Web Chat-based Application with Large Language Model and Transformers from Hugging Face for Self-Learning on Storytelling Skills

Victoria Agatha[a], Iwan Setyawan[b]
Faculty of Electronics and Computer Engineering
Satya Wacana Christian University
[a]victoriatha61@gmail.com, [b]iwan.setyawan@ieee.org

2024 International Electronics Symposium (IES) | 979-8-3503-9199-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/IES63037.2024.10665795

***Abstract*— The ability of storytelling greatly affects a person's success. These abilities are usually taught by parents at home or teachers at school. With the development of Artificial Intelligent technology, it is now possible to automatically generate stories using Large Language Model (LLM) which can understand and create language like humans. In this paper, the authors propose a self-learning system in storytelling by utilizing and combining four models from Hugging Face Hub. The proposed system is a web chat-based application so that users can communicate with LLM where LLM has received an image input from the users. The four models are as follows. Falcon 7B Instruct model as LLM that gets caption information from BLIP Image Captioning Large model. Its responses in the form of text can be read by the users and can be heard through audio synthesized by the MMS TTS Eng model. The user can also see the detected objects in the image which is detected by DETR ResNet 50 model. Our experiments show that the proposed system is sufficient to produce a good story and fit the context of the image, with an average user score of 89.76.**

Keywords*— *Large Language Model, Storytelling, Hugging Face

I. INTRODUCTION

A person's success is greatly influenced by storytelling skills. Storytelling is a basic human need as social beings who are always in communication with each other [1]. The benefits of storytelling are that it helps develop critical thinking skills, creativity, active participation or engagement in learning, literacy skills, narrative thinking abilities, self-exploration, and interpersonal skills [2]. Storytelling skills are usually taught in the family sphere by parents and school scope by teachers. As technology develops, especially Artificial Intelligence (AI), this conventional method can be replaced by the automation of Machine Learning (ML) and Deep Learning (DL) models in generating stories. The model is Large Language Model (LLM), which is a Natural Language Processing (NLP) algorithm based on ML and DL techniques that have been trained with large amounts of text data so that they can understand and create human-like text, answer questions, and complete other language-related tasks with high accuracy [3], especially in generating story.

Existing NLP applications for storytelling can only generate stories from images which are considered less helpful in developing the users' storytelling skills such as in [4]. Therefore, the novelty and main contribution of this paper is the proposal of a self-learning method to develop storytelling skills in the form of a web chat application where users are asked to input an image then user can chat with chatbot to ask anything related to the image or just ask for a story based on the image. The authors utilize the open-source Transformer library from Hugging Face to use the models shared in Hugging Face Hub for free [5]. An additional feature to support this learning is that the output story in the form of text can be heard via audio so that users can learn how to pronounce it. Another feature is the object detector to show what classification of objects are in the image.

This paper is organized as follows. Section 2 of this paper will describe an overview of related works found in the literature. Section 3 explains in more detail the methods used in the proposed system. The results and experiments are discussed in Section 4. Finally, the conclusion and our future works are shown in Section 5.

II. RELATED WORKS

The storyteller applications have been developed with various combinations of models so that they can accept image input and produce creative stories. Most of these applications use Generative Pre-trained Transformer (GPT) models such as GPT 3.5 turbo or GPT 4 to perform text generation tasks as in [6]. However, the use of this model will incur Application Programming Interface (API) fees by OpenAI developer. Therefore, the author replaced it with the Falcon 7B Instruct model from Hugging Face which can be accessed for free. Hugging Face provides transformers, which are Python libraries that will download models and run them locally so that models shared in Hugging Face Hub can be used for free [7]. PyTorch is required for the installation process of these models.

Falcon 7B Instruct is a ready to use chat or instruct model based on Falcon 7B. Falcon 7B is a strong base model, outperforming comparable open-source models (e.g., MPT 7B, StableLM, RedPajama, etc.) [8]. The advantages of using Falcon 7B Instruct as an LLM to produce narrative stories are smaller storage size, cheaper training models, can create a variety of creative content, better grammar compared to LLaMa 40B which is more used for conversation [9].

In learning to practice storytelling, story visualization in text is not the only thing needed. Audio presentation of the story is also needed in order to enable users to practice as a listener and speaker so that they can improve their comprehension and pronunciation proficiency. Some of the previous works found in literature did not implement this, for example the works found in a [10] and [11].

III. PROPOSED SYSTEM

This paper proposes a self-learning system in storytelling by utilizing and combining a total of four models from Hugging Face Hub. Self-learning system through web application with the main feature, users can import images and ask the chatbot to create stories. With this chat feature, users can also ask anything related to the image or story. The additional features that are quite helpful in the learning process are audio and object detector. The flowchart of the proposed system is presented in Fig. 1.

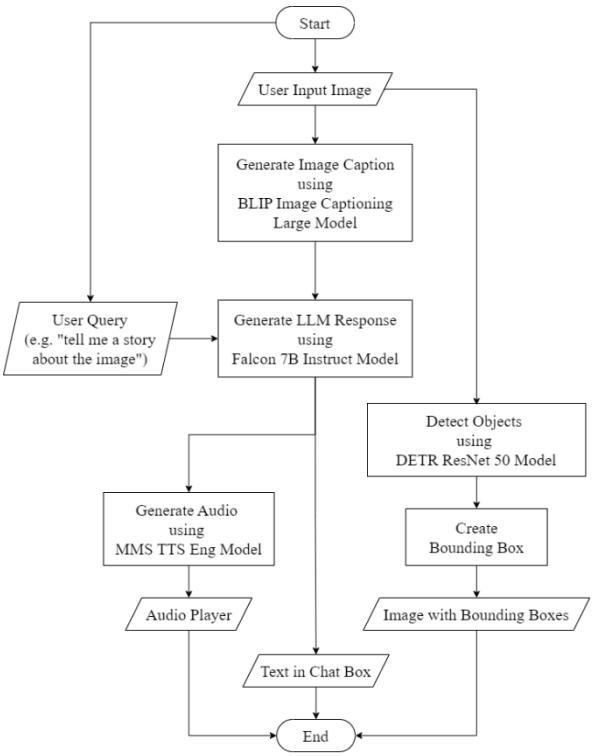


Fig. 1. Flowchart of the proposed system

The flowchart in Fig. 1 shows that the self-learning system in storytelling starts by receiving image input from the user and then generating an image caption using the BLIP Image Captioning Large model and the caption will be used as one of the inputs to the Falcon 7B Instruct model. If the user does not enter an image, the caption input will be considered to be blank. Another input from the Falcon 7B Instruct model is a user query in the form of text, e.g. “Tell me a story about the image”. The Falcon 7B Instruct model will generate an LLM response in the form of text displayed in a chat box. The text output is also converted into audio form by the MMS TTS Eng model so that users can listen and learn how to pronounce the words by using the audio player.

To find out what objects are present in the image, the DETR ResNet 50 model is utilized in detecting objects and generating the values of the four bounding box coordinate points and the class name of each detected object. From the values of these points, they are connected by lines to form a complete bounding box for each detected object and then displayed in the image along with the the name of its respective class. The following is a more detailed explanation of each step performed in the proposed system.

A. Web Chat Application

Streamlit is a free and open-source framework for creating web applications for data scientists and machine learning engineers. The author takes advantage of the framework because it can build an attractive User Interface (UI) quickly, only with Python programming language without requiring front end programming experience [12]. Streamlit provides quick access so that users can select files and upload them. The images to be processed by the models used in this application must be in jpg, jpeg, png, tiff, webp, gif, and tif formats. The UI for image file capture is shown in Fig. 2. After the file is uploaded, the file will be displayed and written caption below the image file.

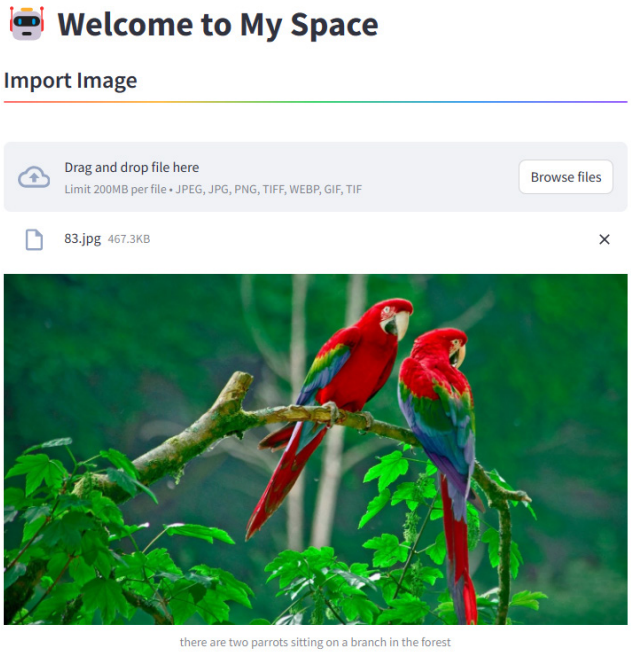


Fig. 2. Example of displayed image and caption

B. Image Caption

To produce a caption for an image, one of the models available in Hugging Face Hub is BLIP Image Captioning Large. Bootstrapping Language-Image Pre-training or BLIP is a Vision Language Pre-Training (VLP) framework that can cover a wider range of tasks derived from understanding and producing visual-language [13]. This BLIP is more effective because it involves two models, namely Multimodal mixture of Encoder-Decoder (MED) and Captioning and Filtering (CapFilt). The MED, which is a visual-language integrated model is trained with CapFilt which plays a role in utilizing noisy web data by bootstrapping text. Synthesis text will be generated captioner and noisy text removed with filters.

C. Chatbot

Falcon 7B Instruct is a causal decoder model with 7B parameters created by TII based Falcon 7B and finetuned on a mix of chat or instruction datasets. This model is licensed under Apache 2.0. Falcon 7B has been trained on 1500B tokens from RefinedWeb enhanced with curated Corpora [8].

LangChain is an open-source framework to help develop NLP and AI-based applications such as chatbots, virtual agents, and others [14]. Falcon 7B Instruct imported from Hugging Face Hub as LLM. Then, with the LLMChain function from the LangChain library, the LLM will be associated with a prompt or command containing a template to act as a chatbot to answer user questions or requests based on the caption of the image input provided by the user. User queries can be asked on the chatbot by entering text input in the box provided and the chat history is presented in Fig. 3.

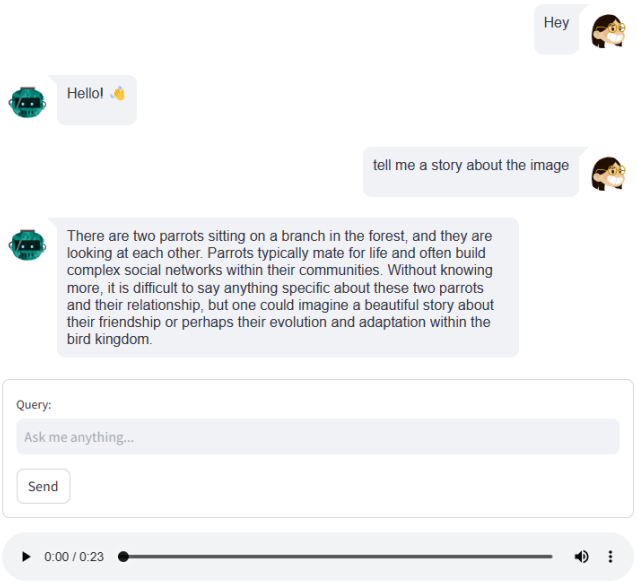


Fig. 3. Example of user-chatbot conversation and audio player

D. Object Detection

The DETR model stands for DEtection TRansformer used to detect objects in images. This model is based on Transformers and bipartite loss matching for direct and parallel set prediction. DETR uses ResNet 50 from the Convolutional Neural Network (CNN) architecture to study the input representation of 2D images. The model will flatten and complete it with positional encoding and then given to the transformer encoder. The transformer decoder will take input from the output encoder and object queries, i.e. a small number of positional embeddings that have been learned. Output embeddings from the decoder will be given a Feed Forward Network (FFN) and bipartite matching loss will predict bounding boxes with the classification of presence or not (no object ∅) objects [15].

Seen in Fig. 4, the application provides a button to generate bounding box images of objects that have been detected by the model. In the bounding box there is also the class name of the object. Below the image, information is displayed about the bounding box coordinates, class names, and the probabilities of these objects.

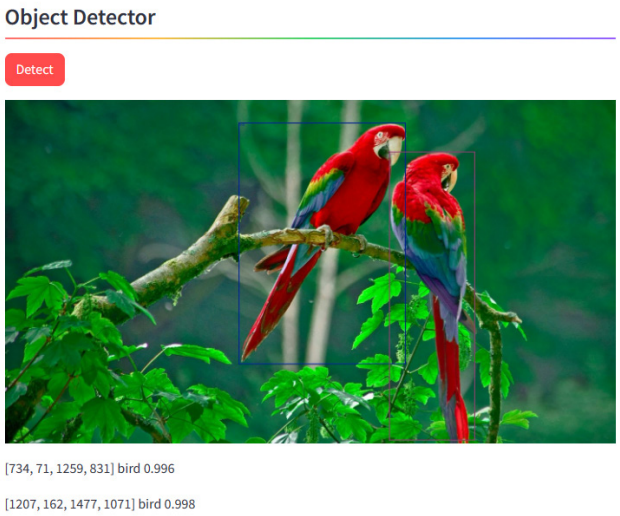


Fig. 4. Detected objects of a sample image with bounding boxes

E. Text to Audio Convertion

Massive Multilingual Speech (MMS) contains Text to Speech (TTS) in English (Eng) is used to convert LLM text responses to audio shown in Fig. 3. Variational Inference with adversarial learning end to end Text to Speech is an end-to-end speech synthesis model that predicts speech waveforms based on the sequence of input text [16]. This model is a conditional Variational Autoencoder (VAE) consisting of posterior encoder, decoder, and conditional prior. A set of spectrogram acoustic signals is predicted by a flow-based module, which is formed a Transformer text encoder and several coupling layers. The spectrogram is decoded with a stack of transposed convolutional layers, such as the HiFi-GAN vocoder. The model also consists of stochastic duration predictors so that it can synthesize speech with varying rhythms from the same text input.

IV. Result and Disscusion

The experiment was conducted with 100 random images taken from the internet. One by one images are uploaded and the chatbot is asked to tell a story about the image. LLM responses in the form of text from each of the 100 random image inputs were assessed by 10 users in the range of 0-100, with 0 being the lowest score and 100 the highest score indicating a perfect match between the image and the generated story. The average user scores for each of the 100 images are shown in Fig. 5.

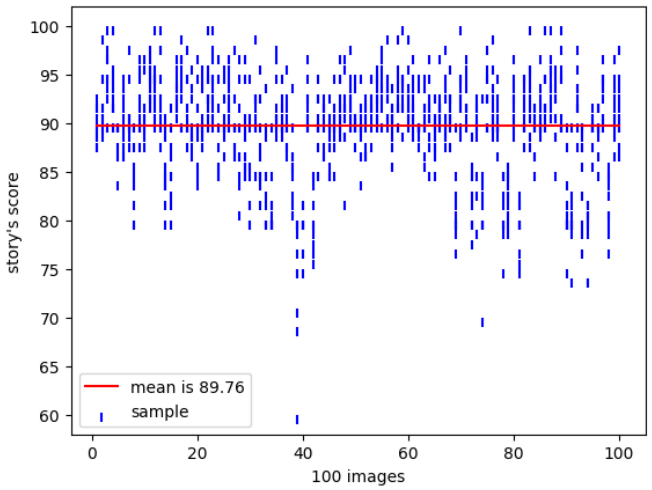


Fig. 5. User scores for stories generated using 100 random images

The overall average score obtained is 89.76, as shown in Table I. It can be seen that 66.6% of the user rating samples is above the overall average value. This shows that the system has been able to achieve the target to produce a good story and fit the context of the image. However, the proposed system still has difficulties in consistently producing good stories for each input images. In other words, the system is unable to generate stories with satisfactory quality for some of the input images. This can be seen in Fig. 5 and is also shown in Table I by the relatively large standard deviation of the user score (both for individual users and the average standard deviation for all 10 users).

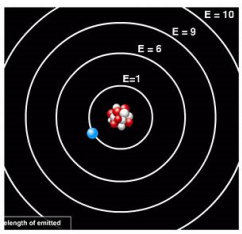
TABLE I. MEAN AND STANDARD DEVIATION OF USER SCORES

User	Mean	Standard Deviation
1	93.42	6.94
2	90.48	4.70
3	89.02	5.52
4	89.63	4.50
5	89.40	5.52
6	89.73	4.16
7	89.69	5.35
8	88.69	4.14
9	89.52	5.07
10	88.01	4.06
Average	89.76	5.0

Fig. 6 is an example of two pairs of images and stories that fit and don't fit the context. The story that does not fit the context is due to the BLIP Image Captioning Large model which cannot produce caption that match the image. This can be influenced by image quality factor or lack of vocabulary owned by the BLIP Image Captioning Large model or the LLM model itself, the Falcon 7B instruct. The image shown in Fig. 6(b) is also the image with the largest user score variation (standard deviation of 6.09), which may indicate that the image is indeed hard to interpret/describe or that it can be interpreted in a lot different ways.

I am standing in a field of sunflowers, a woman in a yellow dress with a sunflower in her hand. The sun is shining down on me refreshingly, and I'm surrounded by the vibrant colors and beauty of nature--rich greenery, a breathtaking view of the sky, and the itchy feeling of the fuzz on the inside of a sunflower. In this moment, I'm feeling content and thankful. I close my eyes as I take in the sweet scent of the sun.

(a)



Once upon a time, a little robot named Spot was wandering around in space and came across a mysterious object. He was so intrigued he decided to approach it and investigate. He was astonished to find a strange machine that seemed to be zipping through the stars at incredible speeds. He followed it to its source, the main controls and power source located at the core of the machine.

(b)

Fig. 6. Example of (a) picture number 35 with well-matched story and (b) picture number 39 with poorly-matched story

In order to find out the relative performance of the proposed system to other story-generating AI, we compared the performance of our proposed system to Microsoft's Copilot. Copilot is an AI assistant that serves as an everyday companion for various tasks. It's a conversational chat interface to search for information, supporting task like answering questions, generate text like emails and summaries, and create images based on text prompts [17]. It can also write code in several programming languages. Copilot is integrated into Windows, Edge, Office apps, and Bing, offering unique features in each platform. It's a sophisticated AI system that combines OpenAI's GPT models with Microsoft's technologies. This includes the Bing search engine's web-scraping database, Microsoft Natural Language Processing, Text to Speech (TTS) for generating lifelike speech responses, Retrieval Augmentation Generation (RAG) to ground and add context, and Azure cloud services [17]. It operates in different modes such as balanced, precise, and creative, to cater to various user needs. To perform the comparison, we fed the same 100 images used in the previous experiment to Copilot and asked it to generate stories from each image. We then asked the same 10 users to rate the stories generated by Copilot. The average user score (and its standard deviation) for each Copilot mode is summarized in Table II.

TABLE II. MEAN AND STANDARD SCORES OF EACH COPILOT MODE

Copilot Mode	Mean	Standard Deviation
Creative	91.51	2.33
Balanced	91.52	2.67
Precise	92.26	2.60

From this table, we can see that Copilot gives a better performance both in terms of average user score and consistency (as shown by the lower standard deviation of the user scores). The main advantage of Copilot is the use of OpenAI's proprietary GPT model. This engine is very powerful, but the use of this engine to develop an application such as the one proposed in this paper requires substantial monetary investment. Therefore, we can argue that our proposed system, which is based on free engine, still give a comparable performance (average user score differing by less than 2% compared to most Copilot modes).

V. CONCLUSION AND FUTURE WORKS

In this paper, the authors propose a web chat-based application for self-learning on storytelling skills with additional audio feature and image object detector. This application uses models from Hugging Face which are free to use. The authors tested the proposed system by rating stories generated from 100 images. The rating is performed by 10 users and the result is presented in Fig. 5. The average and standard deviation of each user is presented in Table I. Based on these results we can conclude that the quality of the stories generated by the system is good as indicated by the average user score of 89.76. However, the authors also note that the average standard deviation of the scores is not insignificant at 5. This indicates that the system still has problems generating good (matching) stories for some of the images. In general, the user score variation is similar for each input image, except for the image shown in Fig. 6(b). The proposed system give respectable performance compared to commercial, proprietary systems such as Microsoft's Copilot with only a less than 2% lower score, on average, on most modes.

In the future, the authors will continue to improve the proposed system, for example by experimenting on other captioner models that can produce more accurate image captions. We will also do more extensive testing of the proposed system by involving more users. A test of the quality of the input image (i.e., how difficult it is for a human to create a story describing an image) will also be very useful to determine the relative performance of the system. Furthermore, the authors will perform a more comprehensive statistical analysis of the results. Finally, the authors will add an option to use more than one language so that users can learn the desired language, not only in English.

REFERENCES

[1] S. E. Worth, "Storytelling and narrative knowing: An examination of the epistemic benefits of well-told stories," *Journal of Aesthetic Education*, vol. 42, no. 3. pp. 42–56, Sep. 2008, doi: 10.1353/jae.0.0014.

[2] D. E. Agosto, "If I Had Three Wishes: The Educational and Social/Emotional Benefits of Oral Storytelling," *Storytell. Self, Soc.*, vol. 9, no. 1, pp. 53–76, 2013, doi: 10.13110/storselfsoci.9.1.0053.

[3] E. Kasneci *et al.*, "ChatGPT for good? On opportunities and challenges of large language models for education," *Learning and Individual Differences*, vol. 103. 2023, doi: 10.1016/j.lindif.2023.102274.

[4] S. S. Singh, "A Large Language Model (LLM) Based App to Generate Stories from Pictures," 2023. https://github.com/sssingh/pic-to-story (accessed Mar. 25, 2024).

[5] S. M. Jain, *Introduction to Transformers for NLP: With the Hugging Face Library and Models to Solve Problems*. 2022.

[6] Z. Xie, T. Cohn, and J. H. Lau, "The Next Chapter: A Study of Large Language Models in Storytelling," 2023, doi: 10.18653/v1/2023.inlg-main.23.

[7] F. Tomas, "6 Ways For Running A Local LLM (how to use HuggingFace)." https://semaphoreci.com/blog/local-llm.

[8] E. Almazrouei *et al.*, "The Falcon Series of Open Language Models," vol. 2, pp. 1–57, 2023, [Online]. Available: http://arxiv.org/abs/2311.16867.

[9] V. Johnson, "Falcon vs. LLaMA: A Comparison of Two Large Language Models." https://www.cloudbooklet.com/ai-text/falcon-vs-llama-which-llm-is-better (accessed Apr. 01, 2024).

[10] C. Zang *et al.*, "Let Storytelling Tell Vivid Stories: An Expressive and Fluent Multimodal Storyteller," vol. 1, pp. 1–8, 2024, [Online]. Available: http://arxiv.org/abs/2403.07301.

[11] W. Wang, C. Zhao, H. Chen, Z. Chen, K. Zheng, and C. Shen, "AutoStory: Generating Diverse Storytelling Images with Minimal Human Effort," vol. 1, pp. 1–19, 2023, [Online]. Available: https://arxiv.org/abs/2311.11243.

[12] "A faster way to build and share data apps." https://streamlit.io/ (accessed Feb. 08, 2024).

[13] J. Li, D. Li, C. Xiong, and S. Hoi, "BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation," in *Proceedings of Machine Learning Research*, 2022, vol. 162, pp. 12888–12900, [Online]. Available: https://arxiv.org/abs/2201.12086.

[14] "What is LangChain?" https://www.ibm.com/topics/langchain (accessed Feb. 11, 2024).

[15] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-End Object Detection with Transformers," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2020, vol. 12346 LNCS, doi: 10.1007/978-3-030-58452-8_13.

[16] V. Pratap *et al.*, "Scaling Speech Technology to 1,000+ Languages," vol. 1, pp. 1–41, 2023, [Online]. Available: https://arxiv.org/abs/2305.13516.

[17] M. Muchmore, "What Is Copilot? Microsoft's AI Assistant Explained," 2024. https://www.pcmag.com/explainers/what-is-microsoft-copilot (accessed Jun. 01, 2024).