International Journal for Housing Science and Its Applications
Publish August 10, 2025. Volume 46, Issue 4 Pages 253-260

https://doi.org/10.70517/ijhsa46420

Research on the Optimization and Training Strategy of English Language Model Based on Computational Complexity Analysis

Yun Wang[1,*]
[1] International Cooperation Department, Pingdingshan Polytechnic College, Pingdingshan, Henan, 467000, China
Corresponding authors: (e-mail: pzxygjxy@163.com).

Abstract Language modeling provides a resource carrier for students' English learning. In this paper, ZO-VRAGDA algorithm is designed to reduce the complexity of multi-task solving for English language models. By calculating the complexity of the model processing task, the intelligent body is guided to decompose the task into multiple subtasks. The efficiency and accuracy of the model in completing the task are optimized by invoking appropriate tools and reminding the error-prone points. The English language model is introduced in the language classroom to recommend personalized learning resources for students and improve teaching quality. The study shows that with different numbers of neurons and iterations, the training time of the model based on computational complexity analysis in this paper is 5.54s-7.05s, 698.53s-1213.94s and 115s and 2722s in the 2 datasets, respectively, which is better than the comparison model. In different complex task processing, the confusion degree is reduced to 41 with only 99.22s, 104.21s, 97.91s. The similarity degree is improved to 27 with only 113.53s, 60.77s, 93.31s.

Index Terms ZO-VRAGDA algorithm, complexity analysis, subtask decomposition, English language modeling

I. Introduction

Big language models are important breakthroughs in the field of natural language processing in recent years, which are capable of handling complex natural language tasks, such as text generation, machine translation, sentiment analysis and question-answer systems, by means of multilayer neural network structures and self-attention mechanisms trained by deep learning techniques [1]-[4]. These models typically have billions or even more parameters, allowing them to exhibit excellent performance and strong generalization capabilities on a wide range of tasks [5].

The evolution of large language models can be traced back to early neural network models in the 1990s, but the real breakthrough came with the proposal of the Tranformer model in 2017, which significantly improves the parallelism of the model and its ability to handle long dependencies through the self-attention mechanism [6]-[8]. Subsequently, pre-trained models such as BERT and GPT-3 were introduced, which further validated the superior performance of large language models on multiple tasks [9], [10]. Big Language Models have attracted more attention in the academic world, and many scholars have found that Big Language Models have facilitated the transformation of related disciplines towards digitalization, causing significant changes in the research objects, such as its bidirectional ability to generate natural language text and to be able to comprehend the input text making it a broad application prospect in the field of education [11]-[13]. In English language learning, the three roles played by the Big Language Model, i.e., language consultant, language companion, and language assessment expert, can significantly improve the level of interactivity and personalized learning in the smart classroom, and provide students with a more efficient and interesting learning experience [14]-[16]. However, the computational complexity of the Tranformer model grows with a certain regularity, the progress of a single operation is delayed, as well as the intrinsic storage of the model needs to be expanded, and the computational time-consumption increases [17], [18]. There is a lack of an effective breakthrough method.

This paper proposes an optimization method for analyzing the computational complexity using the ZO-VRAGDA algorithm for the problem of large task volume in English language model-assisted teaching. The ZO-VRAGDA algorithm is introduced to analyze and categorize the historical task execution process of the intelligent body and mine the task processing law. Calculate the complexity of the total task and select the appropriate tool to execute the disassembled subtasks to reduce the processing difficulty. Meanwhile, combining with data enhancement scheme homogenizes the task description way and reduces the amount of semantic retrieval. The application of English language modeling generates discourse materials for students that meet their learning needs, breaks through the limitation of high repetition of teaching materials, and guides students to actively participate in classroom learning.

II. Computational complexity-based modeling of the English language

II. A. Computational complexity analysis

In this section, we analyze the iterative complexity of the ZO-VRAGDA algorithm for solving the problem. Before proceeding to the detailed complexity analysis, we make the following assumptions about stochasticity.

Assumption 1: $g(x,y)$ satisfies all assumptions.

Assumption 2: The variance of the stochastic gradient estimator is bounded, i.e., there exists a constant $\sigma>0$ such that for any x and y, the following equation holds

$$\begin{aligned}&\mathrm{E}_{(u,\xi)}\left\|\hat{\nabla}_x G(x,y;\xi)-\nabla_x g_{\mu_1}(x,y)\right\|^2\le\sigma^2\\&\mathrm{E}_{(v,\zeta)}\left\|\hat{\nabla}_y G(x,y;\zeta)-\nabla_y g_{\mu_2}(x,y)\right\|^2\le\sigma^2\end{aligned}\tag{1}$$

Under Assumption 2, this can be simply calculated to obtain $\mathrm{E}\left\|\hat{\nabla}_x G(x,y;\mathrm{B})-\nabla_x g_{\mu_1}(x,y)\right\|^2\le\frac{\sigma^2}{r}$, $\mathrm{E}\left\|\hat{\nabla}_y G(x,y;\mathrm{B})-\nabla_y g_{\mu_2}(x,y)\right\|^2\le\frac{\sigma^2}{r}$.

Assumption 3: For each part of the problem $G(x,y;\xi)$ is $\bar{l}$-smooth, i.e., there exists a constant $\bar{l}>0$ such that $\forall x_1,x_2\in\square^{d_1}$, $y_1,y_2\in\square^{d_2}$.

$$\begin{aligned}&\left\|\nabla_x G(x_1,y_1;\xi)-\nabla_x G(x_2,y_2;\xi)\right\|\le\bar{l}\left[\left\|x_1-x_2\right\|+\left[y_1-y_2\right]\right]\\&\left\|\nabla_y G(x_1,y_1;\zeta)-\nabla_y G(x_2,y_2;\zeta)\right\|\le\bar{l}\left[\left\|x_1-x_2\right\|+\left[y_1-y_2\right]\right]\end{aligned}\tag{2}$$

Lemma 1: $g(x,y)$ is $\bar{l}$-smooth and $\Psi(\cdot)$ is $\bar{L}$-smooth, where $\bar{L}:=\bar{l}+\frac{\bar{l}^2}{2\mu}$.

Proof: using Jensen's inequality and Assumption 3, $\forall x_1,x_2\in\square^{d_1}$, $y_1,y_2\in\square^{d_2}$, with

$$\begin{aligned}&\left\|\nabla_x g(x_1,y_1)-\nabla_x g(x_2,y_2)\right\|=\left\|\mathrm{E}\left[\nabla_x G(x_1,y_1,\xi)-\nabla_x G(x_2,y_2,\xi)\right]\right\|\\&\le\mathrm{E}\left\|\nabla_x G(x_1,y_1,\xi)-\nabla_x G(x_2,y_2,\xi)\right\|\le\bar{l}\left[\left\|x_1-x_2\right\|+\left\|y_1-y_2\right\|\right]\end{aligned}\tag{3}$$

Similarly it can be shown that $\left\|\nabla_y g(x_1,y_1)-\nabla_y g(x_2,y_2)\right\|\le\bar{l}\left[\left\|x_1-x_2\right\|+\left\|y_1-y_2\right\|\right]$. By Lemma 1, it also follows that $\Psi(\cdot)$ is $\bar{L}$ - smooth, where $\bar{L}:=\bar{l}+\frac{\bar{l}^2}{2\mu}$.

Definition 1: If $\mathrm{E}\|\nabla\Psi(\bar{x})\|\le\grave{\mathrm{o}}$, then $\bar{x}$ is the $\grave{\mathrm{o}}$-stabilizing point of the problem.

Lemma 2: If Assumption 3 is satisfied, then

$$\begin{aligned}\mathrm{E}g(x_{t+1},y_{t+1})\ge&\mathrm{E}g(x_t,y_t)+\frac{\beta}{2}\mathrm{E}\left\|\nabla_y g(x_{t+1},y_t)\right\|^2-\frac{\alpha}{2}\mathrm{E}\left\|\nabla_x g(x_t,y_t)\right\|^2\\&-\frac{\beta}{2}\mathrm{E}\left\|\nabla_y g(x_{t+1},y_t)-n_t\right\|^2+\frac{\alpha}{2}\mathrm{E}\left\|\nabla_x g(x_t,y_t)-m_t\right\|^2\\&+\frac{\beta}{2}(1-\bar{l}\,\beta)\mathrm{E}\left\|n_t\right\|^2-\frac{\alpha}{2}(1+\alpha\bar{l})\mathrm{E}\left\|m_t\right\|^2\end{aligned}\tag{4}$$

Proof: first, by Lemmas 1 and (3), there are

$$\begin{aligned}&\mathrm{E}g(x_{t+1},y_{t+1})-\mathrm{E}g(x_{t+1},y_t)\\&\ge\mathrm{E}\left\langle\nabla_y g(x_{t+1},y_t),y_{t+1}-y_t\right\rangle\\&-\frac{\bar{l}}{2}\mathrm{E}\left\|y_{t+1}-y_t\right\|^2\\&=\mathrm{E}\left\langle\nabla_y g(x_{t+1},y_t),\beta n_t\right\rangle-\frac{\bar{l}}{2}\beta^2\mathrm{E}\left\|n_t\right\|^2\\&=\frac{\beta}{2}\mathrm{E}\left\|n_t\right\|^2+\frac{\beta}{2}\mathrm{E}\left\|\nabla_y g(x_{t+1},y_t)\right\|^2\\&-\frac{\beta}{2}\mathrm{E}\left\|\nabla_y g(x_{t+1},y_t)-n_t\right\|^2-\frac{\bar{l}\,\beta^2}{2}\mathrm{E}\left\|n_t\right\|^2\\&=\frac{\beta}{2}\mathrm{E}\left\|\nabla_y g(x_t,y_t)\right\|^2-\frac{\beta}{2}\mathrm{E}\left\|\nabla_y g(x_{t+1},y_t)-n_t\right\|^2\\&+\frac{\beta}{2}(1-\bar{l}\,\beta)\mathrm{E}\left\|n_t\right\|^2\end{aligned}\tag{5}$$

where the second equation follows from $\langle \bar{a}, \bar{b} \rangle = \frac{1}{2}\|\bar{a}\|^2 + \frac{1}{2}\|\bar{b}\|^2 - \frac{1}{2}\|\bar{a} - \bar{b}\|^2$, $\forall \bar{a}, \bar{b}$. Similarly, by Lemma 1, one obtains

$$\begin{aligned}
&\mathrm{E}g(x_{t+1}, y_t) - \mathrm{E}g(x_t, y_t) \\
&\geq \mathrm{E}\left\langle \nabla_x g(x_t, y_t), x_{t+1} - x_t \right\rangle - \frac{\bar{l}}{2}\mathrm{E}\|x_{t+1} - x_t\|^2 \\
&= -\mathrm{E}\left\langle \nabla_x g(x_t, y_t), \alpha m_t \right\rangle - \frac{\bar{l}}{2}\alpha^2 \mathrm{E}\|m_t\|^2 \\
&= \frac{\alpha}{2}\mathrm{E}\|\nabla_x g(x_t, y_t) - m_t\|^2 - \frac{\alpha}{2}\mathrm{E}\|\nabla_x g(x_t, y_t)\|^2 \\
&\quad - \frac{\alpha}{2}\mathrm{E}\|m_t\|^2 - \frac{\bar{l}}{2}\alpha^2 \mathrm{E}\|m_t\|^2 \\
&= \frac{\alpha}{2}\mathrm{E}\|\nabla_x g(x_t, y_t) - m_t\|^2 \\
&\quad - \frac{\alpha}{2}\mathrm{E}\|\nabla_x g(x_t, y_t)\|^2 - \frac{\alpha}{2}\left(1 + \alpha\bar{l}\right)\mathrm{E}\|m_t\|^2
\end{aligned} \tag{6}$$

Add (5) and (6) together and cite the reasoning.

II. B.Utilization process

The historical case-driven task planning approach is able to summarize three types of task planning experiences by analyzing the execution history: sub-task decomposition, task execution scheme, and error-prone point reminders, respectively. Among them, the first two originate from the successful cases in the execution history of the intelligentsia, and the last one originates from the summarization of the error-prone cases. Meanwhile, this study also designed the English language model data enhancement scheme to mitigate the effect of different task description styles on semantic retrieval.

1) Subtask decomposition experience for guiding the intelligentsia to split the task. For simple tasks, one or more steps can be accomplished, but the overall execution is linear and sequential; for complex tasks, the intelligent body needs to perform task decomposition first, and then solve the subtasks sequentially, and may even need to decompose the subtasks again, and finally solve them in a tree-like manner.

2) Task execution scheme experience, used to guide the intelligent body how to efficiently complete the current task. For example, to solve task A, tool 1, tool 2, and tool 3 can be invoked sequentially.

3) Error-prone point reminder experience, which is used to guide the intelligent body to make as few mistakes as possible. For example some tools are invoked with parameters that require extra attention, or some execution sequences have specific pre-steps.

Figure 1 shows how the present approach optimizes the original task planning for the intelligentsia. Task 1 and Task 2 in the example are similar tasks, and HAOK's role in the whole task planning is as follows: 1) It is suggested that the intelligent body should not complete subtask 1 directly, but split it into subtasks 1.2, 1.2. Because when the intelligent body executes the task for the first time, it finds that subtask 1 is too large to complete directly. 2) It is suggested that the intelligent body, when completing subtask 2, invoke tools 1 and 3 in turn, and Skipping the wrong, invalid attempts to tool 2.

II. C.Teaching methods

Large-scale pre-training language models can contribute to the development of a student-centered English discourse classroom. In a student-centered discourse classroom, students no longer just listen to the teacher's explanations and sit at their desks copying and reciting grammar notes, but really participate in the English discourse classroom. Before that, how to make every student really participate in the classroom has been a difficult problem, and teachers could not take care of every student in the process of teaching. After using the large-scale pre-training language model to generate pragmatic materials, students have the right to choose the pragmatic materials they need to learn, and teachers can judge whether students have mastered a certain pragmatic usage with the aid of the large-scale pre-training language model, and provide students with suitable pragmatic learning materials.

For example, if students' dialogues or compositions are put into the model for processing, the model will find out the students' linguistic errors and generate personalized materials that are more suitable for the students. The teacher plays the role of a facilitator and is no longer in charge of the whole classroom, nor does he or she use the same language teaching method for different students. For the students, the materials generated by the large-scale pre-training language model are carefully selected from a vast amount of information and can be personalized and assimilated by the students.

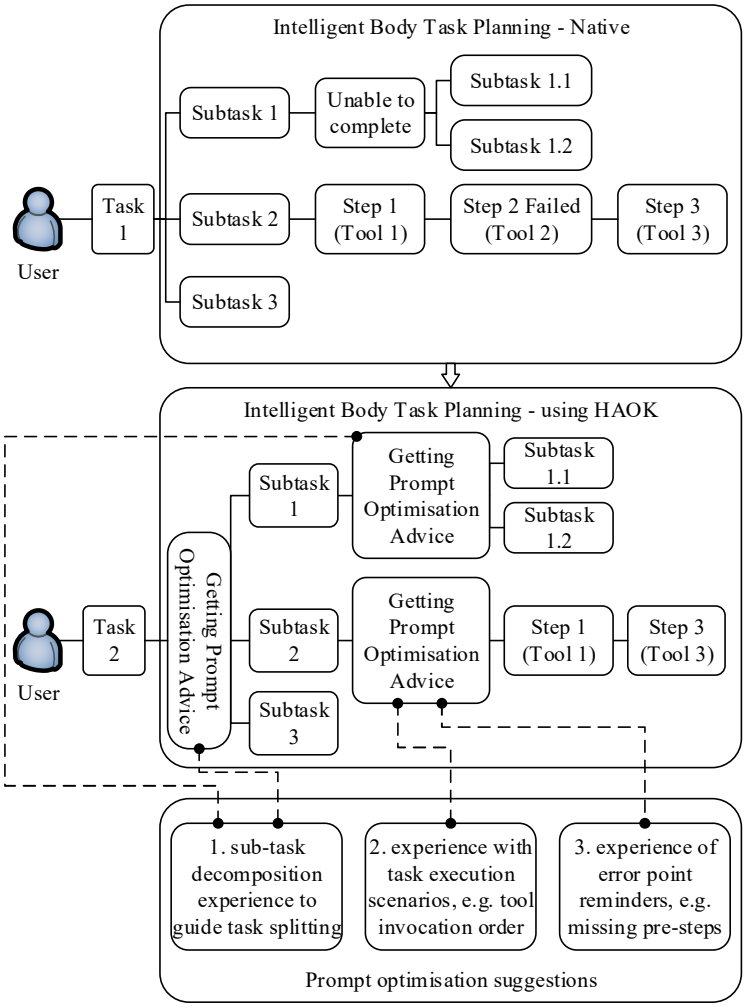


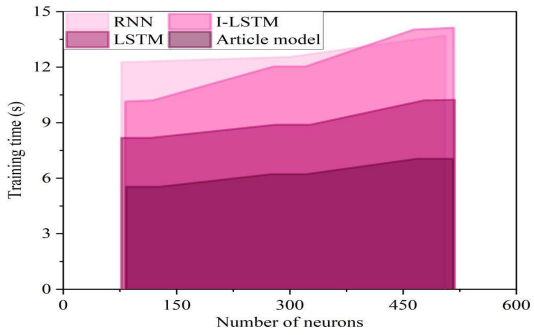
Figure 1: The optimization process of agent task planning

III. Analysis of the optimization effect of English language model based on computational complexity analysis

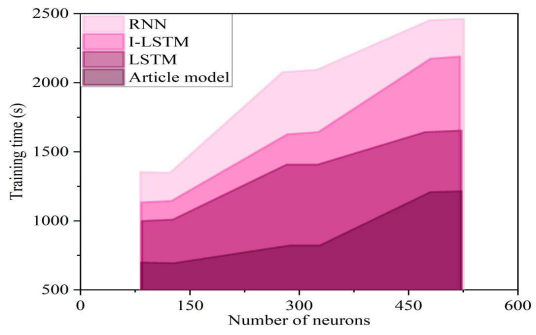
III. A. Model training time overhead comparison

III. A. 1) Comparison of model training time overhead for different number of neurons

In order to verify the effectiveness of the model in task processing in English language teaching after the introduction of the ZO-VRAGDA algorithm to calculate the task complexity, the study conducts semantic analysis experiments with the COIN dataset and Charades dataset, and compares the training time overhead of different language models with different numbers of neurons, to judge the efficiency of the model's task processing after the subtask decomposition. The study selected 350 video messages from the C0IN dataset and 5500 video messages from the Charades dataset for model validation. Among them, 263 videos from COIN dataset and 4125 videos from Charades dataset were used in the training set. The test set COIN dataset 87 articles and Charades dataset 1375 articles. Meanwhile, the study takes Spark as the framework of the model validation system, uses the distributed computing system based on remote direct data access protocol for validation, and sets the model learning efficiency to 0.05. Firstly, the model time overhead validation is carried out in terms of the number of neurons of the model and the number of selected generations, and LSTM, RNN, I-LSTM and the model of this paper are introduced for the comparison of the results. Figure 2 shows the training overhead of different models with varying number of neurons. The training time of this paper's model, which introduces the ZO-VRAGDA algorithm to reduce the computational complexity of the task, is always the shortest as the number of neurons varies between 0 and 600. In the training set COIN dataset, the training time of this paper's model varies between 5.54s-7.05s; in the training set Charades dataset, the training time of this paper's model varies between 698.53s-1213.94s, which are all less than the training time of the other three comparison models. It indicates that the task processing speed of this paper's model is faster under different numbers of neurons.



(a) Training time cost of the COIN dataset

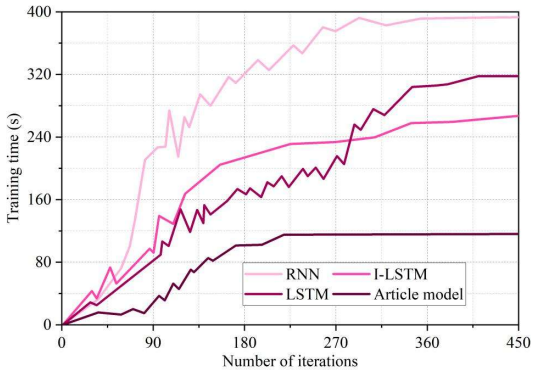


(b) Training time cost of the Charades dataset

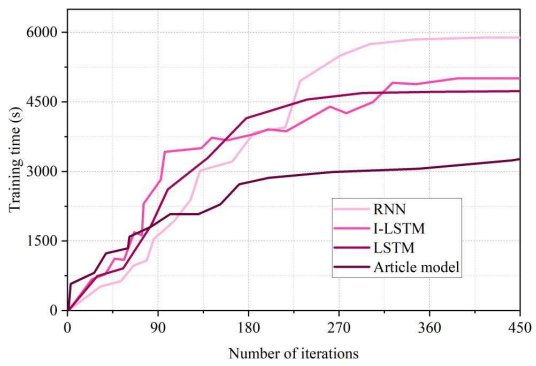
Figure 2: Training time cost under different numbers of neurons

III. A. 2) Comparison of model training time overhead for different number of iterations

Fig. 3 shows the training time overhead of different models with different number of iterations. The training time overhead of this paper's model is the least during 450 training iterations. The model in this paper is basically stabilized at 115s training time after 219 iterations on COIN training dataset. On the Charades training dataset, after 171 iterations, it is basically stabilized at more than 2722s, with smoother fluctuations. Compared with the other three comparative models, the model in this paper reaches a smooth state after fewer iterations, and the training time is much less than the other models. This again verifies that the improved model in this paper has a faster task processing speed.



(a) Training time cost of the COIN dataset



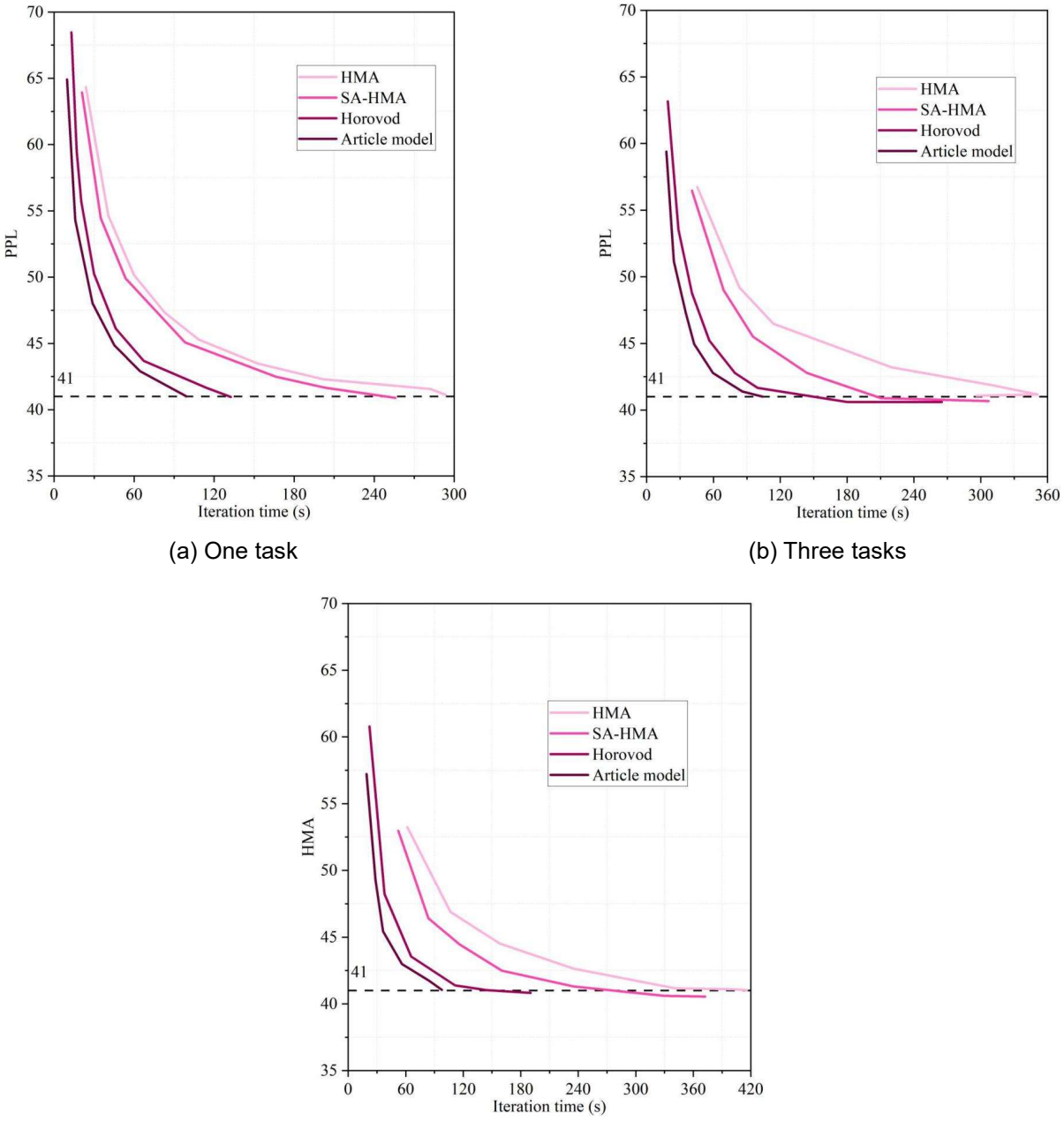
(b) Training time cost of the Charades dataset

Figure 3: Training time cost under different iterations

III. B. Model Task Processing Performance Comparison

III. B. 1) Confusion Level (PPL) Comparison

The language model based on computational complexity analysis is compared with the language model based on simultaneous optimization to determine the performance advantage of this paper's language model in the processing of tasks such as student proficiency analysis and discourse material recommendation. Figure 4 shows the training convergence curves of HMA, SA-HMA, Horovod model and this paper's model under different task sizes. Whether processing 1 complex task, 3 complex tasks or 5 complex tasks simultaneously, the model based on computational complexity analysis in this paper takes the shortest iteration time when reducing the perplexity to the level of 41. It takes only 99.22s for 1 complex task, 104.21s for 3 complex tasks, and 97.91s for 5 complex tasks.Compared with other models based on simultaneous optimization algorithms, this paper's model has a faster speed in processing complex tasks. The reason is that the model in this paper applies the ZO-VRAGDA algorithm to quickly calculate the actual complexity of the complex tasks and decompose them into multiple sub-tasks with low computational effort, which reduces the task processing perplexity of the model.



(a) One task

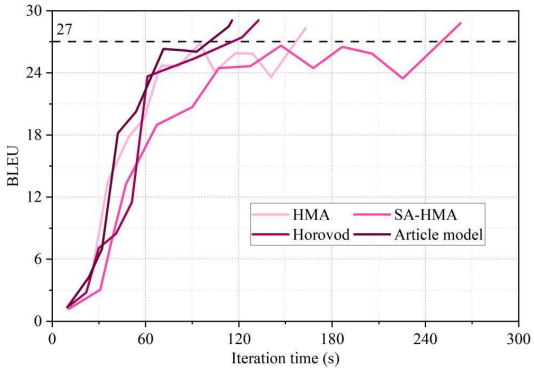
(b) Three tasks

(c) Five tasks

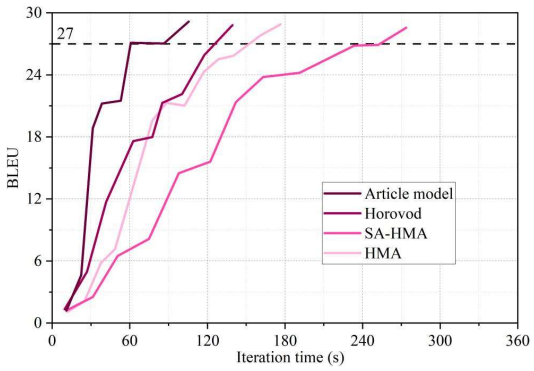
Figure 4: Comparison of training convergence curves

III. B. 2) Similarity (BLEU) Comparison

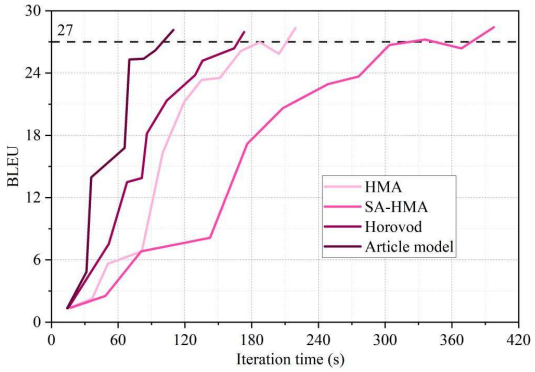
The comparison of the BLEU performance of the 4 models is continued. Figure 5 shows the comparison results. Processing 1, 3, and 5 complex tasks at the same time, the iteration times of this paper's model to reach a BLEU value of 27 are 113.53s, 60.77s, and 93.31s, respectively, which are all faster than the comparison models. The reason is similar to the one analyzed in the previous paper, because the complex task is decomposed into multiple subtasks, and accordingly the similarity between recommended materials can be detected more quickly, and the high-quality material recommendation task can be accomplished.



(a) One task



(b) Three tasks



(c) Five tasks

Figure 5: BLEU comparison result

IV. Conclusion

In this paper, the ZO-VRAGDA algorithm is used to calculate the complexity of the English language teaching task and guide the intelligences to decompose the task to improve the task processing efficiency and effectiveness of the model. In 0-600 number of neurons, the highest training time of this paper's model is only 7.05s and 1213.94s on 2 datasets, which is better than the comparison model. Among 0-450 iterations, this paper's model reaches a stable training time of 115s for 219 iterations and a smooth training time of 2722s after 171 iterations on the 2 datasets. When dealing with 1, 3 and 5 complex tasks and reducing the perplexity to 41, the iteration time is only 99.22s, 104.21s and 97.91s.When the BLEU value is increased to 27, the iteration time is 113.53s, 60.77s and 93.31s, respectively.In the future, the English language model can be applied to actual English teaching, and the model parameters can be further adjusted according to the practice results to improve its English teaching aid effect.

References

[1] Zhang, H., Song, H., Li, S., Zhou, M., & Song, D. (2023). A survey of controllable text generation using transformer-based pre-trained language models. ACM Computing Surveys, 56(3), 1-37.

[2] Son, J., & Kim, B. (2023). Translation performance from the user's perspective of large language models and neural machine translation systems. Information, 14(10), 574.

[3] Assiri, A., Gumaei, A., Mehmood, F., Abbas, T., & Ullah, S. (2024). DeBERTa-GRU: Sentiment Analysis for Large Language Model. Computers, Materials & Continua, 79(3).

[4] Ting, W. A. N. G., Na, W. A. N. G., Yunpeng, C. U. I., & Juan, L. I. U. (2023). Agricultural technology knowledge intelligent question-answering system based on large language model. Smart agriculture, 5(4), 105.

[5] Anil, C., Wu, Y., Andreassen, A., Lewkowycz, A., Misra, V., Ramasesh, V., ... & Neyshabur, B. (2022). Exploring length generalization in large language models. Advances in Neural Information Processing Systems, 35, 38546-38556.

[6] Wu, Y. C., Yin, F., & Liu, C. L. (2017). Improving handwritten Chinese text recognition using neural network language models and convolutional neural network shape models. Pattern Recognition, 65, 251-264.

[7] Ma, X., Zhang, P., Zhang, S., Duan, N., Hou, Y., Zhou, M., & Song, D. (2019). A tensorized transformer for language modeling. Advances in neural information processing systems, 32.

[8] Verianto, E., & Shimbun, A. F. (2024). TRANSFORMER WITH LAGGED FEATURES FOR HANDLING LONG-TERM DATA DEPENDENCY IN TIME SERIES FORECASTING. JIKO (Jurnal Informatika dan Komputer), 7(3), 232-241.
[9] Lee, J. S., & Hsiang, J. (2020). Patent classification by fine-tuning BERT language model. World Patent Information, 61, 101965.
[10] Kalyan, K. S. (2024). A survey of GPT-3 family large language models including ChatGPT and GPT-4. Natural Language Processing Journal, 6, 100048.
[11] Idris, M. D., Feng, X., & Dyo, V. (2024). Revolutionising Higher Education: Unleashing the Potential of Large Language Models for Strategic Transformation. IEEE Access.
[12] Yigci, D., Eryilmaz, M., Yetisen, A. K., Tasoglu, S., & Ozcan, A. (2025). Large language model‐based chatbots in higher education. Advanced Intelligent Systems, 7(3), 2400429.
[13] Shi, H., Xu, Z., Wang, H., Qin, W., Wang, W., Wang, Y., ... & Wang, H. (2024). Continual learning of large language models: A comprehensive survey. ACM Computing Surveys.
[14] Rong, L., Zhang, Y., Tiwari, P., & Yu, M. (2025). BegoniaGPT: Cultivating the large language model to be an exceptional K-12 English teacher. Neural Networks, 107488.
[15] Sharma, S., Mittal, P., Kumar, M., & Bhardwaj, V. (2025). The role of large language models in personalized learning: a systematic review of educational impact. Discover Sustainability, 6(1), 1-24.
[16] Yavuz, F., Çelik, Ö., & Yavaş Çelik, G. (2025). Utilizing large language models for EFL essay grading: An examination of reliability and validity in rubric‐based assessments. British Journal of Educational Technology, 56(1), 150-166.
[17] Alizadeh, K., Mirzadeh, S. I., Belenko, D., Khatamifard, S., Cho, M., Del Mundo, C. C., ... & Farajtabar, M. (2024, August). Llm in a flash: Efficient large language model inference with limited memory. In Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 12562-12584).
[18] Lu, J., Yao, J., Zhang, J., Zhu, X., Xu, H., Gao, W., ... & Zhang, L. (2021). Soft: Softmax-free transformer with linear complexity. Advances in Neural Information Processing Systems, 34, 21297-21309.