Expanding Numerical Reasoning Capabilities with ModernBERT and Flan-T5

Aman Kumar
School of Information
University of California, Berkeley
amankr0110@berkeley.edu

Deric Liang
School of Information
University of California, Berkeley
deric.liang@berkeley.edu

Abstract

This paper presents our contribution to Task 7 (NumEval) of SemEval 2024, focusing on the Numerical Reasoning sub-task of the Numeral Aware Headline Generation task. This task focuses on using numeral-heavy news articles to predict the missing number from its corresponding news headline. The NumHG dataset, which consists of news articles along with their masked headlines and reasoning annotations, forms the basis of this task. We fine-tune ModernBERT, an encoder model, and Flan-T5, an instruction-tuned encoder-decoder model, to achieve this task. Our experiments demonstrate that fine-tuning greatly improves the performance of both models, with Flan-T5 achieving the highest accuracy of 78.8%. We highlight the potential of efficient, fine-tuned models in improving numeral-aware Natural Language Processing systems. We also implement metrics not previously used for this task such as Mean Average Percentage Error and Symmetric Mean Average Percentage Error. Despite these efforts, limitations around tokenization, out-of-vocabulary numbers, and model size remain significant.

1 Introduction

In our project, we contribute to Task 7 (NumEval) of SemEval 2024. The NumEval task draws motivation from the fact that much of the Natural Language Processing work has focused on interpreting words in text, whereas capturing the semantics of numbers has been overlooked and proven difficult. In the summary paper of the NumEval task from (Chen et al., 2024a), the authors highlight an example where the number significantly impacts the meaning of a sentence, with real-world impact on court judgments: ‘Stealing $10’ versus ‘Stealing $100,000’. The magnitude of the numbers can also have an impact on financial trading, health diagnoses, weather forecasts, sports scores, and business costs.

We focus on the Numerical Reasoning sub-task of the Numeral-Aware Headline Generation task. The goal of this sub-task is to build systems with an understanding of numeric semantics for filling in a masked headline with a number, given the text of the news article. The data set for this sub-task is NumHG, which contains news articles, masked journalist-written headlines, reasoning annotations/flags, and the missing number (label). This dataset is outlined in (Huang et al., 2024), and was compiled to provide numeral-rich news articles to address a deficiency in generating precise numbers in headlines. We measure the performance on this task using Accuracy, Mean Absolute Percentage Error (MAPE), and Symmetric Mean Absolute Percentage Error (SMAPE).

Our approach leverages both encoder-only and encoder-decoder models. We fine-tune ModernBERT (Warner et al., 2024) as a novel encoder-only approach with an additional Masked Language Modeling classifier layer for token prediction. ModernBERT has not been applied to this task in previous work due to its recent release (Chen et al., 2024a). Additionally, we fine-tune Flan-T5 (Chung et al., 2022) as an encoder-decoder approach to expand on previous work by (Crum and Bethard, 2024), (Gonzalez et al., 2024), and (Chen et al., 2024b). Through our approach, we aim to contribute to the knowledge base by showcasing how recent models perform on this task, how the performance of low-resource approaches compare with the previous literature, and how relatively little fine-tuning can a drastic impact on model performance on a desired task.

In this paper, we provide an overview of the dataset and previous work done on Numerical Reasoning through a review of the NumEval literature as part of SemEval 2024. We provide a summary of the previous participants’ methods, and then discuss our methods. Finally, we compare our results to previous work and discuss potential refinements.

Method	Accuracy
Qwen-72B-Chat + Task Classification + Data Augmentation	0.95
Finetuned GPT-3.5	0.94
Flan-T5 + CoT + Calculator	0.94
Mistral-7B + CoT + Finetune	0.94
Ensemble (Flan T5 + GPT-3.5)	0.91
Llama 2-7B + CoT	0.90
Flan-T5-LaMini	0.88
Finetuned Mistral-7B	0.86
GPT-3.5	0.77
GPT-3.5-Turbo	0.74

Table 1: Summary of previous Numerical Reasoning work in (Chen et al., 2024a).

2 Background

Table 1 provides a summary of the previous work done on the Numerical Reasoning task as part of the NumEval 2024 submissions. We note that our objective in this paper is not to outperform the state-of-the-art presented in previous work. Rather, we define our objectives as the following:

1. Evaluate the performance of low-resource approaches with fine-tuning relative to the existing literature.

2. Expand beyond the accuracy evaluation of the Numerical Reasoning task and demonstrate how additional metrics can provide additional insight.

As seen in Table 1, the methods that achieve the best results utilize large models while incorporating Chain of Thought reasoning, ensembling, data augmentation, and/or training on secondary tasks.

The best approach (Fan et al., 2024) employs Qwen-Chat, a 72 billion parameter model, trained on a GPU with 80 GB of memory. The authors employ data augmentation by randomly shuffling sentences; disrupting the coherence of the text proved effective for handling more complex mathematical problems. Finally, the authors incorporate secondary datasets of math problems to improve the mathematical ability of the base model, and learn to distinguish when to extract numbers versus when to perform calculations.

Many of the other approaches use GPT-3.5, and although the parameter count of this model is unknown, GPT-3, another model of the same family, contains 175 billion parameters (Brown et al., 2020). While some approaches use zero-shot prompting to achieve 74% and 77% accuracy (Alinejad and Moosavi Monazzah, 2024; Bahad et al., 2024), other approaches enhance GPT-3.5's capabilities through ensembling, fine-tuning, or Chain of Thought (COT) prompting to push the model to take intermediate reasoning steps before discerning the final output, which achieves higher accuracy (94% and 91%) than the zero-shot approach (Qian et al., 2024; Gonzalez et al., 2024). We note a drastic improvement of the few-shot approaches over the zero-shot approaches, which inspires us to explore if few-shot approaches on smaller models can outperform zero-shot approaches on large models.

Given the the compute resources on hand, we implement smaller models to demonstrate that effective few-shot learning on low-resource models can achieve superior results to zero-shot learning on large models and argue that small models still have a place in the modern landscape of Natural Language Processing. Although our objective is not to outperform the state-of-the-art work, this survey of the literature provides the reader background on how to improve on our approaches with more compute resources.

We identify two areas upon which we can build upon previous work. Firstly, the only BERT approach implemented is by (Crum and Bethard, 2024), who use DistilRoBERTa to reduce the BERT model size by 40% to 82 million parameters (Sanh et al., 2020). To further explore the capabilities of BERT on this task, we implement ModernBERT, proposed in December 2024, as a novel BERT approach to this task (Warner et al., 2024). ModernBERT aligns with our objective to approach the task in a computationally efficient manner as it implements alternating attention, flash attention, and unpadding, providing a model designed for inference on common GPUs. Despite the efficiency improvements, ModernBERT improves performance over BERT by using a modernized tokenizer while training on 2 trillion tokens and an 8192 sequence length. We implement ModernBERT-base, with a model size of 150 million parameters. Because ModernBERT follows a Masked Language Modeling setup for its training, it is an appropriate application for our task.

Secondly, the Flan-T5 model was used by multiple teams for headline generation. Crum and Bethard implemented a two-step approach where one T5 model was trained to produce calculation steps, and the second T5 model was trained to execute those calculations and generate the most ac-

News: At least 30 gunmen burst into a drug rehabilitation center in a Mexican border state capital and opened fire, killing 19 men and wounding four people, police said. Gunmen also killed 16 people in another drug-plagued northern city. The killings in Chihuahua city and in Ciudad Madero marked one of the bloodiest weeks ever in Mexico and came just weeks after authorities discovered 55 bodies in an abandoned silver mine, presumably victims of the country's drug violence. More than 60 people have died in mass shootings at rehab clinics in a little less than two years. Police have said two of Mexico's six major drug cartels are exploiting the centers to recruit hit men and drug smugglers, ...
Headline (Question): Mexico Gunmen Kill ___
Answer: 35
Annotation: Add(19,16)

Table 2: An example from NumHG.

curate number (Crum and Bethard, 2024). This allowed the overall model to perform better on more complex headlines. Meanwhile, Chen et al. trained Flan-T5 to produce Chain of Thought responses prior to the final answer. The weakness of these models is that they generate a large number of tokens during inference, costing more compute as well as time (Chen et al., 2024b). We aim to fine-tune base Flan-T5, with 248 million parameters, on the task directly, with far fewer examples, and thus lower time and cost during training as well as inference.

3 Methods

3.1 Data

We leverage the NumHG data to explore the Numerical Reasoning task. Table 2, which originates from the (Chen et al., 2024a) paper, is an example of the inputs and labels contained within. The original data included 97,110 training examples and 27,745 test examples split into five folds. Among the training examples, 67,684 were extractive in nature, meaning that they contained the correct answer within the context. However, 29,426 examples required further reasoning, such as addition or division, to produce the correct answer.

Operator	Description	Ratio
Copy(v)	Copy v from the article	65.00%
Trans(e)	Con[v]ert e into a number	17.37%
Paraphrase(v_0, n)	Paraphrase the form of digits to other representations	8.27%
Round(v_0, c)	Hold c digits after the decimal point of v_0	3.10%
Subtract(v_0, v_1)	Subtract v_1 from v_0	2.15%
Add(v_0, v_1)	Add v_0 and v_1	1.73%
Span(s)	Select a span from the article	1.34%
Divide(v_0, v_1)	Divide v_0 by v_1	0.54%
Multiply(v_0, v_1)	Multiply v_0 and v_1	0.50%

Table 3: Overview of predefined operators. v, v_0, and v_1 denote the selected numerals, and e denotes the English word. s and c denote a span from the article and a constant, respectively.

The NumHG data contains "reasoning annotations" which describe the operations needed to predict the correct headline number. Table 3, which originates from the NumHG overview paper (Huang et al., 2024), provides the full list of operations needed to generate the labels. The goal of this task is to train the models to learn how to do this reasoning for predicting the headline numbers.

Following the methodology found in (Crum and Bethard, 2024; Fan et al., 2024), the masked headline and the news article are concatenated to form the input text feature. One modification from the approach in (Crum and Bethard, 2024), which is consistent with (Fan et al., 2024), is that the reasoning annotation is not concatenated to the input text as well, as we aim to explore the numeric reasoning capabilities of our models rather than train the model to take a shortcut and make predictions based on the annotations.

In order to reduce time and compute costs, we sampled the dataset to produce three smaller datasets. From the test examples, we randomly sampled 2,774 (10%) test examples to produce a smaller test set. This test set was used to evaluate all the models and methodologies in our analysis. From the training examples, 9,771 (10%) were randomly sampled to produce a training set containing a mixture of extractive and reasoning examples (Mixed Training Set). Furthermore, to create a training set that solely included reasoning examples, we sampled 33% of the 29,426 reasoning examples in the original dataset to produce the Reasoning Training Set containing 9,711 examples.

3.2 Modeling

We generated predictions on baseline and fine-tuned ModernBERT (150M) and Flan-T5 (248M) base models on the Mixed Training Set and the Reasoning Training Set. The baseline models are simply pre-trained, out-of-the-box models from the HuggingFace library (Wolf et al., 2020). We continued to take advantage of the HuggingFace library, particularly its capabilities for managing compute resources, to conduct fine-tuning on the NumHG data on the Numerical Reasoning task. ModernBERT was set up to do masked token prediction, whereas Flan-T5 was prompted to predict the number replacing the [MASK] token with the following prompt: "Question: Based on the context, what number should replace the [MASK] token in the headline. Only provide the number as response. Headline: {1}. Context: {0}."

We employed fine-tuning to strengthen the performance on the specific task of predicting a missing number given a masked headline and the corresponding news article. For fine-tuning the Flan-T5 model, we leveraged full fine-tuning which involves applying gradient descent on all parameters of the model. For fine-tuning the ModernBERT model, we leveraged LoRA (Low-rank Adaptation of Large Language Models). LoRA was proposed in (Hu et al., 2021) as a method for parameter-efficient fine-tuning "which freezes the pre-trained model weights and injects trainable rank decomposition matrices into each layer of the Transformer architecture" and allows us to achieve effective fine-tuning results for reasoning on numbers with limited compute resources.

To supplement the models' reasoning capabilities beyond fine-tuning, we leverage a word-to-number library to convert numeric words to numbers when necessary. This is a simple approach to address one aspect of reasoning by building in understanding of word and number mappings, and serves to improve the accuracy of our models as all the labels exist in numeric form, rather than as word.

3.3 Metrics

For evaluation, we calculate Accuracy, Mean Absolute Percentage Error (MAPE), and Symmetric Mean Absolute Percentage Error (SMAPE).

Accuracy is a simple and interpretable metric to measure the fraction of correct predictions from the model; however, it penalizes heavily if the answer is slightly incorrect. Although none of the previous work on numerical reasoning evaluate their models on metrics aside from accuracy, we seek other measures which account for relative distance of the prediction from the actual label.

Headline: "Family Electricity Bill: $ [MASK] ." **Article:** "Imagine opening your electric bill and seeing a figure of $284,460,000,000 under the amount owed..." **Prediction:** 284 **Actual label:** 284460000000
Headline: "SC Company Laying Off All but [MASK] Workers Over Tariffs." **Article:** "The State reports TV-maker Element Electronics is citing the tariffs as the reason it is essentially closing its doors: It intends to shut down its Winnsboro plant and lay off 126 of its 134 employees..." **Prediction:** 126 **Actual label:** 8

Table 4: Examples of extreme under-prediction and over-prediction from ModernBERT. Example 1 also demonstrates a mis-predicted label because the actual label is not in the vocabulary.

One method to measure numeric distance is to calculate the MAPE which denotes how far the predictions were from the label as a percentage. If the prediction matches the label, the percentage error for that example is 0. MAPE tries to gather, on average, how close the predictions are to the labels. For MAPE, lower is better.

The MAPE metric has no upper bound, and some extreme outliers, such as the example in Table 4, can give rise to an extremely high MAPE. Therefore, we calculate the MAPE twice: once with all observations, and another excluding Percentage Error (PE) greater than 1000%. The examples with over 1000% PE are treated as outliers. This exclusion typically disregards 5-10% of the test data.

The main disadvantage of MAPE is its unequal treatment of predictions that are far smaller than the actual label versus predictions that are far greater than the actual labels. See Table 4 for examples. In the first example, where the prediction (284) is much lower than the actual label (284,460,000,000), the Percentage Error is $\frac{|284-284460000000|}{284460000000} \approx 100\%$. However, in the second example, where the prediction (126) is much larger than the actual label (8), the Percentage Error is $\frac{|126-8|}{8} = 1475\%$. The issue, which is often referred to as asymmetry, is that these MAPE calculations do not take into account the raw magnitude in which the predictions and labels differ. The difference between prediction and label is much higher for the first example than the second, however, the second had a much greater Percentage Error.

The Symmetrics Mean Average Percentage Error (SMAPE) metric was proposed to resolve the asymmetry issue (Makridakis, 1993). The metric calculates averages in two steps. First, for each prediction and label pair, we calculate the ratio of the prediction error and the average of the prediction and label. Afterwards, the average of each of these ratios is calculated over the entire dataset. In other words, this metric represents the average percentage error between predictions and actual values, relative to the average magnitude of both. Unlike MAPE, SMAPE is bounded between 0% and 200%. Below is the formula used to calculate SMAPE.

$$SMAPE = \frac{1}{n} \sum_{i=1}^{n} \left| \frac{\hat{y}_i - y_i}{(\hat{y}_i + y_i)/2} \right|$$

4 Results and Discussion

Table 5 summarizes the results of the various models evaluated in our work. "Excluded observations" included test observations for which we cannot calculate MAPE or SMAPE because: (a) The predicted label is non-numeric; (b) The actual label is non-numeric; or (c) The denominator is zero (invalid division). For MAPE, the denominator is the actual label, and for SMAPE, it is the average of the prediction and actual label. As previously specified, in addition to the baseline model, each model was trained on the Mixed Training Set and Reasoning Training Set pertaining to each row in the "Model" column. The metrics are reported in three groups: the "Overall test set" group presents the model evaluations on all the test data, the "Reasoning test set" group presents the model evaluations on just the reasoning examples from the test data, and the "Non-reasoning test set" group presents the model evaluations on just the non-reasoning examples from the test data.

We observe that the ModernBERT and Flan-T5 baseline models perform poorly on this task despite being pre-trained on various datasets and instruction tasks, with ModernBERT-Base having 52.2% accuracy and Flan-T5-Base having 38.1% accuracy. Both sets of fine-tuned models perform drastically better than the base models. The fine-tuned models improved in reasoning accuracy and non-reasoning accuracy. When trained on Mixed Training Set, both models performed better on the non-reasoning headlines when compared to the reasoning headlines. On the other hand, when trained on the Reasoning dataset, both models performed better on the reasoning headlines than the non-reasoning headlines. We observe that the drop in overall accuracy when training on the Reasoning Training Set compared to the Mixed Training Set is more drastic for Flan-T5 than for ModernBERT; the drop in accuracy emphasizes the importance of balanced training data to overall performance.

It is important to note that the overall accuracy is skewed as the test set contains more non-reasoning headlines than reasoning headlines. This is why we see that the biggest improvement in overall accuracy was achieved after training on the Mixed Training Set. While fine-tuning the models on the Reasoning Training Set helped it perform better on reasoning headlines, it did not improve the models' accuracy on the non-reasoning headlines as much and resulted in a smaller improvement on the overall test set relative to the models fine-tuned on the Mixed Training Set.

These findings are consistent when evaluating MAPE and SMAPE across the models, where the Reasoning Training Set improves performance on the reasoning test set but the trade off is worse performance on the non-reasoning test set and overall test set. The MAPE shows that when our fine-tuned models make incorrect predictions, the predictions are approximately 20-30% off overall from the actual label (without outliers). The SMAPE shows that when our fine-tuned models make incorrect predictions, the predictions are approximately 20-45% off from the average prediction/actual label magnitude.

To compare the two model types, after three epochs of training, the fine-tuned Flan-T5's accuracy improved more than the fine-tuned ModernBERT. We believe this is explained by the differences in fine-tuning strategy. ModernBERT was fine-tuned using LoRA, while Flan-T5 was fully fine-tuned. As a result, the parameters of the Flan-T5 model were changed to a greater extent than the parameters of the ModernBERT model. We also note that Flan-T5 does a much better job at

	Model	Accuracy	Excluded observations	MAPE (all)	MAPE (no outliers)	SMAPE
Overall test set	ModernBERT-Base	0.522	829	0.820	0.194	**0.217**
	ModernBERT-Base: Mixed reasoning ft	0.726	31	0.895	0.190	0.229
	ModernBERT-Base: Reasoning only ft	0.713	36	**0.429**	**0.175**	0.259
	Flan-T5-Base	0.381	33	731.62	0.721	0.815
	Flan-T5-Base: Mixed reasoning ft	**0.788**	26	3.70	0.228	0.282
	Flan-T5-Base: Reasoning only ft	0.633	26	1.246	0.301	0.448
Reasoning test set	ModernBERT-Base	0.484	285	0.675	0.241	0.273
	ModernBERT-Base: Mixed reasoning ft	0.724	1	2.194	0.221	0.237
	ModernBERT-Base: Reasoning only ft	0.751	1	**0.342**	**0.150**	0.198
	Flan-T5-Base	0.317	12	973.35	1.23	1.32
	Flan-T5-Base: Mixed reasoning ft	0.690	8	4.287	0.279	**0.272**
	Flan-T5-Base: Reasoning only ft	**0.745**	1	2.342	0.222	0.237
Non-reasoning test set	ModernBERT-Base	0.540	544	0.885	0.173	**0.192**
	ModernBERT-Base: Mixed reasoning ft	0.727	30	**0.283**	0.175	0.225
	ModernBERT-Base: Reasoning only ft	0.695	35	0.470	0.186	0.288
	Flan-T5-Base	0.410	21	169.04	0.617	0.748
	Flan-T5-Base: Mixed reasoning ft	**0.834**	18	0.385	**0.169**	0.219
	Flan-T5-Base: Reasoning only ft	0.581	25	0.748	0.330	0.536

Table 5: Model outcomes for predicting masked numerical headline on different samples of test data. ft refers to fine-tuning,

returning numeric answers; as an encoder-decoder model with prompting, we can explicitly tell the model what to return (i.e. a number only), whereas this is not possible with the ModernBERT setup. We found that fine-tuning beyond three epochs did not substantially improve results.

The Flan-T5 - Mixed reasoning model is strictly the best performing model trained for the Numerical Reasoning task in regards to accuracy. However, examining MAPE and SMAPE, it seems that when ModernBERT's predictions are incorrect, they are closer to the correct answer than Flan-T5's incorrect predictions. This may indicate that ensembling these models, as in (Gonzalez et al., 2024), would improve performance by covering these weaknesses. The Flan-T5-Base: Mixed Reasoning model also demonstrates an improvement over the zero-shot GPT-3.5 implementations in (Alinejad and Moosavi Monazzah, 2024; Bahad et al., 2024), even though it is a smaller model trained on limited training data.

One of the limitations of using language models for numerical tasks is tokenization and their behavior. Tokenizers are trained on large corpus of text. They identify the most frequent subwords or words for a given vocabulary size. As a result, a number can be represented by one or more tokens, and there is little consistency throughout the vocabulary. The number "708" can be represented as a single token or as a combination of "70" and "8". In ModernBERT, the masked LM classifier only chose the most probable token that would replace the missing number in the headline. Despite ModernBERT having a large vocabulary size of 50,386, there were cases where it was unable to predict the correct number because the number did not exist in its vocabulary as an individual token; see Table 4 as an example. Since Flan-T5 is an encoder-decoder model, it was able to use a combination of two or more tokens in its prediction. Numbers, unlike words, are continuous and infinite. Future studies may improve by taking from xVal, which proposes a novel approach to tokenizing where all numbers are represented as a single [NUM] token, but the numerical value is stored as a separate vector (Golkar et al., 2024). The [NUM] token is scaled based the magnitude and tends to improve accuracy on numerical tasks.

5 Conclusion

We explored the potential of small fine-tuned models for tackling the numerical reasoning subtask under the Headline Generation proposed in the SemEval 2024 NumEval challenge. By fine-tuning ModernBERT and Flan-T5 base models on a small sample of NumHG dataset, we improved their performance on the subtask. The Flan-T5 fine-tuned on the Mixed-Reasoning dataset achieved the highest accuracy of 78.8%. While Flan-T5 performed better than ModernBERT, fine-tuning ModernBERT was more computationally efficient due to LoRA and only performed marginally worse. We also attempted to evaluate Gemma3, Google's latest open-weights LLM, but were not able to fully experiment with it in the time allotted. For future experiments, we would like to explore other tokenization and embedding techniques that are better suited for representing numbers. Recent large language models have been trained on tool-use and programming. While the LLMs may not be able to solve math problems on their own, they may be able to utilize calculators and other tools to accurately perform arithmetic.

6 Author Contributions

Kumar and Liang worked on data processing and intake. Liang worked on research, fine-tuning, experimentation, and evaluation of ModernBERT models. Kumar worked on research, fine-tuning, experimentation, and evaluation of Flan-T5 models. Kumar provided research and implementation of MAPE, while Liang provided research and implementation of SMAPE. Kumar and Liang worked on write-up and final review of this paper and the presentation in the project GitHub.

References

Sina Alinejad and Erfan Moosavi Monazzah. 2024. Sina alinejad at SemEval-2024 task 7: Numeral prediction using gpt3.5. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 1087–1091, Mexico City, Mexico. Association for Computational Linguistics.

Sankalp Bahad, Yash Bhaskar, and Parameswari Krishnamurthy. 2024. Noot noot at SemEval-2024 task 7: Numerical reasoning and headline generation. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 913–917, Mexico City, Mexico. Association for Computational Linguistics.

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, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners.

Chung-chi Chen, Jian-tao Huang, Hen-hsen Huang, Hiroya Takamura, and Hsin-hsi Chen. 2024a. SemEval-2024 task 7: Numeral-aware language understanding and generation. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 1482–1491, Mexico City, Mexico. Association for Computational Linguistics.

Kaiyuan Chen, Jin Wang, and Xuejie Zhang. 2024b. YNU-HPCC at SemEval-2024 task 7: Instruction fine-tuning models for numerical understanding and generation. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 973–979, Mexico City, Mexico. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Hinoki Crum and Steven Bethard. 2024. hinoki at SemEval-2024 task 7: Numeral-aware headline generation (English). In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 34–39, Mexico City, Mexico. Association for Computational Linguistics.

Yuming Fan, Dongming Yang, and Xu He. 2024. CTYUN-AI at SemEval-2024 task 7: Boosting numerical understanding with limited data through effective data alignment. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 47–52, Mexico City, Mexico. Association for Computational Linguistics.

Siavash Golkar, Mariel Pettee, Michael Eickenberg, Alberto Bietti, Miles Cranmer, Geraud Krawezik, Francois Lanusse, Michael McCabe, Ruben Ohana, Liam Parker, Bruno Régaldo-Saint Blancard, Tiberiu Tesileanu, Kyunghyun Cho, and Shirley Ho. 2024. xval: A continuous numerical tokenization for scientific language models.

Andres Gonzalez, Md Zobaer Hossain, and Jahedul Alam Junaed. 2024. NumDecoders at SemEval-2024 task 7: FlanT5 and GPT enhanced with CoT for numerical reasoning. In *Proceedings of* *the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 1260–1268, Mexico City, Mexico. Association for Computational Linguistics.

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.

Jian-Tao Huang, Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2024. NumHG: A dataset for number-focused headline generation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 12323–12329, Torino, Italia. ELRA and ICCL.

Spyros Makridakis. 1993. Accuracy measures: theoretical and practical concerns. *International Journal of Forecasting*, 9(4):527–529.

Zhen Qian, Xiaofei Xu, and Xiuzhen Zhang. 2024. ZXQ at SemEval-2024 task 7: Fine-tuning GPT-3.5-turbo for numerical reasoning. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*, pages 218–223, Mexico City, Mexico. Association for Computational Linguistics.

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

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi 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 M. Rush. 2020. Huggingface's transformers: State-of-the-art natural language processing.