BatteryGPT: Battery Anomaly Detection Based on Multimodal Large Language Model

Danpeng Cheng
School of Electrical and Electronic Engineering
Huazhong University of Science and Technology
Wuhan, China
cdp0000052@hust.edu.cn

Shuping Wang
State Grid Anhui Electric Power Research Institute
Anhui Provincial Key Laboratory of New Type Power Systems Fire Safety and Emergency Technology
Hefei, China
wangshuping516@126.com

Changhao Li
State Grid Anhui Electric Power Research Institute
Anhui Provincial Key Laboratory of New Type Power Systems Fire Safety and Emergency Technology
Hefei, China
346550617@qq.com

Heng Xie
State Grid Anhui Electric Power Research Institute
State Grid Anhui Electric Power Co., Ltd.
Hefei, China
xieh0012@ah.sgcc.com.cn

Wuxin Sha
School of Electrical and Electronic Engineering
Huazhong University of Science and Technology
Wuhan, China
d201980975@hust.edu.cn

Yuan-Cheng Cao*
School of Electrical and Electronic Engineering
Huazhong University of Science and Technology
Wuhan, China
yccao@hust.edu.cn
*Corresponding author

2024 5th International Conference on Power Engineering (ICPE) | 979-8-3315-3114-0/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICPE64565.2024.10929157

***Abstract*—In the context of pursuing carbon neutrality goals, the importance of battery energy storage systems is becoming increasingly prominent. Battery defect detection has become critical to ensuring the safety and enhancing the efficiency of these storage systems. Current detection methods face challenges such as limited sample availability, insufficient recognition accuracy, low sensitivity to defect details, and a lack of domain-specific knowledge. To address these issues, this paper proposes a lithium-ion battery defect detection technology based on the multimodal large language model, BatteryGPT. BatteryGPT integrates a pre-trained image encoder, a multimodal text encoder, and a large language model, utilizing simulated anomaly data to locate and describe anomalies in battery images while supporting multi-turn conversations. By aligning the features of the visual and text encoders, the system derives localization results and fine-tunes the input encoding of the large language model. This approach enables the detection of unseen anomalies in batteries with only a small number of normal samples. BatteryGPT offers a more efficient solution for battery anomaly detection, with the potential to further improve the research and production efficiency of electrochemical energy storage systems.**

Keywords—Lithium-ion batteries, battery energy storage, deep learning, Industrial Anomaly Detection, Large Vision-Language Models

I. INTRODUCTION

This template, Under the global strategy of achieving carbon neutrality, batteries are playing an increasingly critical role in various applications, such as integrating renewable energy into the grid, powering portable electronic devices, and serving as the energy source for electric vehicles. As the core component of these energy storage systems, battery performance directly impacts the reliability, safety, and environmental friendliness of products. Effective defect detection can identify and eliminate potentially hazardous batteries in advance, preventing catastrophic incidents like fires and explosions. Additionally, it is a key factor in improving battery production efficiency, reducing costs, and promoting the sustainable development of green energy. Therefore, in-depth research and optimization of battery defect detection technology are of profound significance for fostering the healthy development of the battery industry and building a safe and efficient energy system.

The core objective of battery anomaly detection is to identify and locate anomalies within battery images. Current industrial anomaly detection techniques can generally be classified into two main types: those based on reconstruction and those centered around feature embedding. Reconstruction-based approaches primarily aim to reconstruct abnormal samples into corresponding normal ones, with anomalies being detected by calculating the reconstruction error. These reconstruction methods can utilize various network architectures [1-3]. SCADN is a classic reconstruction method that detects anomalies by masking certain parts of the image and using a Generative Adversarial Network (GAN) to reconstruct it. However, such methods often struggle to adequately capture the details in normal regions during reconstruction [1]. They mainly focus on local features such as pixels, brightness, and structure, while lacking a deeper understanding of higher-level semantic information.

Feature embedding-based methods detect anomalies by embedding defect patches into normal samples to construct a dataset and then measure the distance between test samples and

This work was supported by the Science and Technology Project of State Grid Corporation of China (Development of high-performance special extinguishing agent for lithium-ion battery fire, No. 5500-202220118A-1-1-ZN).

the nearest normal samples in the dataset. The CutPaste method, based on convolutional neural networks (CNN), generates pseudo-anomalous samples by randomly cutting and pasting image patches on normal data [4]. This approach leverages the spatial distribution irregularities of image patches to simulate real industrial defect patterns, enabling effective recognition and learning of anomalies. However, these methods typically require a large number of defect samples for each category to learn their distributions. As a result, their accuracy in identifying new categories is limited, making them less suitable for the dynamic and complex environment of battery production and research.

In contrast, multimodal large language model (MLLM)-based approaches can learn new object categories from multiple data sources and achieve defect localization and classification with only a small number of normal samples, thus enabling few-shot learning. Common few-shot learning methods in the field of defect detection include CLIPs [5, 6]. However, these methods typically only provide anomaly scores for test samples during inference, making it difficult to establish and unify the threshold that distinguishes normal samples from anomalous ones. Inappropriate thresholds may reduce detection accuracy in subsequent applications. Large language models (LLMs) like GPT-3.5 [7] and LLaMA [8] have demonstrated exceptional performance in numerous natural language processing (NLP) tasks. Recently, new methods such as Vicuna [11], and PandaGPT [12] have expanded the application of LLMs in visual processing by aligning visual and textual features. MLLMs have the potential to directly assess whether a sample contains anomalies and pinpoint their location, showcasing strong practical applicability. However, these models face significant challenges when applied directly to battery anomaly detection. Firstly, data scarcity poses a major issue. Existing battery anomaly detection datasets are limited, making direct fine-tuning prone to overfitting and catastrophic forgetting. Secondly, these models lack domain-specific knowledge, which hinders their ability to accurately understand local details of defects, reducing their effectiveness in battery defect detection.

This paper proposes a novel multimodal large language model, BatteryGPT, which integrates data from both visual and language modalities to detect defects in lithium-ion batteries and support multi-turn dialogues. By combining the powerful language understanding and generation capabilities of large language models (LLMs) with the advanced visual recognition technology of visual transformers, BatteryGPT enables intelligent identification and diagnosis of various complex defects in lithium-ion batteries. The model utilizes a multimodal information fusion mechanism, offering excellent generalization capabilities by simultaneously analyzing both image and textual features to provide accurate and easily interpretable defect detection and descriptions. Additionally, we developed a high-quality battery dataset, comprising 66 optical images of defect-free pouch cells and 40 internal ultrasound images of pouch cells without bubbles. This study enhances the automation and accuracy of lithium-ion battery defect detection, serves as an example of MLLM applications in advanced energy storage battery industry research, and offers substantial support for the research and development of the lithium-ion battery industry.

II. BATTERYGPT MODEL

A. Network Architecture

BatteryGPT integrates multimodal information from both language and images to detect internal and external defects in lithium-ion batteries as shown in Fig. 1. The model consists of two image encoders, which are responsible for encoding input information and detection results, respectively. It also includes a text encoder for processing textual descriptions of defects to assist in the detection process. Additionally, the model incorporates a large language model (Vicuna) for interacting with users. Through multiple linear projection layers, the model efficiently integrates different input sources. BatteryGPT supports multi-turn dialogues with users, significantly enhancing the automation and accuracy of lithium-ion battery defect detection. This also facilitates effective sharing and dissemination of detection knowledge, providing strong support for the sustained and healthy development of the lithium-ion battery industry.

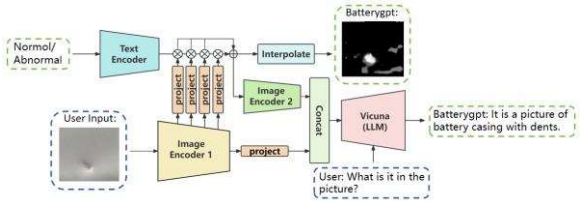

Fig. 1. Structure diagram of the BatteryGPT model.

1) Encoders

The model integrates two image encoders and one text encoder, all based on the Transformer architecture. Both image encoders utilize the Vision Transformer (ViT) architecture [9] and employ a patch-splitting module to divide the input images into multiple non-overlapping patches of 16×16 pixels, allowing for effective feature extraction. The text encoder is based on the CLIP text encoder, which extracts text features that are highly consistent with the image features [6]. The first image encoder is responsible for extracting features from the images to be inspected, while the text encoder encodes textual descriptions related to "defect" or "without defect." By calculating the similarity between text features and image features, the model can identify whether defects exist within the images. To facilitate interaction with users, the labeled results from the second image encoder are input into the large language model (Vicuna), along with the user's text encoding, enabling intelligent interpretation and feedback on the defect detection results.

2) Decoder

By employing a residual connection approach, the features obtained from the fusion of image and text are added together, preserving the original information while enhancing feature expressiveness. After feature fusion, to better accommodate the spatial information processing requirements, the feature dimension d^\prime is transformed into a two-dimensional spatial dimension H×W to restore the spatial structure of the image. Finally, interpolation techniques are used to process the

fused features, allowing for precise marking of defect locations in the output images. This method not only effectively retains the fusion characteristics of multimodal information but also improves the accuracy of defect localization, providing reliable visual interpretations for defect detection in lithium-ion batteries.

3) Fusion Section

To enhance the accuracy of image understanding, feature maps extracted by the encoder at four different stages are fused during the image encoding process. These staged feature maps capture information at various levels and scales within the image, providing richer visual cues for defect detection. To further optimize the integration of text encoding and image encoding, the feature maps from these four stages undergo linear projection transformations to ensure that the transformed feature maps have the same dimensions as the text encoding. This transformation not only helps unify the feature space but also facilitates more efficient calculations of similarity between images and text, thereby improving the accuracy and reliability of defect detection. The specific transformation formula is as follows:

$$F'_{patch} = F_{patch} \times W + b \tag{1}$$

where $F_{patch} \in R^{N \times d}$ represents the image features, $W \in R^{d \times d'}$ is the linear projection matrix, $b \in R^{d'}$ is the bias term, and $F_{patch}{}' \in R^{N \times d'}$ represents the projected feature matrix. Here, d is the dimension of the image features, and d^\prime is the dimension of the text features.

$$\text{Similarity}\left(F'_{patch}, F_{text}\right) = \cos\theta = \frac{F_{patch}{}' \cdot F_{text}}{|F_{patch}{}'||F_{text}|} \tag{2}$$

where F_{text} represents the text feature vector, and $\left|F'_{patch}\right|$ and $|F_{text}|$ denote the L1 norms of the image feature vector and the text feature vector, respectively.

At the same time, to more efficiently integrate the decoded output results, encoded input features, and user input information from the large language model (LLM), the decoded input features are passed to the second image encoder, and a linear projection is performed on the final layer output of the first image encoder to adjust the feature dimensions. Subsequently, the projected features are concatenated with the output from the second image encoder, and this concatenated result is input into the LLM along with the text encoding from the user. This fusion strategy effectively integrates information from multiple sources, enhancing the model's understanding and responsiveness to the input, thereby optimizing interaction performance and improving the interpretability of the detection results.

4) Chat Section

This study employs the large language model (Vicuna) as the core component for dialogue interaction, leveraging its exceptional semantic understanding and built-in knowledge base to provide an in-depth analysis of detection results. This assists users in intuitively grasping the defects at different locations, along with their potential shapes and sizes. Such information is crucial for evaluating the potential impact of defects on the manufacturing and performance of batteries.

To support multi-turn interactions, the model incorporates a context retention mechanism, which utilizes the inputs and outputs from historical dialogues as contextual information, alongside the current user input, to be passed to the large language model. This is-context learning with a multi-turn dialogue framework enhances the coherence and accuracy of interactions and allows the model to dynamically adapt to user feedback and inquiries, thereby providing more precise and personalized explanations and recommendations.

B. Loss Function

To train the model's segmentation capabilities, the loss function employs both cross-entropy loss and Dice loss. The cross-entropy loss is used to evaluate the accuracy of the model's classification of each pixel or region. This loss function encourages the model to accurately predict the presence or absence of defects at each pixel point by calculating the cross-entropy between the predicted class probability distribution and the true class labels. This loss function is particularly effective in multi-class classification problems and helps improve the classification accuracy of the model. The formula is as follows:

$$\mathcal{L}_{\text{C}} = -\frac{1}{N}\sum_{i=1}^{N}\sum_{c=1}^{C} y_i \log(p_i) \tag{3}$$

where N is the total number of tokens, y_i represents the true labels, and p_i is the probability predicted by the model that token i contains a defect.

The Dice loss is used to measure the overlap between the predicted results and the true labels, especially in cases of imbalanced class distribution. It encourages the model to maximize the coverage of the true target area by calculating the Dice coefficient between the predicted area and the ground truth area. This loss function exhibits good robustness for detecting and segmenting small targets, effectively addressing challenges posed by the small size or uneven distribution of defect areas. The formula is as follows:

$$\mathcal{L}_{\text{Dc}} = 1 - \frac{2\sum_{i=1}^{N}(p_i \cdot y_i) + \epsilon}{\sum_{i=1}^{N} p_i^2 + \sum_{i=1}^{N} y_i^2 + \epsilon} \tag{4}$$

where N is the total number of pixels in the sample, p_i is the predicted probability value (predicted pixel value) by the model, and y_i is the true label value (actual pixel value). ϵ is a smoothing constant used to prevent division by zero errors, set to 1e-6.

III. Results and Discussion

A. Image Data Construction

In this study, two types of datasets were established. The first dataset consisted of 66 images of defect-free lithium-ion

pouch cells, collected using industrial cameras. The second dataset included 40 ultrasound images of defect-free lithium-ion pouch cells obtained through ultrasonic non-destructive testing. Each class of images was augmented through affine transformations to expand the total to 200 images. Additionally, 20 images of defective pouch cells and 20 ultrasound images of batteries containing bubbles were collected.

All images varied in size from 200×200 pixels to 700×700 pixels. The training dataset exclusively included defect-free images, while the testing dataset contained both defect-free and defective images. To ensure class balance during training, Poisson image editing techniques were applied for data augmentation. Specifically, defects were randomly generated on each defect-free image, creating a corresponding defective image for every original, thereby ensuring a 1:1 ratio of defect-free to defective images, as shown in Fig. 2. This method guaranteed that the number of defect-free images equaled the number of defective images in the training dataset, effectively mitigating the negative impact of class imbalance on model training.

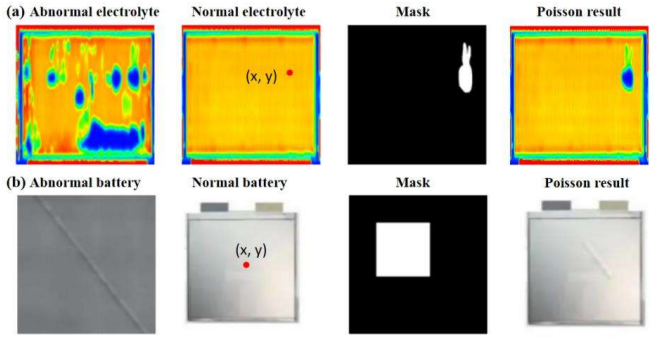

Fig. 2. Creation of Defect Data Using Poisson Image Editing Technique

Poisson Image Editing is an advanced image editing technique that achieves a smooth blending effect by finely adjusting the local pixels of an image [10]. The expression for the Poisson equation is as follows:

$$\nabla^2 u = \nabla \cdot b \tag{5}$$

where, In this context, u is the image to be solved, b represents the gradient information in the image, ∇ is the gradient operator, and ∇^2 is the Laplace operator. In image blending, the image u is obtained by minimizing the following energy function:

$$E(u) = \frac{1}{2}\int_{\Omega} (\nabla u - b)^2 \, d \tag{6}$$

Here, Ω is the defined domain, ∇u represents the gradient of the blended image, and b is the gradient information from the input image. This energy function ensures that the gradient of the blended image u is as close as possible to the gradient b of the input image, thereby ensuring a smooth transition and natural blending of the images.

Specifically, the defect areas in the source image are labeled with a mask, creating a binary mask image that matches the dimensions of the source image, where the white region indicates the area to be cloned. The precise coordinates of the mask are generated using a random function to ensure diversity in the cloned area. Additionally, the coordinates of the center point of the blending area are determined in a randomly selected defect-free image (the target image), denoted as (x, y), to accurately locate the region to be blended. As shown in Figure 2, the red dot indicates the exact position of the blending center point. This method of randomly selecting coordinates not only increases the diversity and flexibility of the cloning operation but also ensures that the defect area can seamlessly blend with the target image, significantly enhancing the overall visual consistency.

Poisson blending utilizes the gradient information of the target image, ensuring a natural transition in color, brightness, and texture within the blending area, resulting in an almost imperceptible boundary effect. This method effectively preserves the integrity of the target image's background, enabling high-fidelity cloning operations even in complex backgrounds. Poisson blending is suitable for seamlessly inserting one image patch into another, avoiding issues of abrupt edges or inconsistent lighting that traditional copying methods may cause, thereby enhancing the realism and naturalness of the blended area. It serves as a powerful tool in the field of image processing.

B. Text Data Preparation

Each set of image data is accompanied by corresponding text descriptions, which not only aids in leveraging textual information to enhance the accuracy of defect detection but also imbues the detection results with semantic characteristics. This facilitates more effective interaction and integration with large language models, thereby improving the model's interpretability and interactivity. This paper reviews the literature related to prompt engineering, aiming to optimize the output accuracy of the language model.

To enable the model to more accurately recognize and describe defect information, specific prompt templates have been designed, incorporating various characteristics of defects in their descriptions. These templates provide detailed descriptions of both defective and non-defective states while integrating key information such as defect type, morphology, and location. This guides the model to generate outputs that are more aligned with real-world scenarios. This approach enhances the language model's understanding and responsiveness to defect detection tasks and improves the accuracy of defect identification and the interpretability of outputs.

When designing the prompts, precise descriptions were employed for non-defective batteries, such as “[Battery]”, “[Battery without flaw]”, “[Battery without defect]”, and “[Battery without damage]”. The purpose of these expressions is to clearly define the intact state of the battery, providing the model with unambiguous non-defective reference samples. For defective batteries, more detailed descriptions were crafted, including “[Broken battery]”, “[Damaged battery]”, “[Battery with flaw]”, “[Battery with defect]”, “[Battery with damage]”,

"[Battery with scratch]", and "[Battery with dent]". These prompts cover various defect types such as breakage, scratches, and dents, while enriched language descriptions enhance the model's ability to recognize defect characteristics, enabling it to more accurately identify and distinguish various defect states of the battery. This design helps improve the model's discernibility and output accuracy in complex detection scenarios.

The design of the prompt templates employs a flexible approach, wherein the placeholder $[d]$ can be replaced with previously mentioned descriptions of either non-defective or defective states, thus encompassing various characteristics of the battery's condition. Specifically, the prompt templates include the following forms: "a photo of a $[d]$", "a photo of the $[d]$", "a photo of a $[d]$ for anomaly detection", and "a photo of the $[d]$ for anomaly detection".

This templated design effectively guides the language model to focus on the physical characteristics and defect conditions of the battery by incorporating detailed descriptions of its states. Additionally, the specific prompts tailored for anomaly detection further reinforce the model's focus on defect identification tasks, enhancing its ability to discern and respond accurately to abnormal features. This flexible and targeted prompt strategy significantly improves the model's detection performance, providing robust support for battery defect analysis.

C. *Training Process and Results*

During the training phase, the weights of Image Encoder 1 (ViT-H, 630M parameters) and the Text Encoder (OpenCLIP, 302M parameters) were fixed. Image Encoder 2 was initialized with the weights from ViT-H and subsequently participated in gradient updates alongside the other network components. To accelerate the forward propagation speed, the pixel dimensions of all input images were resized to 224×224.

The model was trained on two A100 GPUs, with a total of 80 training epochs. During training, the batch size was set to 32, and a gradient accumulation strategy was employed to optimize memory usage. The learning rate was set to 1e-3, and the Adam algorithm was chosen as the optimizer. These configurations ensured that the model could be efficiently trained on large-scale datasets while effectively reducing the consumption of computational resources.

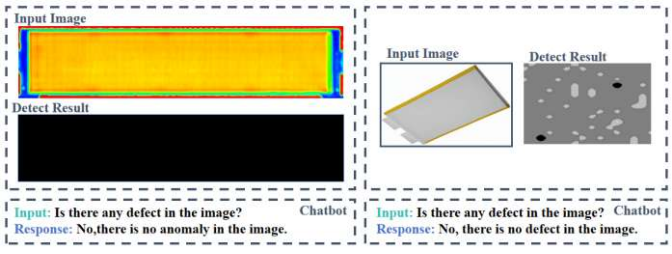

Fig. 3. Detection Results of Battery Images without defect

Fig. 3. presents the detection results for flawless batteries, including the identification of defects in pouch cells and internal ultrasound images. In analyzing these two images, the model accurately classified them as defect-free. First, in the detection results of the battery ultrasound image, the model displayed a completely black image, indicating that no abnormal signals were detected, which aligns with the defect-free labeling. Second, in the detection of the pouch cells, the model's output exhibited three different grayscale levels. This grayscale variation arises because both detection images utilized the same threshold settings, while the battery casing image included variations in background and battery thickness. Consequently, the model demonstrated a high sensitivity to object edges within the image, capable of capturing subtle structural differences. Despite these grayscale changes, the model still accurately identified the battery casing as defect-free. This achievement highlights the model's flexible adaptability when handling different types of images, maintaining high accuracy even in complex backgrounds or under conditions of high edge sensitivity. This confirms the model's significant robustness in defect judgment across diverse imaging scenarios.

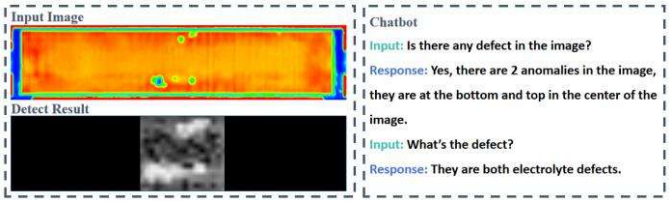

Fig. 4. The detection results of ultrasound images for defective batteries.

Fig. 4. presents the detection results of the model on defective images and its application in multi-turn dialogue. Taking the ultrasound image of the battery as an example, BatteryGPT successfully identified it as an ultrasound image of the electrolyte and detected the presence of two bubble defects within the electrolyte. Further analysis revealed that the model accurately located the clustered areas of the bubbles, with one cluster positioned in the upper half of the center and another in the lower half. This approach allows BatteryGPT not only to efficiently detect defects but also to transform complex visual information into semantically meaningful descriptions, facilitating natural language interaction and in-depth analysis with users. This process demonstrates BatteryGPT's strong capabilities in defect detection and semantic interaction, particularly in the efficient information transfer and understanding achieved after integration with a large language model.

TABLE I. IMAGE AUC OF DIFFERENT MODELS

Method	BatteryGPT	CutPaste	SCADN
Optical images of pouch cells	94.29	90.32	89.31
Ultrasound images of pouch cells	98.68	91.77	96.82
Mean	96.49	91.05	93.07

BatteryGPT was compared with two unsupervised anomaly detection models, CutPaste and SCADN, and the results are shown in Table I. BatteryGPT demonstrated superior Image AUC (Area Under Curve) across all categories. The CutPaste model generates pseudo-anomalous samples by randomly cutting and pasting image patches on normal data, using the spatial inconsistency of the patches to roughly simulate real defects. This model employs a CNN (Convolutional Neural

Network) architecture. On the other hand, the SCADN model identifies anomalies by masking certain areas of the image and reconstructing it using a GAN (Generative Adversarial Network). Both methods are typical representatives of unsupervised anomaly detection, therefore they were chosen as comparison benchmarks. To ensure fairness in the comparison, all three models were trained under the same conditions for 80 epochs, with a learning rate set at 1e-3 and using the Adam optimization algorithm. The comparison results indicate that Transformer-based models have the potential to become a new benchmark in the field of anomaly detection. Notably, when integrated with textual information, the model's sensitivity to anomalous structures is further enhanced.

IV. Conclusion

This paper presents a novel multimodal large language model, BatteryGPT. The model integrates information from both visual and linguistic modalities to successfully achieve defect detection in lithium-ion batteries. By combining the powerful language processing capabilities of large language models (LLMs) with advanced visual recognition algorithms, BatteryGPT can identify and analyze various complex defects in lithium-ion batteries. Through the establishment of a multimodal information fusion mechanism, BatteryGPT simultaneously integrates image features and textual descriptions, enabling defect segmentation and localization while producing precise and comprehensible defect descriptions, achieving an average Image AUC of 96.49%. Furthermore, a high-quality dataset containing a variety of battery samples has been constructed. In the future, the performance and applicability of BatteryGPT will be further enhanced by incorporating more battery-specific knowledge, process parameters, and historical data. Additionally, improving its autonomous learning and adaptability, along with deepening collaboration with the industry, will facilitate its validation and widespread adoption in large-scale industrial applications. In summary, BatteryGPT not only provides a new solution for defect detection in lithium-ion batteries but also lays the groundwork for the future development of intelligent battery inspection systems, further advancing the evolution of battery energy storage systems.

Acknowledgment

This work was supported by the Science and Technology Project of State Grid Corporation of China (Development of high-performance special extinguishing agent for lithium-ion battery fire, No. 5500-202220118A-1-1-ZN).

References

[1] X. D. Yan, H. D. Zhang, X. M. Xu, X. W. Hu, P. A. Heng. "Learning Semantic Context from Normal Samples for Unsupervised Anomaly Detection," 35th AAAI Conference on Artificial Intelligence / 33rd Conference on Innovative Applications of Artificial Intelligence / 11th Symposium on Educational Advances in Artificial Intelligence, Electr Network, Vol. 35, pp 3110-3118, Feb 02-09, 2021.

[2] J. Pirnay, K. Chai. "Inpainting Transformer for Anomaly Detection," 21st International Conference on Image Analysis and Processing (ICIAP), Lecce, ITALY, Vol. 13232, pp 394-406, May 23-27, 2022.

[3] J. Wyatt, A. Leach, S. M. Schmon, C. G. Willcocks, Ieee. "AnoDDPM: Anomaly Detection with Denoising Diffusion Probabilistic Models using Simplex Noise," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), New Orleans, LA, pp 649-655, Jun 18-24, 2022.

[4] C. L. Li, K. Sohn, J. Yoon, T. Pfister, S. O. C. Ieee Comp. "CutPaste: Self-Supervised Learning for Anomaly Detection and Localization," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Electr Network, pp 9659-9669, Jun 19-25, 2021.

[5] J. Jeong, Y. Zou, T. Kim, D. Zhang, A. Ravichandran, O. Dabeer, et al. "WinCLIP: Zero-/Few-Shot Anomaly Classification and Segmentation," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, CANADA, pp 19606-19616, Jun 17-24, 2023.

[6] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, et al. "Learning Transferable Visual Models From Natural Language Supervision," International Conference on Machine Learning (ICML), Electr Network, Vol. 139, Jul 18-24, 2021.

[7] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, et al. "Training language models to follow instructions with human feedback," 36th Conference on Neural Information Processing Systems (NeurIPS), Electr Network, Nov 28-Dec 09, 2022.

[8] C. L. Yin, K. P. Du, Q. Nong, H. C. Zhang, L. Yang, B. Yan, et al. PowerPulse: Power energy chat model with LLaMA model fine-tuned on Chinese and power sector domain knowledge. Expert Systems, 2024, 41(3):

[9] K. Han, Y. H. Wang, H. T. Chen, X. H. Chen, J. Y. Guo, Z. H. Liu, et al. A Survey on Vision Transformer. Ieee Transactions on Pattern Analysis and Machine Intelligence, 2023, 45(1): 87-110

[10] P. Pérez, M. Gangnet, A. Blake. Poisson image editing. Acm Transactions on Graphics, 2003, 22(3): 313-318

[11] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. See https://vicuna.lmsys.org (accessed 14 April 2023), 2023.

[12] Yixuan Su, Tian Lan, Huayang Li, Jialu Xu, Yan Wang, and Deng Cai. Pandagpt: One model to instruction-follow them all. arXiv preprint arXiv:2305.16355, 2023.