Infrrd.ai at SemEval-2024 Task 7: RAG-based end-to-end training to generate headlines and numbers

JiangLong He, Saiteja Tallam, Srirama Nakshathri,
Navaneeth Amarnath, Pratiba KR, Deepak Kumar
Infrrd.ai
{jianglong,saitejatallam,srirama}@infrrd.ai
{navaneethamarnath,pratibakr,deepakumar}@infrrd.ai

Abstract

We propose a training algorithm based on retrieval-augmented generation (RAG) to obtain the most similar training samples. The training samples obtained are used as a reference to perform contextual learning-based fine-tuning of large language models (LLMs). We use the proposed method to generate headlines and extract numerical values from unstructured text. Models are made aware of the presence of numbers in the unstructured text with extended markup language (XML) tags specifically designed to capture the numbers. The headlines of unstructured text are preprocessed to wrap the number and then presented to the model. A number of mathematical operations are also passed as references to cover the chain-of-thought (COT) approach. Therefore, the model can calculate the final value passed to a mathematical operation. We perform the validation of numbers as a post-processing step to verify whether the numerical value calculated by the model is correct or not. The automatic validation of numbers in the generated headline helped the model achieve the best results in human evaluation among the methods involved.

1 Introduction

In our busy lives, we barely have time to read the newspapers or an online article. Even a short period will not be enough to read all the latest news articles from different sources. The headline attached to the article attracts the reader only if it is interesting or provokes interest in the reader. A reader may have different interests and may not cover all areas. Some may be interested in movies, politics, science and technology, economy, environment, governance, sports, celebrities, weather, etc. The information fed to a reader is large and must be condensed and remembered. A unique headline condenses the unstructured text into a few words with some numbers. The numbers are presented to attract the reader's attention from the part of the unstructured text. These numbers can be based on positive or negative sentiments. A negative sentiment has a greater influence than a positive sentiment. A positive sentiment aims to create new information in the reader's mind. However, a negative sentiment harms the reader's mind by correcting and updating the information. Named entities, such as name, location, and number, are easier to remember for a long time than the rest of the text. The named entities are used more often by many people in several contexts with high-frequency usage. The narration in the text is constructed with the entities with a relationship. The occurrence of numbers in the relationship is more frequent than in other entities, especially in news articles.

We perform text summarization on unstructured text to obtain specific and highlighted information. It is helpful in many areas and reduces the time spent on unnecessary or irrelevant texts. Several events would occur in the process from the beginning to the end. All events may not be relevant or may appear as information overload. To reduce the list of events, we are using a text summary. Medical report summary, annual report summary, election results summary, movie reviews, product reviews, and sports reviews will highlight the main results of the research or the results of the conducted activities. A person must read the content to prepare a summary of the text. The likelihood is that many relevant points of the content should be included as part of the summary. Each person can prepare different lists of points using their previous knowledge and preferences. The main points of the different lists will be a central part of any summary. The core part can form a headline to attract readers to read the contents of the unstructured text.

Summary generation is a time-consuming process in which many people must contribute to preparing the highlights of the text content. In natural language processing (NLP), a model has to

Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024), pages 940–951

process text content sequentially using a seq2seq model and generate these highlights. It reduces the time required to generate highlights, increases knowledge aggregation, and filters interesting content. A vast literature of text summarization tasks in NLP shows the required attributes of a machine. A summary is presented in plain text in most cases and may not always contain a number. In the generation of headlines, we need a few numbers to highlight the content. The numbers in a headline play an important role in attracting readers' attention. Here, the model must process the unstructured text sequentially and locate the numbers. All numbers in the text cannot be part of the headline. Only a few numbers can cover the complete information from the unstructured text. A model has to identify the numbers that cover the news content in order to generate the headline (Cai et al., 2023; Ding et al., 2023; Zhang et al., 2020).

The main contributions of the proposed method are as follows:

- Retrieval-augmented generation (RAG) of training samples to fine-tune a large language model (LLM).
- Chain-of-thought (COT) based generation of mathematical operations by the model.
- Verification of the computed value by the model to increase the confidence of the extracted numbers from the unstructured text.

2 Related Work

Large language models (LLMs) have started to demonstrate reasoning, calculation, knowledge acquisition, planning, and many more (LLAMA2; MISTRAL; OpenAI). At this point, we need to explore the potential capabilities of LLMs by proposing a wide range of problems that deal with a kind of artificial intelligence embedded in the model. In this paper, we study the numerical ability of a model.

EQUATE benchmark was prepared to make quantitative reasoning on different measures in the natural language inference (NLI) (Ravichander et al., 2019). The data set is prepared to understand whether a model can reason on the text. The results show that the models have the ability to reason and obtain an inference for the statements. The scale of the predicted number was done to understand whether a model can find the magnitude of the predicted numbers through an NLP (Chen et al., 2019). A language model does not explicitly distinguish numbers from words (Chen et al., 2023). The notation of a number cannot be clearly understood by the model. This can be due to a missing number in the training data. We cannot provide all numbers to the model by any means, so we need to use different symbols to express the numbers in the text so that the model can use the numbers to perform the reasoning tasks. However, there was scope to add additional challenges to the data set. The NumGLUE was proposed to identify the performance of LLMs through natural language understanding (NLU) (Mishra et al., 2022). There are eight different tasks based on common sense reasoning, arithmetic calculation, quantitative prediction, fill-in-the-blanks, and arithmetic word problems. Natural language optimization (NLOpt) is a competition to solve arithmetic word problems for linear programming problems (Ramamonjison et al., 2023). We proposed an ensemble approach to detect the named entities (NEs) in an unstructured text (He et al., 2022). The solution was generic and detected most of the entities in the text. Of all these tasks, the headline generation or summarization focused on numbers was missing.

The NumHG data set was prepared to cover the headline generation task by focusing on numbers (Chen et al., 2024, 2021; Huang et al., 2023). The NumEval competition is held to evaluate different models that can understand the numbers and generate the headline according to the ground truth specified (NumEval). The model can choose any random number from the text, which may not be relevant in many cases. The model is pushed to perform the calculation that provides the fill-in-the-blank task, where the model has to calculate the missing number from the headline or summary. A model must use mathematical operations to get the answer. The computational ability of the model is explored in this approach and is known as the task of 'numerical reasoning.' Several mathematical operations can be performed using a model. However, the news data set mainly covers the reproduction of the number, the conversion of a word into a number, and the rounding of the number. The distribution of mathematical operations is very narrow. We do not have sufficient samples for other types of mathematical operations, and the model may not be well suited for these types of operations even after fine-tuning. The data set is designed in such a way

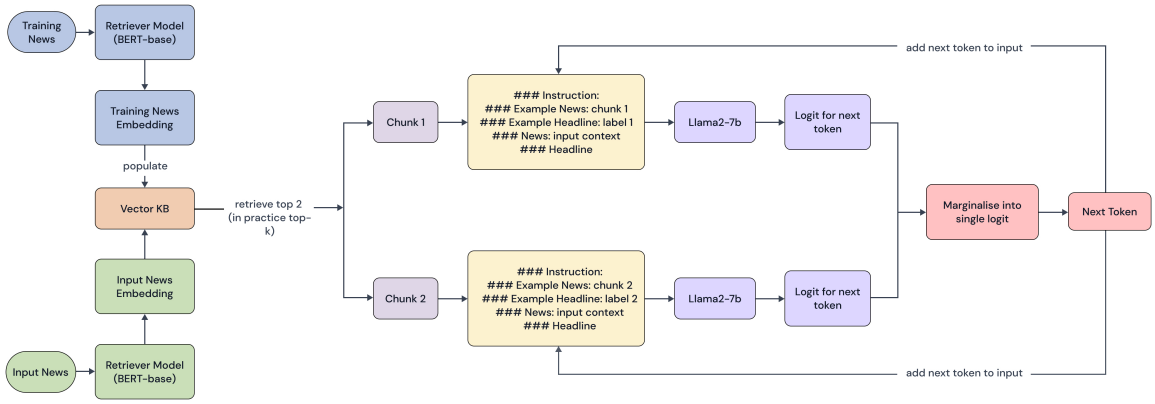

Figure 1: The retrieval-augmented generation (RAG) approach is used in the proposed method to generate headlines and numbers. A vector database is populated using the training news of the NumHG data set, and the training news is retrieved from the database during the inference. The prompted text is as follows: ### Instruction: Generate a Headline for the News and generate Numerical Reasoning for the numbers in the generated headline. Wrap the Numerical Reasoning with XML tags <NR> </NR>. Use the Example News and Example Headline as references. ### News: Input News. ### Headline: .

Table 1: The augmentation of headlines using the chain-of-thought (COT) approach. The NumHG data set provides arithmetic operations in the "calculation" field to be used for numerical reasoning tasks, which are fill-in-the-blanks tasks. The arithmetic operations from the numerical reasoning task are used as a reference to perform augmentation in the ground truth headline.

Ground truth headline	Guy Beat by Police Gets $1K, Lawyers Get $459K
Augmented headline	Guy Beat by Police Gets $1K <NR> paraphrase(1,000,K) </NR>, Lawyers Get $459K <NR> paraphrase(Add(100000,359000),K) </NR>

that the model must be aware of all the numbers presented in the text. Then, the model has to select a few numbers and make the calculation. The numbers are not known, and the mathematical operation is not known to the model in this NumEval task. The model must artificially identify the numbers and choose an appropriate mathematical operation to generate an answer.

3 Proposed method

A given LLM may not have full knowledge of the generation of headlines using a text. We need to provide some support to the model to excel in the task of headline generation. We propose a retrieval-augmented generation (RAG) model that is trained from end to end. The system consists of three modules: Knowledge Base, dense retrieval, and generation modules. Figure 1 shows the block diagram of the proposed approach.

The Knowledge Base (KB) was built using the training data provided. A pre-trained BERT base model is used to encode training news samples into vectors (Devlin et al., 2018). Then, these vectors are indexed using FAISS to enable the task of searching for vector-based similarity (Douze et al., 2024). The same BERT-based model is used during training and inference time. A given input text is encoded in a vector. Encoded vectors are used to search for dense vector similarity. Top-k similar news articles are retrieved by a dense retrieval module. Each result obtained will be added to the original input in a specific template to generate the headline. The purpose is to provide similar examples to generative models to help generate a better result. The prompt used in the experiment is shown in Figure 1. We use the LLAMA2-7b model to generate the output headline (Touvron et al., 2023; LLAMA2). We use the RAG token model (RAG) on the selected Top-k retrieved examples as shown in Figure 1. Each retrieved news is sent along with the input news to obtain two separate token predictions from the LLAMA2-7b model. These token predictions are marginalized, and the process is repeated until all tokens are generated.

Table 2: The ROUGE scores of different models using simple prompt to generate the headlines. The prompt is "Generate a headline for the following passage." The second row in the method name indicates the type of data set and the number of samples in the data set used for evaluation.

Method Name	ROUGE-1	ROUGE-2	ROUGE-L
ChatGPT (gpt-4-1106-preview) (Dry run - 100)	37.61	12.53	32.25
ChatGPT (gpt-4) (Dry run - 100)	36.37	12.25	30.56
ChatGPT (gpt-3.5-turbo) (Dev set - 2365)	35.44	13.16	31.08
LLAMA2-7b (Dev set - 2365)	11.78	4.41	10.37

Table 3: The ROUGE scores of different models using in-context learning approach to generate the headlines. The second row in the method name indicates the type of data set and the number of samples in the data set used for evaluation.

Method Name	ROUGE-1	ROUGE-2	ROUGE-L
Llama-2-13b-chat-hf_results 13b parameter model was used instead of 7b parameter model. The RAG examples were from the dry run set of 100 examples. (Dev set - 2365)	40.98	17.09	36.19
ChatGPT (gpt-3.5-turbo) - BM25 approach The training dataset words are stored in BM25. The search was performed using the development set. (Dev set - 2365)	40.67	17.11	36.15
openbuddy-llama2-70B-v13.2-AWQ_results 70b parameter model was used instead of 7b parameter model. The RAG examples were from the dry run set of 100 examples. (Dev set - 2365)	40.56	16.44	35.90
Mistral-7B-Instruct-v0.2_results Different model with same parameter size is used. The RAG examples were from the dry run set of 100 examples. (Dev set - 2365)	40.48	16.15	35.59
ChatGPT (gpt-3.5-turbo) - RAG examples The training dataset is converted into vectors using Fasttext approach. The similarity search was performed. (Dev set - 2365)	40.41	16.53	35.71
Llama-2-7b-chat-hf_results The RAG examples were from the dry run set of 100 examples (Dev set - 2365)	40.19	16.42	35.62

$$p_{RAG-Token}(y|x) \approx \prod_i^N \sum_{z \in top-k(p(.|x))} p_\eta(z|x) p_\theta(y_i|x, z, y_{1:i-1}) \quad (1)$$

where input sequence x, retrieve documents z, target sequence y. A retriever $p_\eta(z|x)$ with parameter η (BERT-base model) and a generator $p_\theta(y_i|x, z, y_{1:i-1})$ with parameter θ (LLAMA2-7b model). The current token is generated using the previous $i-1$ tokens $y_{1:i-1}$.

3.1 Training

We trained the model from end to end to optimize both the retriever and the generative model together. Here, we aim to train a numerically literate model, which means that the model should be able to understand the numbers in the news and be able to perform mathematical operations on these numbers to arrive at a precise numerical value that will be used in the headline. We encapsulate the numbers in the headline of the news text with XML tags, as shown in the example in Table 1. The tags are an easier way to instruct the model to locate the numbers instead of the model itself looking for the numbers. The content within the XML tags is annotated in the training dataset of the NumHG data set under the numerical reasoning tasks as a "calculation" field. The headline generation task provides a headline as a ground truth. We introduce the information from the numerical reasoning task to the

Table 4: The ROUGE scores of different approaches used to improve the fine-tuned model. The details of number of samples used in the training set. The development set was used for evaluation. The performance improvements with the followed approaches is expressed mnemonically.

Method Name	ROUGE-1	ROUGE-2	ROUGE-L
Dev_Headline_Generation_RagEnd2End + Post_processing The number of training samples used are 14,720. All the numbers in the ground truth headlines are wrapped with XML tags for numerical reasoning task. BRIO generated outputs are used as post- processing step to replace problematic headlines. (Dev set - 2365)	48.08	23.06	43.32
Dev_Headline_Generation_RagEnd2End + XML tags The number of training samples used are 14,720. All the numbers in the ground truth headlines are wrapped with XML tags for numerical reasoning task. (Dev set - 2365)	48.01	23.03	43.32
Dev_Headline_Generation_RagEnd2End + 2nd Inference The number of training samples used are 14,720. All the numbers in the input news are wrapped with XML tags for numerical reasoning task. The second time inference is performed with retrieval using generated headline and news. (Dev set - 2365)	47.56	22.71	43.12
Dev_Headline_Generation_RagEnd2End The number of training samples used are 14,720. All the numbers in the input news are wrapped with XML tags for numerical reasoning task. (Dev set - 2365)	47.46	22.85	43.01
Dev_Headline_Generation_Bart-Large (3 epochs) (Dev set - 2365)	46.40	21.88	41.19
Dev_Headline_Generation_Brio (74 epochs) (Dev set - 2365)	46.08	21.14	40.53

headline generation task. In this respect, we use the "calculation" field in the data set to augment the headline of each training sample. By doing so, we explain the calculation of numbers with the Chain-of-thought (COT) generated in the headline (Wei et al., 2023). Table 1 shows the ground truth headline and the augmented headline.

3.2 Hyperparameters

The model was trained using both the training and the development set for the final submission. The model was trained for 3 epochs on a single A100 GPU with a batch size of 5. The number of documents retrieved is set to 3 for the training period and 5 for the inference period. The learning rate is set to 2e-4, and the linear decay warm-up scheduler is used as a learning rate scheduler with 30 warm-up steps. We use greedy search during inference time to speed up the execution.

We have used LORA (Hu et al., 2021) to fine-tune the LLAMA2-7b model. LORA reduces the number of trainable parameters and memory requirements. The LORA configuration used for the final submission of the headline generation task is as follows: "r": 8, "lora_alpha": 16, "lora_dropout": 0.05, "target_modules": ['gate_proj', 'up_proj', 'o_proj', 'v_proj', 'q_proj', 'k_proj', 'down_proj'].

4 Experiments

LLMs have shown the ability to perform well in unknown tasks even without training. Therefore, it is useful to check whether the model is well suited for the NumEval dataset.

4.1 Out of the box

We have tested several LLMs, and the results are presented in Table 2. Most models perform almost similarly in out-of-the-box scenarios. GPT4 used to take a long time to generate a response. Hence, we experimented on the dry run set provided by the NumEval competition. No examples were provided with inputs to the model. The model must depend on its internal knowledge to generate a headline. The ROUGE scores tabulated above show that the

Table 5: The accuracy of the different methods for the numerical reasoning task. The prompts used for the generation of answer may result in different score.

Method Name	Accuracy (%)
Out-of-the-box LLAMA2-7b	6.53
Out-of-the-box ChatGPT (gpt-3.5-turbo)	43.90
Fine-tuning LLAMA2-7b	86.31
Chain-of-thought (COT)	89.54
COT+under_sampling	82.85
COT+over_sampling	89.00
COT+minority_combined	91.44

model depends on its knowledge to generate a headline. The numerical reasoning task was performed using the fill-in-the-blanks task approach and the out-of-the-box results for LLAMA2 were very low compared to ChatGPT models. The results are summarized in Table 5.

4.2 In-context learning

We use the retrieval-augmented generation (RAG) approach to perform context-based learning. Here, we supplement the input with the retrieved examples. The model must understand the examples shown to generate a headline. The examples provided by the RAG approach are crucial. The number of samples in the knowledge base affects the performance of the model. We have tabulated the ROUGE scores for the RAG-based generation of headlines using different models. There is certainly an improvement from out-of-the-box to context learning. The difference is high in terms of the scores tabulated in Table 3.

4.3 Fine-tuning

The fine-tuning of the LLAMA2-7b model was performed using the RAG method from end to end. The other models were also fine-tuned using different sample sets in the knowledge base (KB). The score was not so much improved compared to LLAMA2-7b. In addition, the BRIO model was trained with different headlines generated by different models from Table 3. The fine-tuned BRIO model also provided performance closer to the best performance method (Liu et al., 2022). However, it was used in the post-processing stage to add the headline if the main model did not perform the mathematical operations correctly. The results of the fine-tuned models are presented in Table 4.

Table 6: The numerical accuracy of different competing methods on the test set.

Position	Team Name	Accuracy Private Leaderboard
1	CTYUN-AI	0.95
2	Zhen Qian	0.94
3	YNU-HPCC	0.94
4	NCL_NLP	0.94
5	NumDecoders	0.91
6	**Infrrd.ai**	**0.90**
7	Hc	0.88
8	NLPFin	0.86
9	NP-Problem	0.86
10	AIRah	0.83
11	Noot Noot	0.77
12	GPT-3.5 (Baseline)	0.74
13	Sina Alinejad	0.74
14	StFX-NLP	0.60

4.4 Chain-of-thought (COT)

The numerical reasoning task requires the collection of the input text to perform the calculation. We provide a series of steps as instructions for the model. Suppose the model has to extract two numbers, say 19 and 16, and then perform addition to calculate the final score. We provide the model with a mathematical operation such as ADD(19,16) in the reasoning steps to get the final answer such as 35. The annotation of the input text to generate the instructions was simple in the NumEval data set. However, the model did not complete the calculations for some samples. Sometimes the model would not provide the final answer. The failure is unknown but based on the last step taken by the model. The last step was completed when the answer was empty. The COT-based reasoning improved the accuracy of the model compared to the LLAMA2-7b fine-tuned model. The results are tabulated in Table 5. The COT approach is superior compared to the fine-tuned model in the numerical reasoning task.

The results are tabulated in tables. 1-4 are based on experiments carried out on the development set. These experiments are conducted to identify the appropriate tools to help improve the performance of the model. The results show that current LLMs may not know completely how to generate a headline for a given text piece. The model needs the support of examples to generate headlines. The model also requires fine-tuning to reproduce answers closer to the ground truth.

Table 7: Automated evaluation of headline generation performed on the results of the test set.

Team Name	Overall	Num Acc. Copy	Reasoning	ROUGE 1	ROUGE 2	ROUGE L	BERTScore P	BERTScore R	BERTScore F1	MoverScore
ClusterCore	38.233	51.571	13.942	33.467	11.837	28.927	31.876	42.232	37.026	56.405
Noot Noot	38.393	57.481	3.6331	31.47	11.139	27.284	25.389	43.977	34.539	55.559
Infrrd.ai	**65.840**	**68.354**	**61.263**	**46.789**	**22.36**	**42.095**	**51.005**	**47.260**	**49.134**	**59.731**
np_problem	73.487	76.908	67.257	39.816	17.577	34.339	27.800	48.557	37.816	57.024
hinoki	62.347	66.284	55.177	43.072	19.719	38.999	47.223	43.444	45.342	58.711
Challenges	72.956	82.170	56.176	31.220	12.235	26.859	19.530	47.559	33.132	55.362
NCL_NLP	62.122	65.536	55.904	43.506	19.388	38.878	46.402	45.039	45.734	58.861
YNU-HPCC	69.044	73.018	61.807	48.852	24.681	44.175	51.553	50.095	50.381	60.551
NoNameTeam	55.715	57.681	52.134	40.646	17.261	35.745	44.256	40.387	42.324	57.736

5 Discussion and Results

The results of the test set based on the proposed method are presented in Tables 6, 7 and 8. The number of samples in the numerical reasoning task is 4921, and the number of samples in the headline generation task is 5227. The same proposed approach is used to estimate the results of the test set. The number of epochs used to train the model for the numerical reasoning task and the headline generation task are 1 and 3, respectively. Since LLMs consist of a large number of parameters, it is very difficult to make any internal changes. We could at least change some blocks that are connected to the LLMs either on the input or the output side. The changes to these blocks will be discussed in this section.

Table 8: Human evaluation of headline generation using reward points awarded by the human evaluator on the selected test set samples.

Team Name	Num Acc. (50 Headlines)	Recommendation (100 News)
ClusterCore	1.60	31
Noot Noot	1.68	11
Infrrd.ai	**1.81**	**22**
np_problem	1.57	14
hinoki	1.67	16
Challenges	1.70	10
NCL_NLP	1.73	16
YNU-HPCC	1.69	15
NoNameTeam	1.59	12

5.1 Numerical reasoning

The basic LLAMA2-7b model used in the out-of-the-box approach did not perform well in the fill-in-the-blank task. However, after fine-tuning the LLAMA2-7b model. The model was able to generate the answer for most of the samples correctly. However, there was still a gap in achieving higher numerical accuracy. The chain-of-thought (COT) approach is used to improve the model's accuracy. The improvement was marginal but observable in terms of numerical accuracy. The chain of thought forced the model to perform numerical reasoning in steps. The model would not complete some of the steps, but the answer was better than the simple fine-tuning approach. When the model fails to complete the last step, the answer is obtained through automated calculation. In addition to COT, we also conducted experiments for minor samples like undersampling and oversampling to train the model. The score was almost the same, and the changes were minimal. The higher the number of parameters (13b) in the model, the better the result than the smaller number of parameters (7b) in the model. Finally, we used the RAG-based approach to complete the fill-in-the-blank task. The combination of RAG and COT improved the model's ability to generate answers more accurately than the other approaches tested during the training period. Tables 5 and 6 show numerical accuracy on the development set and test set, respectively. We performed post-processing on the numerical value generated by the model by filling the empty values through understanding mathematical operations, but the numbers used by the model were not correct in most cases, leading to a wrong value and not improving the numerical accuracy.

5.2 Headline generation

We began to test the ability of LLMs to generate headlines using the out-of-the-box approach. The ROUGE scores were not satisfactory compared to the competition benchmarks (Huang et al., 2023). We used a context-based learning approach with which LLMs could understand the given examples and generate a much better headline. Even then, the ROUGE scores were below the benchmark. We

started fine-tuning the LLMs and got closer to the benchmark. We evaluated various types of models in the development set using all possible combinations. One interesting thing to identify is the numerically aware LLMs. Initially, we placed XML tags in the input text which provided a small improvement in the ROUGE scores in Table 4. We also performed inference for the second time using the first generated headline as part of RAG, which also provided a small improvement in the scores. Finally, we placed XML tags around the numerical values in the headlines to execute the COT approach. LLMs could understand that the answer should be selected from the input text and generate a mathematical operation that can complete the final answer. This gave the best score in the development set. Tables 7 and 8 show automated and human evaluation metrics for the submitted methods. The proposed method is second for most performance measures that are automatically calculated. In human assessment, the proposed method ranks first and second in terms of numerical accuracy and recommended headline, respectively.

5.2.1 Post-processing

We believe that the main contribution to numerical accuracy in Table 8 is verification. Verification is an important step in the process of finalizing the answer, which automatically improves confidence. One of the reasons for the verification is to select the numbers of the input text and compare them in the generated headline. There may be many numbers, but all of them cannot be used in the generation of headlines. Only a few numbers are used to complete the numerical calculation. Sometimes the model does not complete the calculation. We need to verify the steps followed by the model and fix some of the steps to improve the performance of the model in an automated way. The model generates the mathematical operations in an XML tag. The mathematical operations are processed with the numbers, which are part of the operation to verify that the number generated by the model is correct. We provide a few examples without errors in Appendix A.1 and with errors in Appendix A.2 in the verification stage. If the model fails to verify the generated numbers with the actual mathematical operation, a headline generated by the BRIO model is used to replace the headline generated by the main model (Liu et al., 2022). The number of samples with the replaced headlines is less than one percentage of the total number of samples in the test set. The percentage metric aligns with the top-2 reported results for the development set in Table 4. We were unable to fully explore the ensemble approach for LLMs but tried to combine the results of multiple models. A series of simple rules have been used to check and replace the headlines.

We instructed the model to generate a list of headlines, and then a new model was trained using reinforcement learning with human feedback (RLHF) (Böhm et al., 2019). The ROUGE scores from the RLHF-based model were better than a single headline generator but much less than the context-based learning approach. So, we have not reported the ROUGE scores for the RLHF approach. We are fine-tuning the model to confirm the ground truth, which may seem overfit for samples and deviate from the generalization capability of the model. Instead of a single ground truth headline, if we generate at least three headlines for the given input text similar to RLHF, that may help us understand whether the model falls into the category of generating the most common headline among the three. The performance measures will also change with multiple headlines as the ground truth. We speculate that a human factor would be added if several headlines were used as the ground truth rather than a single headline that is more like a robotic approach.

6 Conclusion

We proposed a RAG-based fine-tuning of LLMs to generate headlines and numerical values through reasoning. The model was trained from end to end to optimize the output of results. The model was trained for 3 epochs for headline generation and 1 epoch for numerical reasoning. We would like to train the model for a longer number of epochs in the future to confirm whether the model can improve performance. The verification step used to validate the generated numbers by the model is very useful to improve the confidence of the generated headline. There may be several numbers in the news, but the extraction and verification of numbers that can contribute to headline generation is a more concentrated approach. The additional verification stage helped the human evaluator select our proposed methodology as the most efficient among the competitors. We would like to explore further the rationale steps followed by the model in COT and improve the model performance by taking advantage of mathematical operations. We

would like to divide operations into a subset of operations. A combination of results from multiple models is attempted without fully exploring the ensemble approach. We would like to explore the possibility of combining models at different levels such as input, architecture, and so on.

References

Florian Böhm, Yang Gao, Christian M. Meyer, Ori Shapira, Ido Dagan, and Iryna Gurevych. 2019. Better rewards yield better summaries: Learning to summarise without references. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3110–3120, Hong Kong, China. Association for Computational Linguistics.

Pengshan Cai, Kaiqiang Song, Sangwoo Cho, Hongwei Wang, Xiaoyang Wang, Hong Yu, Fei Liu, and Dong Yu. 2023. Generating user-engaging news headlines. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3265–3280.

Chung-Chi Chen, Hen-Hsen Huang, and Hsin-Hsi Chen. 2021. Nquad: 70,000+ questions for machine comprehension of the numerals in text. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, CIKM '21, page 2925–2929, New York, NY, USA. Association for Computing Machinery.

Chung-Chi Chen, Hen-Hsen Huang, Hiroya Takamura, and Hsin-Hsi Chen. 2019. Numeracy-600K: Learning numeracy for detecting exaggerated information in market comments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6307–6313, Florence, Italy. Association for Computational Linguistics.

Chung-Chi Chen, Jian-Tao Huang, Hen-Hsen Huang, Hiroya Takamura, and Hsin-Hsi Chen. 2024. Semeval-2024 task 7: Numeral-aware language understanding and generation. In *Proceedings of the 18th International Workshop on Semantic Evaluation (SemEval-2024)*. Association for Computational Linguistics.

Chung-Chi Chen, Hiroya Takamura, Ichiro Kobayashi, and Yusuke Miyao. 2023. Improving numeracy by input reframing and quantitative pre-finetuning task. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 69–77, Dubrovnik, Croatia. Association for Computational Linguistics.

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*.

Zijian Ding, Alison Smith-Renner, Wenjuan Zhang, Joel R Tetreault, and Alejandro Jaimes. 2023. Harnessing the power of llms: Evaluating human-ai text co-creation through the lens of news headline generation. *arXiv preprint arXiv:2310.10706*.

Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The faiss library.

JiangLong He, Mamatha N, Shiv Vignesh, Deepak Kumar, and Akshay Uppal. 2022. Linear programming word problems formulation using ensemblecrf ner labeler and t5 text generator with data augmentations.

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. 2023. Numhg: A dataset for number-focused headline generation. *arXiv preprint arXiv:2309.01455*.

Yixin Liu, Pengfei Liu, Dragomir Radev, and Graham Neubig. 2022. BRIO: Bringing order to abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2890–2903, Dublin, Ireland. Association for Computational Linguistics.

LLAMA2. Llama2. `https://huggingface.co/docs/transformers/model_doc/llama2`. Accessed: 2024-02-14.

Swaroop Mishra, Arindam Mitra, Neeraj Varshney, Bhavdeep Sachdeva, Peter Clark, Chitta Baral, and Ashwin Kalyan. 2022. NumGLUE: A suite of fundamental yet challenging mathematical reasoning tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3505–3523, Dublin, Ireland. Association for Computational Linguistics.

MISTRAL. Mistral. `https://mistral.ai`. Accessed: 2024-02-14.

NumEval. Numeval competition. `https://sites.google.com/view/numeval/numeval`. Accessed: 2024-02-14.

OpenAI. Openai. `https://openai.com`. Accessed: 2024-02-14.

RAG. Rag. `https://huggingface.co/docs/transformers/en/model_doc/rag`. Accessed: 2024-02-19.

Rindranirina Ramamonjison, Timothy T. Yu, Raymond Li, Haley Li, Giuseppe Carenini, Bissan Ghaddar, Shiqi He, Mahdi Mostajabdaveh, Amin Banitalebi-Dehkordi, Zirui Zhou, and Yong Zhang. 2023. Nl4opt competition: Formulating optimization problems based on their natural language descriptions.

Abhilasha Ravichander, Aakanksha Naik, Carolyn Rose, and Eduard Hovy. 2019. EQUATE: A benchmark evaluation framework for quantitative reasoning in natural language inference. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 349–361, Hong Kong, China. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

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.

Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, and Xueqi Cheng. 2020. Structure learning for headline generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9555–9562.

A Appendices

We present a few examples from the verification stage. The examples provide insight into the verification stage followed in the proposed method.

A.1 Examples without errors in the verification stage

A.1.1 Example 1

News: (Jan 23, 2008 3:50 PM) A blast in the northern Iraqi city of Mosul claimed at least 15 lives and left 132 wounded today, Reuters reports. The explosion came after Iraqi troops detonated a roadside bomb near a building used as a munitions dump by militants, a US general said. At least 15 tons of explosives are estimated to have been stored in the building. Witnesses reported the resulting explosion was one of the biggest ever seen in Mosul, which, along with other areas of Nineveh province, has become a center for Sunni extremist fighters following their expulsion from Baghdad and the western Anbar province. A separate attack near Kirkuk, also in the north, killed 16 today, Reuters adds.

Pred_seq: Iraqi Blast Kills 15 <NR> Copy(15) </NR> in Mosul

Processed_headline: Iraqi Blast Kills 15 in Mosul

The method searches for any XML tag in the generated sequence. Here, the verification has found a tag. The method searches for the number '15' in the text because it is copied directly from the news section. The number '15' is found by the verification stage and removes the XML tag present in the generated headline.

A.1.2 Example 2

News:(Apr 22, 2009 9:11 AM CDT) Sick of waiting for a grand jury to act, Kathleen Savio's family has filed a wrongful death suit against husband Drew Peterson, the Chicago Tribune reports. The grand jury, which expires next month, has been investigating the death of Savio, Peterson's third wife, and the disappearance of fourth wife Stacy Peterson since November 2007. The Savios have hired John Kelly, the high-powered attorney who won the Brown family's civil suit against OJ Simpson, to represent them. We felt that the current grand jury is going to bring forward an indictment, or they'll be dismissed and we can proceed, said Kelly, who acknowledged it was unusual to file a civil case while a grand jury is still meeting. Defense attorneys will now be able to subpoena evidence from the criminal case for the civil one—but the state action will take precedence if the grand jury does return an indictment.

Pred_seq: Savio Family Files Wrongful Death Suit Against Drew Peterson

Processed_headline: Savio Family Files Wrongful Death Suit Against Drew Peterson

The method searches for any XML tag in the generated sequence. Here, the verification passes the generated headline without any changes due to the absence of XML tags.

A.1.3 Example 3

News: (Jul 31, 2012 3:13 AM CDT) The American Red Cross is facing a blood donation shortage serious enough that some people may have to cancel

elective surgeries. Donations are at their lowest in 15 years. In June, the organization came up 50,000 units short. July isn't looking good, either—a particular problem given the increased number of accidents typical in the summer, when people travel, a rep tells NBC News. Storms in the eastern and midwestern US both increased demand and cut supply, as the Red Cross was forced to cancel drives. With students, who account for 20% of donations, donating far less in the summer, the problem is compounded. We normally try to keep a three-day supply on hand locally, and we are down to a one—day supply, warns an Ohio Red Cross worker. And the need never, ever goes away, notes another representative.

Pred_seq: Red Cross Faces Worst Blood Shortage in 15 <NR> Copy(15) </NR> Years

Processed_headline: Red Cross Faces Worst Blood Shortage in 15 Years

The method searches for any XML tag in the generated sequence. Here, the verification has found a tag. The method searches for the number '15' in the text because it is copied directly from the news section. The number '15' is found by the verification stage and removes the XML tag present in the generated headline.

A.2 Examples with errors in the verification stage

A.2.1 Example 1

News: (Dec 20, 2016 5:40 PM) Forty-three days after the election, all the votes have finally been tallied and certified. History will show Hillary Clinton beating Donald Trump by a final count of nearly 3 million votes, the Hill reports. According to a tweet Tuesday from the nonpartisan Cook Report, Clinton received 65,844,610 votes (48.2%) to Trump's 62,979,636 (46.1%). However with the Electoral College officially making Trump the 45th US president on Monday, history will also show Clinton as the second Democrat in the past five elections to win the popular vote but lose the presidency. Meanwhile, the Huffington Post reports Trump had the third worst popular-vote performance by a winning candidate on record.

Pred_seq: 43 <NR> Subtract(43,1) </NR> Days After Election, All Votes Have Been Counted

Issues: Wrong Subtracted Value ('43', '<NR> Subtract(43,1) </NR>') | (43, 42), Value 43 not found in Snippet

Selected_headline: History Will Show Clinton Won Popular Vote by 3M Votes

The method searches for any XML tag in the generated sequence. Here, the verification has found a tag. When we search for the number '43', it is not found in the news section. We discard the generated headline and select the headline generated by the BRIO model as the generated headline.

A.2.2 Example 2

News: (Jul 12, 2014 4:20 PM CDT) Here's a dream come true for couch potatoes: You're not going to have to stop watching this movie for an entire month when it is ultimately released. Ambiancé, by Swedish director Anders Weberg, will be a full 720-hour film to be screened only once, starting on New Year's Eve in the year 2020, on every continent simultaneously, and then destroyed, reports The Verge. The filmmaker, who says this movie will be his last, has just released a teaser on Vimeo that is a short 72 minutes long, or roughly the length of a normal film. The short trailer, due out in two years, will be 7 hours and 20 minutes long (notice a theme?), reports Smithsonian magazine. The final trailer, out in 2018, will be 72 hours long. The film, a surreal dream-like journey beyond places, according to the film's site, is a sort of memoir that gives an abstract nonlinear narrative summary of the artist's time spent with the moving image. And it's expected to set the record as the longest film ever made. The teaser is only available until July 20. (Meanwhile, it's not too late to catch part of the world's longest concert.)

Pred_seq: World's Longest Film Will Be Destroyed After 1 <NR> Subtract(2020,2014) </NR> Screening

Issues: Wrong Subtracted Value ('1', '<NR> Subtract(2020,2014) </NR>') |(1,6), Subtraction operation is failed

Selected_headline: Longest Film Ever Made, 720-Hour Movie to Be Made, Destroyed

The method searches for any XML tag in the generated sequence. Here, the verification has found a tag. But, the model has failed to complete the subtraction operation. We discard the generated headline and select the headline generated by the BRIO model as the generated headline.

A.2.3 Example 3

News: (Dec 20, 2008 3:27 PM) The US could double the number of its troops in Afghanistan to about 60,000 by next summer, the highest such estimate to date, Reuters reports. American troops

currently number 31,000, but joint chiefs chairman Mike Mullen said today that an additional 20,000 to 30,000 will be needed by the spring or early summer. We're going to fill that requirement so it's not a matter of if, but when, he said. The majority will be deployed to the fragile south of the country, and Mullen was candid about the dangers. When we get additional troops here, I think the violence level is going to go up, he said. The fight will be tougher. Mullen also stressed that normalizing relations between Pakistan and India would ease violence in Afghanistan, and that any military progress must go hand in hand with economic development.

Pred_seq: US Could Double Troops in Afghanistan by Summer '09 <NR> Subtract(2009,2008) </NR>

Issues: Wrong Subtracted Value ('09', '<NR> Subtract(2009,2008) </NR>') |(09,1), Value 2009 not found in Snippet

Selected_headline: US Could Send 60K More Troops to Afghanistan

The method searches for any XML tag in the generated sequence. Here, the verification has found a tag. When we search for the number '2009', it is not found in the news section. We discard the generated headline and select the headline generated by the BRIO model as the generated headline.