


Fig. 3. The percentage of different question types in the training set.

Table 3. The comparison of the diversity between our dataset and textual dataset.

Diversity categories	Our dataset	Textual dataset
Entities	**154**	16
Fields	**4**	2
Numerical operations	**5**	2

3.2 Construction Process

In the following, we outline our dataset construction process and approach. Given the diverse dimensions of entities we focus on, gathering such a dataset from the web can be challenging. To ensure precision and control over the entire dataset generation process, we employ a template-based approach. Specifically, we begin by creating passage and question templates. Then, we gather relevant contextual information. Finally, we use this contextual information to instantiate the templates and generate the final dataset.

3.2.1 Template Creation

We created multiple passages and question templates for each dimension inspired by the framework proposed by [38]. In this framework, text can be mapped as a world state representation consisting of containers, entities, and quantities. We replaced the quantity in the state with the entity's dimension value. **Fig. 4** demonstrates the process of creating the templates with the modified framework. We first constructed some fragments using abstract and regular nouns following the proper syntactic structure, which describe the observations or changes of the entity's dimension values in the container. Then, we spliced these fragments sequentially to form the passage template, and each splicing brought about a transition of the world state. Finally, we derived question templates based on the state transitions of the passage template, which are queries about the entity's current state. **Table 4** presents a subset of the templates for each of the four dimensions, which includes several abstract nouns such as CONT (representing a person), ENT (representing an entity), VERB (representing a verb), and NUM (representing the value of the dimension).

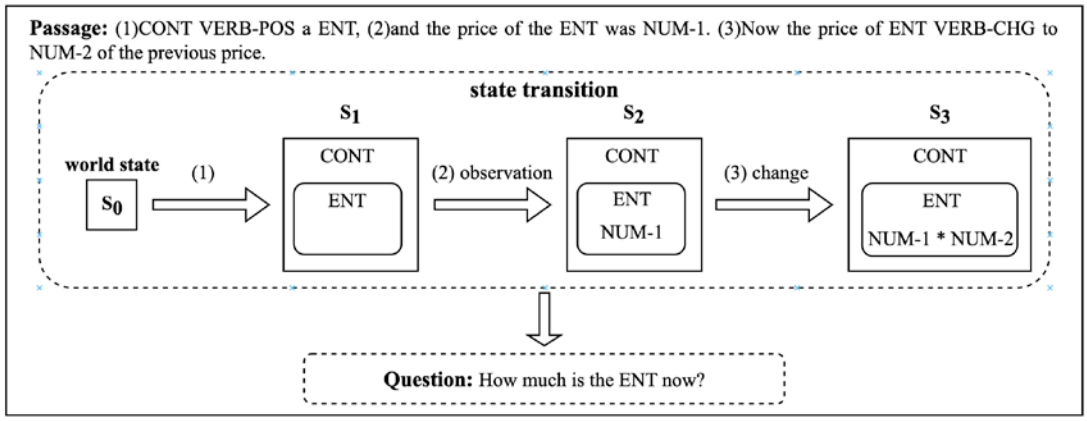


Fig. 4. The whole process of creating passage and question templates using the world state framework.

3.2.2 Context Collection

While our templates provide a structure and style for the sentences in our dataset, contextual information is still required to generate complete sentences. To address this, we collected sets of words that the abstract nouns represent from multiple sources. We obtained the set of ENT from [9], who developed an unsupervised method for collecting quantitative information from web data and used it to create Distribution over Quantities (DoQ), which contains the exact numerical distribution of a large number of nouns in different dimensions. We collected a total of 154 nouns from DoQ, where each noun comes with the corresponding distribution

information, such as range, mean, and median. In addition, we collected 16 common English names and 11 verbs from the network as the sets of CONT and VERB, respectively. By combining our templates with this contextual information, we were able to generate complete sentences for our dataset.

Table 4. A subset of templates for all dimensions. Where colored fonts are abstract nouns that refer to different objects.

Dimensions	Passages	Questions
Price	CONT-1 VERB-POS a ENT-1, and the price of the ENT-1 was NUM-1. Now the price of ENT-1 VERB-CHG to NUM-2 of the previous price.	How much is the ENT-1 now?
Length	CONT-1 VERB-POS a ENT-1 and a ENT-2, the length of the ENT-1 is NUM-1, the length of the ENT-2 is NUM-2. CONT-2 also VERB-POS a ENT-1, the length of CONT-2'S ENT-1 is NUM-3 longer than CONT-1'S ENT-1.	How long is the CONT-2'S ENT-1?
Mass	CONT-1 VERB-POS a ENT-1 weighing NUM-1 and a ENT-2 weighing NUM-2. After a period of time, the weighing of the ENT-1 VERB-ENGCHG by NUM-3.	How much does the ENT-1 weigh now?
Speed	At the beginning, the average speed of the ENT-1 was NUM-1, after a while, the speed of the ENT-1 VERB-POSCHG by NUM-2. The speed of the ENT-2 is NUM-3.	What is the speed of the ENT-1 now?

3.2.3 Template Instantiation

After obtaining the templates and contexts, we used the contextual information to instantiate these templates. First, we randomly selected entities, containers, and verbs from the collected sets of words. We then replaced the corresponding abstract nouns in the templates with these terms. To generate NUMs in the templates, we used the perc_25 and perc_75 from the distribution information. It is crucial to ensure that all generated numbers are within the distribution range of the corresponding entities to obtain a correct and stable distribution of numbers. We generated all numbers with the entity's own distribution for the training and IID test sets. However, for the OOD test set, we shuffled the distributions of all entities of the same dimension. This procedure resulted in different correlations between contexts and numbers in the OOD test set compared to the training set, as illustrated in **Fig. 2**. Finally, we analyzed the instantiated passages and questions to obtain all the answers by determining the category of the questions and performing numerical operations on the relevant numbers in the passages.

4. Evaluation and Results

We assessed the robustness of various numerical encoding and decoding methods against contextual bias on our proposed dataset.

4.1 Basic Architecture

We chose the GenBERT and NumNet [35] models for our evaluation experiments as the basic architectures since they have demonstrated strong text comprehension and numerical reasoning capabilities. Notably, both of these models achieved impressive results on the

numerical question-answering dataset known as DROP [34].

4.1.1 GenBERT

The GenBERT model combines the Transformer encoder-decoder architecture [39] with the pre-trained language model, as depicted in **Fig. 5**. In contrast to the original BERT model, GenBERT distinguishes itself by incorporating both an encoder and a decoder, which are initialized with the bert-base weights. To incorporate BERT's representations of the input text during decoding, the weights of the model's encoder and decoder are interconnected. When generating numerical output, the decoder employs a decoding strategy similar to the Transformer, wherein the output from the previous step serves as the input for the next step, with a total of 20 decoding steps. In addition, the model also includes an answer type head and two span extraction heads. The input for the answer type head is the last layer's hidden state corresponding to the first token in the sequence. Its output determines whether the answer type originates from span extraction or decoder generation. The span extraction heads provide positional information, explicitly indicating the starting and ending positions of the answer within the input text. We recommend the reader refer to [14] for more details.

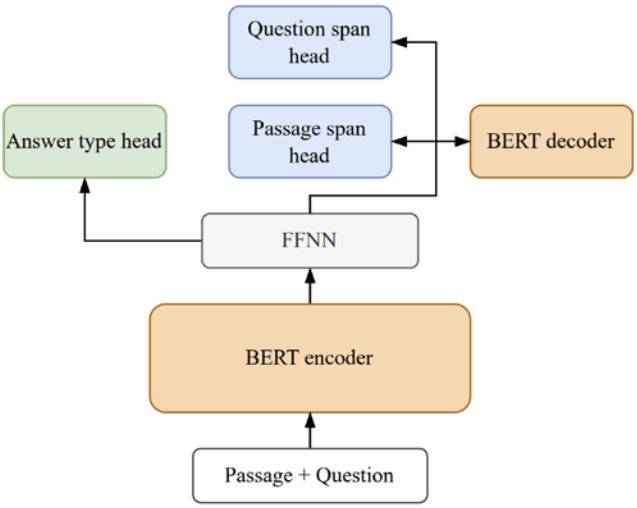


Fig. 5. The architecture of GenBERT.

4.1.2 NumNet

The NumNet model consists of three components: encoding, reasoning, and prediction modules, as depicted in **Fig. 6**. The encoding module uses a pre-trained language model, Roberta [40], which is an enhanced version of BERT with a more significant number of model parameters and more training data for encoding input passages and questions. The reasoning module, a numerically-aware graph neural network (NumGNN), constructs a heterogeneous directed graph to encode numerical relationships between numbers in the input text. The prediction module contains span prediction and symbol prediction. Since the NumNet model uses a symbolic prediction strategy (i.e., assigning a plus sign, minus sign, or zero to each number) when answering numerical operation questions, this strategy is unsuitable for answering certain question types, such as multiplication and division in our dataset. We modified the symbolic prediction part to MLP, which uses regression to predict numbers directly to answer arbitrary types of numerical reasoning questions. We recommend the reader

refer to [35] for more details.

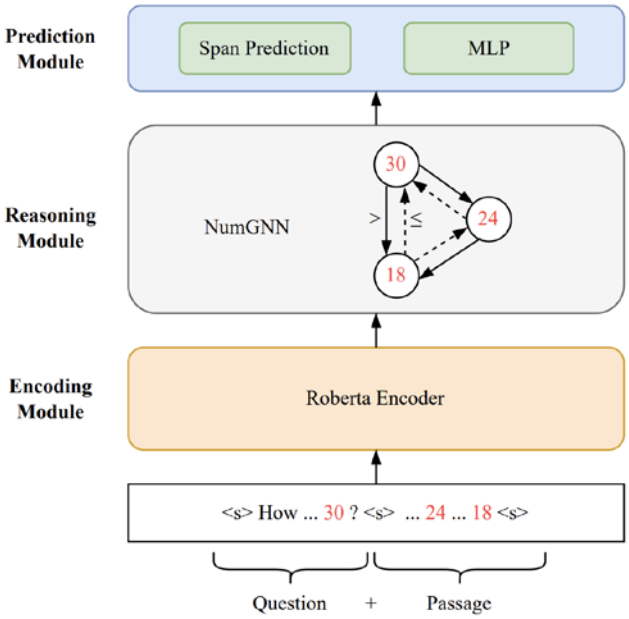


Fig. 6. The architecture of NumNet.

4.2 Encoding and Decoding Methods

We evaluated several common numerical encoding and decoding methods for evaluation. **Table 5** shows these methods and provides some explanations.

(1) Character: Splitting the numbers into character levels and encoding each digit separately.

(2) Sub-word: The number is divided into piece-by-piece forms. For example, “1234” is divided into “12-34”.

(3) Digit-aware: Each digit of the number is followed by a token indicating its position in the number, giving the model more digit information.

(4) Left padding: The length of all numbers is uniform by left padding, and we stipulate the maximum padding length as 10.

(5) Underscore: The numbers are separated by an underscore token, and the model can obtain information about the location of the digit by counting the number of underscores.

(6) Scientific notation: Representing a number in scientific notation form provides the model with direct information on the exponent and mantissa of the number.

(7) Digit-by-digit: When decoding the numbers, using the digit-by-digit generation strategy, we stipulate that the maximum decoding length is 20.

(8) Regression: Using a fully connected layer with an output size of 1 as the decoder to directly predict answers.

(9) Log value: Unlike the regression decoding, this method predicts at log scale and then computes the final answer by an exponential function.

We built on the basic architectures and modified the corresponding encoders and decoders sequentially to evaluate the robustness of the pre-trained language models against contextual bias when employing the aforementioned numerical encoding and decoding methods.

Table 5. Different numerical encoding and decoding methods.

Encoding	Example
Character	8 1 5 . 1
Sub-word	8 15 . 1
Digit-aware	8 [hundred] 1 [ten] 5 [unit] . 1 [tenth]
Left padding	0 0 0 0 8 1 5 . 1
Underscore	8_1_5_._1
Scientific notation	8 . 1 5 1 e 2
Decoding	**Detail**
Digit-by-digit	Maximum decoding step is 20
Regression	Output numbers directly via MLP
Log value	Predicting numbers at log scale

4.3 Experimental Setup

We evaluated the numerical encoding and decoding methods based on the GenBERT and NumNet models in two settings: (1) modifying the encoder to the different numerical encoding methods listed in **Table 5** while keeping the decoder unchanged, and (2) modifying the decoder to the different numerical decoding methods listed in **Table 5** while keeping the encoder unchanged.

We use mean absolute error (MAE) as the evaluation metric. The formula for calculating MAE is as follows:

$$MAE = \frac{1}{m}\sum_{i=1}^{m}\left|(y_i - \hat{y}_i)\right| \tag{1}$$

where m denotes the number of samples, y_i denotes the label value, and $\hat{y}_i$ denotes the predicted value. A smaller value of MAE indicates that the model's prediction is more accurate.

4.4 Main Results

Tables 6 and **7** present the results of various numerical encoding and decoding methods on our dataset. By comparing the performance of these methods on the IID test set and the OOD test set, we can measure their robustness. We observe that both models exhibit notably poorer performance on the OOD test set compared to the IID test set, indicating that contextual bias severely affects the language model's capability to estimate number magnitude accurately. In addition, the performance of the various methods based on the NumNet model is significantly weaker than those based on the GenBERT model. This suggested that the modified NumNet model's numerical understanding and reasoning abilities are relatively limited.

Based on the data in **Table 6**, it is apparent that the digit-aware method exhibits significant improvement over the baseline method on both models. In particular, on the NumNet model, this method achieves the highest performance on both the IID and OOD test sets. This observation implies that encoding methods that incorporate more digit information can better estimate number magnitude and are more likely to be resilient against contextual bias.

The scientific notation method demonstrates superior performance on the IID test set,

implying that providing mantissa and exponent information enables the model to comprehend numbers effectively when the distribution of numbers is consistent. However, when there is a substantial change in the distribution of numbers, the representation of all numbers changes only slightly in the exponent at the end. It may be difficult for the model to capture this variation and consequently misinterpret the magnitude of the numbers.

The sub-word method exhibits extremely poor performance on both test sets, notably falling short of methods such as character and digit-aware. We suspect this result from the sub-word pieces being an inadequate method for encoding numbers, as similar numbers can have completely different sub-word divisions. For instance, “1234” is divided into “12-34”, while “1250” is divided into “1-250”.

The models do not perform exceptionally on the test sets when employing the left-padding method and even exhibit poorer performance on the IID test set. This suggests that normalizing the length of all numbers may hinder the model’s ability to comprehend the quantitative aspects of numbers intuitively.

Table 7 reveals that the regression and log value methods appear to perform well on the OOD test set, but this does not necessarily indicate robustness to contextual bias. This is due to the limitations of the fully connected layer in capturing the complex relationship between context and numbers. Additionally, using a fully connected layer as the decoder to learn mathematical operation ability proves challenging, as reflected in their performance on the IID test set.

We are surprised that in most cases the result of the log value is not better than the regression. Upon analyzing the prediction results of the model, we observe that while the prediction error is minimal at the log scale, even a slight error in the exponential value can result in a significant change in the function’s output due to the nature of the exponential function. This effect is especially pronounced when the label value is enormous.

Table 6. The performance of different numerical encoding methods on our dataset.

Basic architectures	Encoding methods	MAE	
		IID test	OOD test
GenBERT	Character$_{(baseline)}$	161.07	10749.49
	Sub-word	332.25	40153.33
	Digit-aware	108.73	**6482.43**
	Left padding	244.31	7282.53
	Underscore	135.83	10382.3
	Scientific notation	106.02	8649.92
NumNet	Sub-word $_{(baseline)}$	8154.44	67646.09
	Character	5316.24	62869.76
	Digit-aware	**4791.35**	**59482.07**
	Left padding	10777.79	60446.56
	Underscore	9622.5	68284.49
	Scientific notation	4996.32	62011.72

Table 7. The performance of different numerical decoding methods on our dataset.

Basic architectures	Decoding methods	MAE	
		IID test	OOD test
GenBERT	Digit-by-digit$_{(baseline)}$	161.07	10749.49
	Regression	1107.77	2340.68
	Log value	3031.81	3357.63
NumNet	Regression$_{(baseline)}$	8154.44	67646.09
	Log value	16402.98	56915.22

4.5 Contextual Bias Phenomenon

During our evaluation of numerical encoding and decoding methods, we observe that these methods exhibit significant weakness when encountering contextual bias. **Fig. 7** exemplifies a typical instance of erroneous prediction during OOD testing, where the model's predicted values markedly exceed the actual label values. This can be attributed to the fact that the model lacks a true comprehension of the numbers during training, instead remembering that the numbers associated with "helicopter" are in the range [111, 241]. Consequently, during the OOD test, when the number around "helicopter" in the passage falls outside the distribution observed during training, the model's predictions are influenced by learned correlations, leading to the misestimation of a number like 15.17 as being large simply because it appears near the term "helicopter". This phenomenon of contextual bias is widespread during the evaluation process, which explains why all the methods presented in **Tables 6** and **7** performed considerably poorer in the OOD test set than in the IID test set.

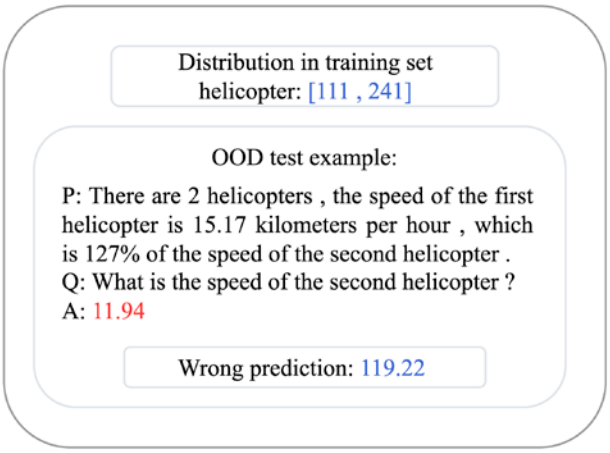


Fig. 7. An example of a typical wrong prediction during the OOD test.

5. Digit-aware position Embedding

Our observation reveals that encoding methods which incorporate more digit information about numbers exhibit improved robustness when confronted with contextual bias during the evaluation. Based on this finding, we propose a digit-aware position embedding strategy that can be seamlessly integrated into existing language models without complex modifications.

In **Fig. 8**, we present the digit-aware position embedding strategy. This strategy involves adding an additional vector to the position embedding in the language model to encode the position information of each digit in the number. Compared to the original position embedding, such an approach can offer an explicit magnitude meaning for each digit. Additionally, for other objects in the input text, such as words, symbols, and special tokens, we utilize a uniform embedding, which can assist the model in distinguishing the numbers from other input words and increase the importance of numbers in the text.

We utilized this strategy to modify the embedding layers of the GenBERT and NumNet models, respectively. The input representation of each number now comprises the sum of token embedding, fragment embedding, position embedding, and digit-aware position embedding, as per our modification. Given that the numbers in our dataset range up to the million level and are accurate to two decimal places, the maximum length of our numerical embedding is 9. With these settings, we evaluated the performance of the modified models on our dataset and compared it with the original model.

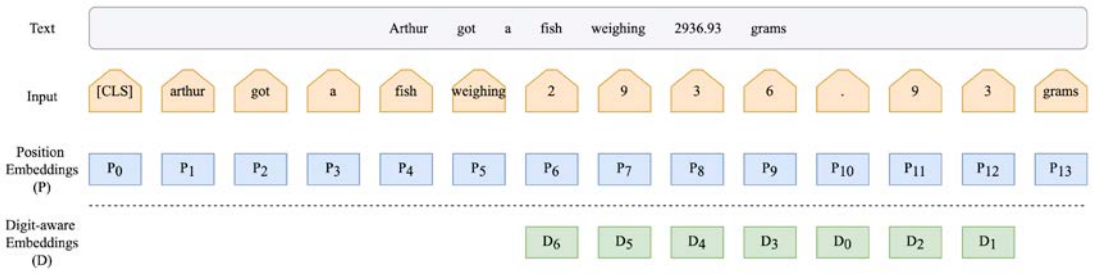


Fig. 8. The digit-aware position embedding strategy. Where the vector D encodes the position of the digit in the number, providing the model information about the magnitude of each digit.

The experimental results are illustrated in **Table 8**. We find that the inclusion of digit-aware position embedding remarkably reduces the mean absolute error of the models on all test sets, which indicates that the strategy effectively enhances the models' ability to comprehend the magnitude of numbers. Furthermore, the modified model outperformed all previously evaluated numerical encoding methods, achieving the best performance on the OOD test set. This result suggests that the digit-aware position embedding strategy can effectively enhance the model's robustness against contextual bias.

Fig. 9 demonstrates the performance comparison between the original GenBERT model and the model enhanced with the digit-aware position embedding strategy on different question types. It is evident that, with the adoption of the novel embedding strategy, the model exhibits significantly improved results across all question types, underscoring the efficacy of this strategy in enhancing the model's numerical reasoning capabilities.

Table 8. The performance of models after employing the digit-aware position embedding strategy.

Methodologies	MAE	
	IID test	OOD test
GenBERT	161.07	10749.49
GenBERT + Digit-aware position embedding	108.69	4945.07
NumNet	8154.44	67646.09
NumNet + Digit-aware position embedding	3147.68	46693.24

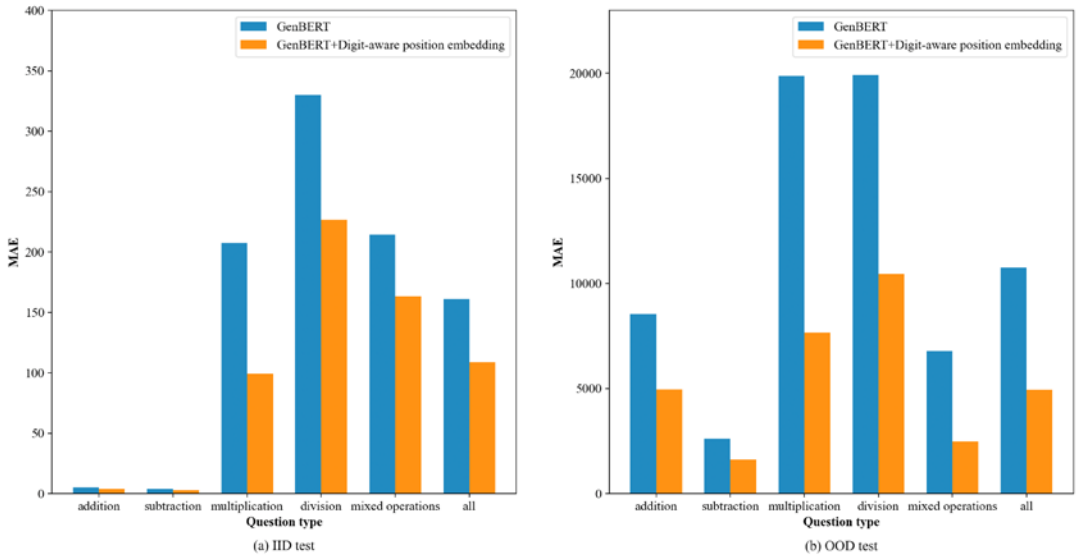


Fig. 9. The performance of the model on different question types.

6. Conclusion

In this paper, we introduce a novel out-of-distribution question-answering dataset and employ the GenBERT and NumNet models as the basic architectures to evaluate various numerical encoding and decoding methods. We demonstrate that deep learning models are susceptible to contextual bias to estimate the magnitude of numbers incorrectly. However, we find that encoding methods that offer detailed digit information are more resistant to contextual bias. We propose a digit-aware position embedding strategy based on this finding and integrate the strategy into the current language model. Experimental results confirm its efficacy in enhancing the model's resilience against contextual bias. Our research reveals the substantial impact of contextual bias on current pre-trained language models, inspiring further efforts from researchers to develop more effective methods to mitigate its influence.

However, there are some limitations to this study. First, due to the difficulty of collecting data in relevant domains, we opted for generating datasets using template-based methods, which may not capture the diversity seen in real-world data. Second, we need further investigation into whether contextual bias also exist in large language models like ChatGPT [41, 42]. In the future, our research will focus on collecting number-related real-world corpora and assessing large language models.

References

[1] J. Devlin, M.-W. Chang, K. Lee et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in *Proc. of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, vol.1, Minneapolis, Minnesota, pp.4171-4186, Jun. 2019.
Article (CrossRef Link)

[2] C. Raffel, N. Shazeer, A. Roberts et al., "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol.21, no.1, pp.5485-5551, Jan. 2020.
Article (CrossRef Link)

[3] W. Ren, L. Wang, K. Liu et al., "Mitigating Popularity Bias in Recommendation with Unbalanced Interactions: A Gradient Perspective," in *Proc. of 2022 IEEE International Conference on Data Mining (ICDM)*, Orlando, FL, USA, pp.438-447, Nov. 2022.
Article (CrossRef Link)

[4] I. Garrido-Muñoz, A. Montejo-Ráez, F. Martínez-Santiago et al., "A Survey on Bias in Deep NLP," *Applied Sciences*, vol.11, no.7, pp.1-26, Apr. 2021. Article (CrossRef Link)

[5] X. Zhang, D. Ramachandran, I. Tenney et al., "Do Language Embeddings capture Scales?," in *Proc. of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pp.292-299, Nov. 2020. Article (CrossRef Link)

[6] K. Hornik, M. Stinchcombe and H. White, "Multilayer feedforward networks are universal approximators," *Neural networks*, vol.2, no.5, pp.359-366, 1989. Article (CrossRef Link)

[7] M. Nielsen, Neural networks and deep learning, CA, USA: Determination Press, 2015.
Article (CrossRef Link)

[8] C. Zhang, S. Bengio, M. Hardt et al., "Understanding deep learning (still) requires rethinking generalization," *Commun. ACM*, vol.64, no.3, pp.107-115, Feb. 2021.
Article (CrossRef Link)

[9] Y. Elazar, A. Mahabal, D. Ramachandran et al., "How Large Are Lions? Inducing Distributions over Quantitative Attributes," in *Proc. of the 57th Annual Meeting of the Association for Computational Linguistics*, Florence, Italy, pp.3973-3983, Jul. 2019.
Article (CrossRef Link)

[10] P. Bhargava and V. Ng, "Commonsense Knowledge Reasoning and Generation with Pre-trained Language Models: A Survey," in *Proc. of the AAAI Conference on Artificial Intelligence*, vol.36, no.11, pp.12317-12325, Jan. 2022. Article (CrossRef Link)

[11] A. Thawani, J. Pujara, F. Ilievski et al., "Representing Numbers in NLP: a Survey and a Vision," in *Proc. of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp.644-656, Jun. 2021.
Article (CrossRef Link)

[12] M. Yoshida and K. Kita, Mining Numbers in Text: A Survey, IntechOpen, 2021.
Article (CrossRef Link)

[13] T. Mikolov, K. Chen, G. Corrado et al., "Efficient Estimation of Word Representations in Vector Space," *arXiv preprint arXiv:1301.3781*, pp.1-12, 2013. Article (CrossRef Link)

[14] M. Geva, A. Gupta and J. Berant, "Injecting Numerical Reasoning Skills into Language Models," in *Proc. of the 58th Annual Meeting of the Association for Computational Linguistics*, pp.946-958, Jul. 2020. Article (CrossRef Link)

[15] T. Berg-Kirkpatrick and D. Spokoyny, "An Empirical Investigation of Contextualized Number Prediction," in *Proc. of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp.4754-4764, Nov. 2020. Article (CrossRef Link)

[16] R. Nogueira, Z. Jiang and J. Lin, "Investigating the limitations of transformers with simple arithmetic tasks," in *Proc. of 1st Mathematical Reasoning in General Artificial Intelligence Workshop, ICLR 2021*, pp.1-14, 2021. Article (CrossRef Link)

[17] J. Kim, G. Hong, K.-m. Kim et al., "Have You Seen That Number? Investigating Extrapolation in Question Answering Models," in *Proc. of the 2021 Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic, pp.7031-7037, Nov. 2021. Article (CrossRef Link)
[18] D. Sundararaman, S. Si, V. Subramanian et al., "Methods for Numeracy-Preserving Word Embeddings," in *Proc. of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp.4742-4753, Nov. 2020. Article (CrossRef Link)
[19] E. Wallace, Y. Wang, S. Li et al., "Do NLP Models Know Numbers? Probing Numeracy in Embeddings," in *Proc. of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, pp.5307-5315, Nov. 2019. Article (CrossRef Link)
[20] C. Jiang, Z. Nian, K. Guo et al., "Learning Numeral Embedding," in *Proc. of Findings of the Association for Computational Linguistics: EMNLP 2020*, pp.2586-2599, Nov. 2020. Article (CrossRef Link)
[21] J. M. Pérez, F. M. Luque, D. Zayat et al., "Assessing the Impact of Contextual Information in Hate Speech Detection," *IEEE Access*, vol.11, pp.30575-30590, Mar. 2023. Article (CrossRef Link)
[22] L. Gao and R. Huang, "Detecting Online Hate Speech Using Context Aware Models," in *Proc. of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, Varna, Bulgaria, pp.260-266, Sep. 2017. Article (CrossRef Link)
[23] E. Mosca, M. Wich and G. Groh, "Understanding and Interpreting the Impact of User Context in Hate Speech Detection," in *Proc. of the Ninth International Workshop on Natural Language Processing for Social Media*, pp.91-102, Jun. 2021. Article (CrossRef Link)
[24] W. Qin, Z. Chen, L. Wang et al., "Read, Diagnose and Chat: Towards Explainable and Interactive LLMs-Augmented Depression Detection in Social Media," *arXiv preprint arXiv:2305.05138*, pp.1-10, 2023. Article (CrossRef Link)
[25] I. O. Mulang', K. Singh, C. Prabhu et al., "Evaluating the Impact of Knowledge Graph Context on Entity Disambiguation Models," in *Proc. of the 29th ACM International Conference on Information & Knowledge Management*, pp.2157-2160, Oct. 2020. Article (CrossRef Link)
[26] S. A. Taghanaki, A. Khani, F. Khani et al., "Masktune: Mitigating Spurious Correlations by Forcing to Explore," in *Proc. of the 36th Conference on Neural Information Processing Systems*, New Orleans, pp.23284-23296, Nov. 2022. Article (CrossRef Link)
[27] S.-Y. Su, P.-C. Yuan and Y.-N. Chen, "How Time Matters: Learning Time-Decay Attention for Contextual Spoken Language Understanding in Dialogues," in *Proc. of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, New Orleans, Louisiana, vol.1, pp.2133-2142, Jun. 2018. Article (CrossRef Link)
[28] J. Pavlopoulos, J. Sorensen, L. Dixon et al., "Toxicity Detection: Does Context Really Matter?," in *Proc. of the 58th Annual Meeting of the Association for Computational Linguistics*, pp.4296-4305, Jul. 2020. Article (CrossRef Link)
[29] I. Yamada, A. Asai, H. Shindo et al., "LUKE: Deep Contextualized Entity Representations with Entity-aware Self-attention," in *Proc. of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp.6442-6454, Nov. 2020. Article (CrossRef Link)
[30] A. Chowdhery, S. Narang, J. Devlin et al., "PaLM: Scaling Language Modeling with Pathways," *Journal of Machine Learning Research*, vol.24, pp.1-113, 2023. Article (CrossRef Link)
[31] Y. Xu, C. Zhu, S. Wang et al., "Human Parity on CommonsenseQA: Augmenting Self-Attention with External Attention," in *Proc. of the 31st International Joint Conference on Artificial Intelligence*, Vienna, Austria, pp.2762-2768, Jul. 2022. Article (CrossRef Link)
[32] Q. Zhang, L. Wang, S. Yu et al., "NOAHQA: Numerical Reasoning with Interpretable Graph Question Answering Dataset," in *Proc. of Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic, pp.4147-4161, Nov. 2021. Article (CrossRef Link)

[33] S. Mishra, M. Finlayson, P. Lu et al., "LILA: A Unified Benchmark for Mathematical Reasoning," in *Proc. of the 2022 Conference on Empirical Methods in Natural Language Processing*, Abu Dhabi, United Arab Emirates, pp.5807-5832, Dec. 2022. Article (CrossRef Link)

[34] D. Dua, Y. Wang, P. Dasigi et al., "DROP: A Reading Comprehension Benchmark Requiring Discrete Reasoning Over Paragraphs," in *Proc. of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Minneapolis, Minnesota, vol.1, pp.2368-2378, Jun. 2019. Article (CrossRef Link)

[35] Q. Ran, Y. Lin, P. Li et al., "NumNet: Machine Reading Comprehension with Numerical Reasoning," in *Proc. of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Hong Kong, China, pp.2474-2484, Nov. 2019. Article (CrossRef Link)

[36] Y. Zhou, J. Bao, C. Duan et al., "OPERA: Operation-Pivoted Discrete Reasoning over Text," in *Proc. of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Seattle, United States, pp.1655-1666, Jul. 2022. Article (CrossRef Link)

[37] K. Chen, W. Xu, X. Cheng et al., "Question Directed Graph Attention Network for Numerical Reasoning over Text," in *Proc. of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp.6759-6768, Nov. 2020. Article (CrossRef Link)

[38] M. J. Hosseini, H. Hajishirzi, O. Etzioni et al., "Learning to Solve Arithmetic Word Problems with Verb Categorization," in *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, pp.523-533, Oct. 2014. Article (CrossRef Link)

[39] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," in *Proc. of 31st Conference on Neural Information Processing Systems (NIPS 2017)*, vol.30, pp.1-11, Dec. 2017. Article (CrossRef Link)

[40] Y. Liu, M. Ott, N. Goyal et al., "RoBERTa: A Robustly Optimized BERT Pretraining Approach," in *Proc. of International Conference on Learning Representations (ICLR 2020)*, pp.1-15, 2020. Article (CrossRef Link)

[41] T. B. Brown, B. Mann, N. Ryder et al., "Language Models are Few-Shot Learners," in *Proc. of 34th Conference on Neural Information Processing Systems (NeurIPS 2020)*, vol.33, pp.1877-1901, Dec. 2020. Article (CrossRef Link)

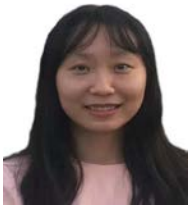
[42] L. Ouyang, J. Wu, X. Jiang et al., "Training language models to follow instructions with human feedback," in *Proc. of 36th Conference on Neural Information Processing Systems (NeurIPS 2022)*, vol.35, pp.27730-27744, 2022. Article (CrossRef Link)

Xuehao Du is currently pursuing the M.S. degree at the School of Advanced Manufacturing Engineering, Hefei University. He received his bachelor's degree in engineering from Anhui Polytechnic University in June 2021. His research interests include natural language processing and deep learning.

Ping Ji is Associate Professor at the School of Advanced Manufacturing Engineering, Hefei University. She received her Master's degree in photonics from Gottingen College of science and art (Germany) in 2010. Her research interests include Media technology and the signal processing and pattern recognition.

Wei Qin received the B.E. degree and Master degree from the School of Computer Science and Information Engineering, Hefei University of Technology, Hefei, China, where he is currently pursuing the Ph.D degree. His research interests include math word problem generation and solving.

Lei Wang received his M.S. degree of Computer Science from University of Electronic Science and Technology of China in 2019. Since 2020, he has been a Ph.D student in the School of Computing and Information Systems at Singapore Management University. His research interests include math word problem solving, recommendation, and interpretability.



Yunshi Lan obtained her Ph.D. degree from the School of Computing and Information Systems at Singapore Management University in 2020. She is currently an associated professor at School of Data Science and Engineering of East China Normal University. Her research interests lie in natural language processing with a focus on question answering, educational NLP.