AlarmGPT: An Intelligent Operation Assistant for Optical Network Alarm Analysis Using ChatGPT

Yidi Wang (1), Jin Li(1), Yue Pang(1), Yuchen Song(1), Lifang Zhang(2), Min Zhang(1), and Danshi Wang(1)*

(1) State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications, Beijing 100876, China,email: *danshi_wang@bupt.edu.cn
(2) The Intelligent Network Innovation Center of Chinaunicom, China Unicom, Beijing,100033, China.

Abstract *The AlarmGPT, utilizing professional alarm knowledge and Embeddings API on ChatGPT, has been proposed for processing and analysing optical network alarms. AlarmGPT can complete professional alarm Q&A and achieve high accuracy in alarm compression and fault analysis.*

Introduction

In optical networks, alarm analysis is a significant component in the day-to-day network maintenance, and it is critical for efficient and accurate fault localization and failure recovery. When an alarm occurs, it is necessary to take immediate action to resolve the issue and prevent further problems through conducting efficient analysis on a large amount of alarm data. In practical, the existing optical network alarm analysis systems were relying heavily on manual process and artificial expertise at the great cost of labour and time. However, millions of alarms in the optical layer may appear in optical transport network (OTN) every month, which brings great challenges to network operation, administration, and maintenance.

To address this issue, techniques of machine learning has been introduced to alarm analysis to implement alarm compression, correlation alarm identification, and alarm root cause localization [1-3]. In addition, the massive alarm data are typically represented in the mode of text, where alarm data processing can be considered as a process of understanding the context from the perspective of natural language processing (NLP) to perform alarm recognition and diagnosis [4-5]. However, these ML-based schemes mainly focused on one or two simple tasks with limited intelligent ability. Recently, with the advancement of large language model (LLM), GPT (Generative Pre-trained Transformer, which is a powerful tool for NLP released by OpenAI) exhibits strong capabilities in most NLP problems for the purpose of artificial general intelligence (AGI), but its performance deteriorates when it comes to specific tasks in specific fields or complex issues in some professional areas.

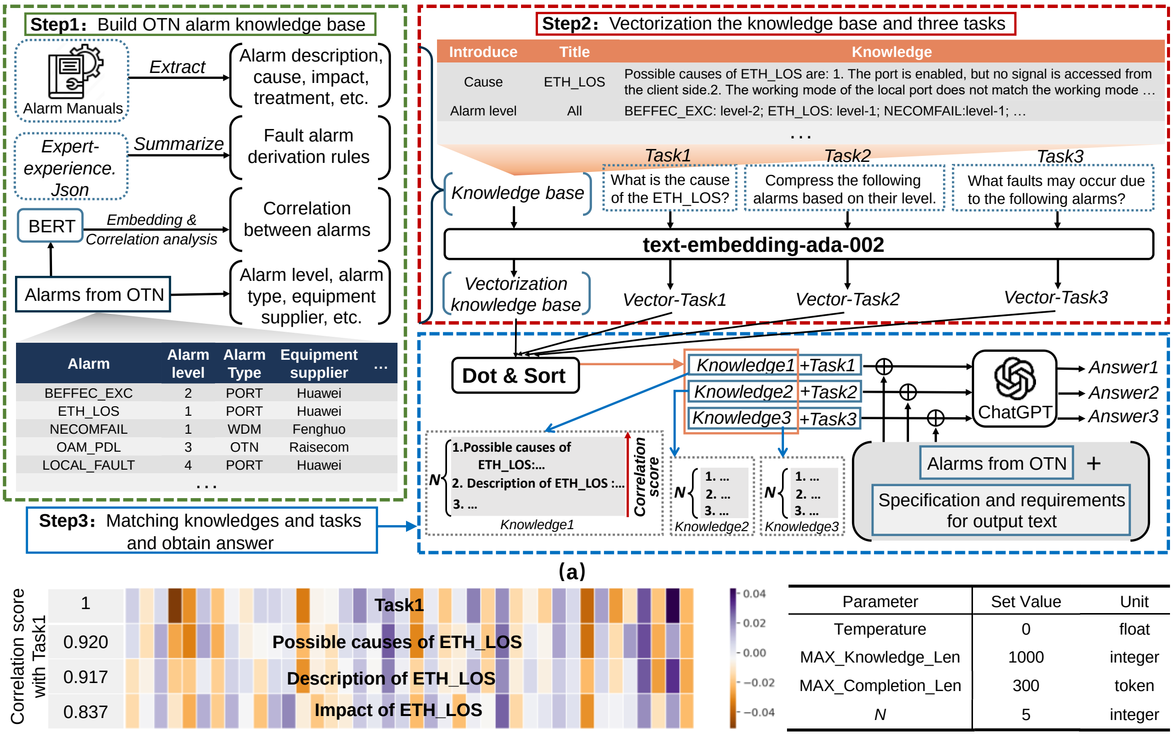
In optical networks, alarms come from different levels and devices, and the types and quantities of alarms are complex, where an efficient and intelligent analysis method requires the strong capabilities of rule understanding and logical reasoning. The emergence of ChatGPT makes this possible, and lately, researchers have developed several GPT-based chatbots or virtual assistants to handle the specific tasks for their respective fields, such as NLX-GPT, GPT-CRITIC and DeID-GPT [6-8]. Inspired by the charm of ChatGPT, we think it is envisioned with optimism that there is also great potential for ChatGPT to be utilized in our areas for multiple prospective applications.

In this article, we propose AlarmGPT based on a professional OTN alarm knowledge base to perform prompt-tuning on ChatGPT, which is used to automatically search for corresponding alarm text knowledge for alarm Q&A, alarm compression, and fault analysis. Experimental results show that not only can the AlarmGPT answer basic knowledges about OTN alarms, but also realize real-time analysis, alarm compression and fault prediction. Fast alarm compression can be performed according to any rule, with a compression ratio of 30%~85%, and can be completed within an average of 4 seconds, and the average fault prediction accuracy can reach 88%.

Operating Principle

As a pre-trained LLM, ChatGPT can understand context and generate human-like response. Due to a lack of professional knowledge, the generated responses may not be accurate, which limited its application in specific fields. To perform effective alarm analysis for optical networks, alarm-related knowledge is embedded into ChatGPT to do prompt-tuning via application programming interface (API) to analyse OTN alarm information. The workflow is shown in Fig. 1(a).

Step1, build a professional knowledge base of OTN alarms, including: 1) basic alarm definition, possible causes, impacts, and pre-processing methods extracted from alarm manual; 2) fault alarm derivation rules extracted from expert experience; 3) correlation information among alarms processed by BERT (Bidirectional Encoder Representations from Transformers [9]);



Parameter	Set Value	Unit
Temperature	0	float
MAX_Knowledge_Len	1000	integer
MAX_Completion_Len	300	token
N	5	integer

(c)

Fig. 1: (a) The process of using AlarmGPT to complete professional task; (b) The heat map(40 dimensions) of the first three knowledge with the highest score associated with task1 after vectorization; (c) Set value and units of some parameters.

4) the information extracted from real OTN alarm data, such as alarm level, alarm type, and equipment supplier, etc.

Step2, the extracted knowledge will be stored in the *knowledge base* file in row units, which will be sent into the word embedding model, each line of knowledge will be output as a 1536-dimensional vector and stored in the *vectorization knowledge base*. Three tasks are also input into the word embedding model in the form of text to get the representation after vectorization (*Vector-Task1, Vector-Task2, Vector-Task3*).

Step3 will take *Task1* as an example. First, perform similarity calculation, dot multiply *Vector-Task1* with all the vectorization knowledge to obtain the correlation score with each knowledge, and then sort based on the correlation score to obtain the top *N* knowledges with the highest score as *knowledge1*, which will serve as auxiliary knowledge to enable ChatGPT to better understand alarm information and issues. The vectorization representation of *Task1* and the three knowledges with the highest correlation scores are displayed in the form of heat maps, as shown in Fig. 1(b). Then, use the API of OpenAI to send *knowledge1*, *Task1*, alarms from OTN, specification and requirements for output text in a certain order and format to do prompt-tuning on ChatGPT. ChatGPT understands and analyses the task, and then give the *Anser1*. Do the same for *Task2* and *Task3*, get *Anser2* and *Anser3*. Some parameter settings during the prompt-tuning process of ChatGPT are listed, as shown in Fig. 1(c). To ensure that ChatGPT can output the most reliable results, we set the Temperature (the parameter between 0 and 1, lower values will make answer more focused and deterministic) to 0 and inform ChatGPT not to answer anything it cannot determine. Use MAX_Knowledge_Len to limit the maximum length of knowledges provided, overly miscellaneous knowledge can affect the judgment of ChatGPT. MAX_Completion_Len and *N* represent the maximum length of generated answers and the number of selected knowledges.

Experimental Results and Analysis

To verify the effectiveness of AlarmGPT, intercepted 1000 out of 29, 541 alarms generated within 2 days from a real OTN network for experimental verification, involving 33 types of alarm information from different devices and levels. We selected 40 adjacent alarms from 1000(with a time span of approximately 10 minutes) each time for testing. The dialogue model uses official model '*gpt-3.5-turbo*' of ChatGPT, and the word embedding model uses '*text-embedding-002*'. A professional alarm knowledge base is established for these alarms according to the method described earlier. Then test according to different tasks. Each test can be completed within 2-5 seconds. In this work, three tasks are selected to verify the professionalism of AlarmGPT-based assistant.

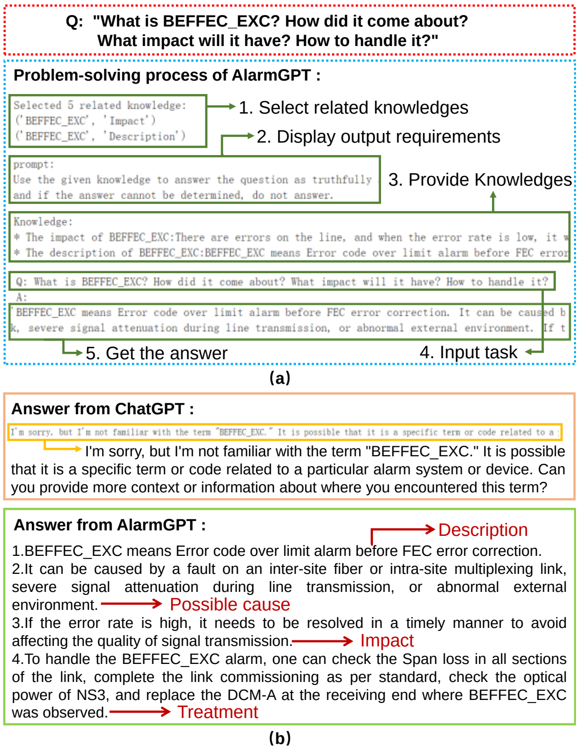


Fig. 2: (a) Display of test results; (b) Content of the answer.

First, ask questions about any alarm and test the mastery of basic alarm knowledge of AlarmGPT. The question and process for solving the given question in AlarmGPT is shown in Fig. 2(a). The response of AlarmGPT is shown in Fig. 2(b), which also shows the response of ChatGPT to the same question without prompt-tuning. ChatGPT indicates that it cannot answer this question. In contrast, AlarmGPT provided comprehensive and accurate answers based on the professional knowledge provided.

Second, test the compression capability. The traditional compression rules are based on the priority of alarm processing, but there is inevitably a need for compression based on other rules during the alarm processing process. The intelligent alarm analysis assistant should have the ability to quickly compress alarms based on different compression rules. Compression test is based on the following four aspects: 1) alarm level 2) alarm occurrence frequency 3) equipment supplier and 4) processing priority. The response of AlarmGPT for alarm compression based on different rules is shown in the Fig. 3(a). The compressed alarms are: high-level alarms, alarms with high frequency of occurrence, alarms from Huawei device, and alarms with higher processing priority. As shown in the Fig. 3(b), AlarmGPT can effectively and quickly compress alarms based on different alarm rules and maintain high compression accuracy.

Finally, perform fault prediction and analysis. Alarm information that caused specific faults from expert experience is extracted and added further analysis results to the alarm knowledge base. Specifically, we specify which alarms will trigger faults, obtained their correlation through BERT, and combined information such as the level, potential impact to determine the specific probability of each alarm triggering a fault. Based on the given alarm, AlarmGPT can analyse the possible types of faults and their probability of occurrence, as shown in Fig. 3(c), the visualization results are shown in Fig. 3(d). The success rate of fault type prediction reached 88%, and the error of prediction probability was within 10%, indicating that AlarmGPT has reliable performance in fault prediction.

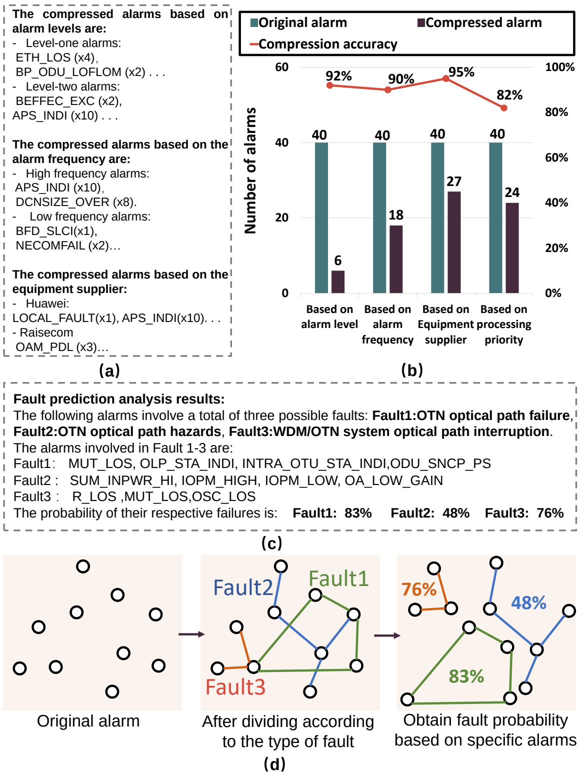


Fig. 3: (a) Alarm compression results; (b) Statistical results and compression accuracy of AlarmGPT; (c) Fault prediction results; (d) Correlation graph obtained from prediction

Conclusions

In this work, we proposed AlarmGPT: based on prompt-tuning to achieve intelligent processing and analysis of the entire alarm process, greatly improving processing efficiency. The experimental results indicate that the method we proposed can effectively apply ChatGPT to optical network alarm processing, providing assistance for alarm analysis.

Acknowledgements

National Natural Science Foundation of China under Grants (61975020 and 62171053). We would like to acknowledge Chinaunicom for sharing the valuable alarm data.

References

[1] D. Wang, L. Lou, M. Zhang, A. C. Boucouvalas, C. Zhang and X. Huang, "Dealing With Alarms in Optical Networks Using an Intelligent System," in IEEE Access, vol. 7, pp. 97760-97770, 2019, DOI: 10.1109/ACCESS.2019.2929872

[2] H. Zhuang, Y. Zhao, X. Yu, Y. Li, Y. Wang and J. Zhang, "Machine-Learning-based Alarm Prediction with GANs-based Self-Optimizing Data Augmentation in Large-Scale Optical Transport Networks," 2020 International Conference on Computing, Networking and Communications (ICNC), Big Island, HI, USA, 2020, pp. 294-298, DOI: 10.1109/ICNC47757.2020.9049750

[3] M. Deng, P. Li, Y. Liu, R. Zhu and Y. Zhang, "Mining Alarm Association Rules in Optical Transmission Networks Using a Statistical Approach," 2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys), Hainan, China, 2022, pp. 987-992, DOI: 10.1109/HPCC-DSS-SmartCity-DependSys57074.2022.00157

[4] C. Wang, N. Yoshikane, D. Elson, and T. Tsuritani, "Automation of Fast Configuration Error Diagnosis in Optical Transport Networks – Natural Language Processing is All You Need," in *Optical Fiber Communication Conference (OFC) 2023*, Technical Digest Series (Optica Publishing Group, 2023), paper M3G.6.

[5] J. Jia et al., "Transformer-based Alarm Context-Vectorization Representation for Reliable Alarm Root Cause Identification in Optical Networks," 2021 European Conference on Optical Communication (ECOC), Bordeaux, France, 2021, pp. 1-4, DOI: 10.1109/ECOC52684.2021.9606141

[6] Fawaz Sammani, Tanmoy Mukherjee, Nikos Deligiannis; "NLX-GPT: A Model for Natural Language Explanations in Vision and Vision-Language Tasks," Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 8322-8332, DOI: 10.1109/cvpr52688.2022.00814

[7] Jang, Y., Lee, J., & Kim, K. E. (2022). GPT-critic: Offline reinforcement learning for end-to-end task-oriented dialogue systems. In International Conference on Learning Representations.

[8] Liu, Z., Yu, X., Zhang, L., Wu, Z., Cao, C., Dai, H., ... & Li, X. (2023). Deid-gpt: Zero-shot medical text de-identification by gpt-4. arXiv preprint arXiv:2303.11032. DOI: 10.48550/arXiv.2303.11032

[9] google-research, "bert," [Online]. Available: https://github.com/google-research/bert. [Accessed 2023]