Image Caption Generator (Captionify)

Khushbakht Khan[1], Sarah Sami[2], Junaid Jamshed[3], Syed Aun Muhammad[4], Moiz Naveed[5]*

*SE-21009, SE-21026, SE-21035, SE-21036, SE-21048**

Department of Software Engineering, NED University of Engineering and Technology, Karachi, Pakistan

Submitted on 3rd January 2024

Abstract

This research explores the evolution of image captioning models in the era of visual content dominance, presenting the development of an advanced image caption generator. Drawing inspiration from traditional Encoder-decoder models and contemporary transformer architectures such as CLIP and ViT, the approach utilizes the Hugging Face Transformers library. Google Colab serves as the platform for resource-intensive model training, followed by server hosting and API integration for seamless interaction with the frontend. The incorporation of a Chatbot API refines generated captions based on user-defined preferences, enhancing aesthetic appeal. The React-based user interface, optimized with Vite, ensures responsiveness, while the Node.js and Express.js backend provides a scalable foundation. React Query expedites API utilization and data retrieval. This paper details a comprehensive process aligned with industry best practices, resulting in an innovative and sophisticated image caption generator that navigates both model architecture and user experience intricacies.

***Keywords**: Image Captioning, Encoder-Decoder Models, Transformer Architectures, Hugging Face Transformers Library, Google Colab, API Integration, React User Interface, Node.js and Express.js, Backend, Frontend*.

1. Introduction

In In the era of visual content dominance, the demand for sophisticated image captioning models has surged, prompting continuous evolution in the field. This research delves into the development of an advanced image caption generator, synthesizing insights from the rich landscape of earlier models, such as Encoder-decoder and Attention-based architectures. Drawing inspiration from recent breakthroughs in transformer models like CLIP and ViT, our approach leverages the Hugging Face Transformers library to construct a robust deep learning model.

The methodology employed encompasses the use of Google Colab for resource-intensive training [1], followed by server hosting and API integration for seamless application interaction. Notably, the incorporation of a Chatbot API enhances the aesthetic appeal of generated captions by considering user-defined preferences. The user interface, developed using React and optimized with Vite [2], ensures responsiveness, while the backend structure, built on Node.js and Express.js, provides a scalable foundation. The integration of React Query expedites API utilization and data retrieval [3].

This paper delineates a comprehensive process, aligning with industry best practices, from model development to frontend deployment. The resulting image caption generator stands as an innovative solution, navigating the intricacies of both model architecture and user experience.

2. Related Work

The need for image captioning models has risen long ago and hence there exists many similar models. Although the means used for creating such models differ verily. The early models include Encoder-decoder models and Attention based models. Furthermore, recently there are transformer-based models and Generative pre-trained transformers (GPTs).

2.1 Early Models

These innovative models utilized networks (CNNs) to capture visual characteristics from images and recurrent neural networks (RNNs) like LSTMs or GRUs to decode them into captions. Examples include “Show and Tell” models. They operate by utilizing networks (CNNs) to extract features from the image. Create a vector of a

specific length. This vector is then inputted into the network (RNN) which generates words sequentially predicting each word based on the preceding words and the encoded image features [4]. The drawbacks of these models include, high expense since training can sometimes require a lot of resources and It's possible that attention may not consistently prioritize the relevant areas.

2.2 Recent Advancements.

Recently, transformer models were introduced. Recent advancements, in natural language processing have shown promise with models, like CLIP (Contrastive Language Image Pretraining) and ViT (Vision Transformer). These models have proven to be highly effective when it comes to image tasks. Use Transformers for both encoding and decoding which enables processing and improves the connection, between words, over distances [5]. As a result the captions become grammatically accurate. Flow smoothly. Their drawbacks include, the possibility that they may not possess common sense reasoning or a comprehensive understanding of real world situations. Additionally there is a chance that it might generate errors due, to misinterpretations of the image.

The field of image captioning has undergone changes over time. It started with models like Show and Tell then progressed to include attention mechanisms in Show, Attend and Tell. Further advancements have been made with models that utilize bottom up and top down attention. Recently transformer based architectures like CLIP and ViT have opened up opportunities, for multimodal understanding [5]. Some of the existing models were also modified like “Show, Tell and Polish” [6] and “Entangled Transformer” [7]. Furthermore, they were more comprehensive surveys conducted to determine the most efficient approach [8].

3. Methodology

3.1 Working:

The image caption generator takes an approach using insights, from machine learning models that specialize in creating captions for images. It starts by preprocessing the COCO dataset [18] which involves making sure all images are of uniform size and captions are tokenized. To extract features a vision model based on Transformers is used to convert information into a context vector. For generating captions a language model called GPT 2 is used. During the training phase the goal is to optimize the model by minimizing any differences between actual captions using pairs of images and their corresponding features, from the COCO dataset. The models performance is continuously improved through evaluation and fine tuning adjusting parameters to produce high quality captions. When it comes to generating captions for images during inference the trained model uses both an encoder and a decoder to process them [19]. The result is an image caption generator that can provide descriptive captions for a wide range of images.

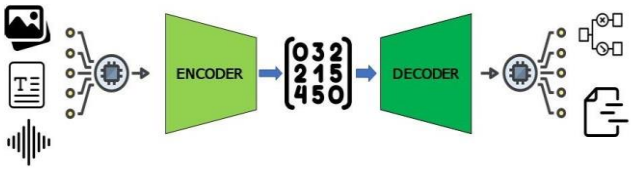

Figure 1: Encoder and Decoder

3.2 Construction:

In constructing the image caption generator, our initial focus was, on building a deep learning model using the Hugging Face [6] Transformers library [7]. We used Google Colab as our platform, which provided us with the resources for training. After that we hosted the model on a server. Wrapped it in an API to make it easier for the application to interact with. To ensure communication between the application and the hosted model we carefully defined an interaction protocol that allowed for transmission of images to the API. This process played a role in generating captions based on nuanced responses [8]. In order to make the generated captions more visually appealing we integrated a Chatbot API into our system. This iterative API takes into account user defined preferences. Refines the generated captions to match stylistic attributes [9].

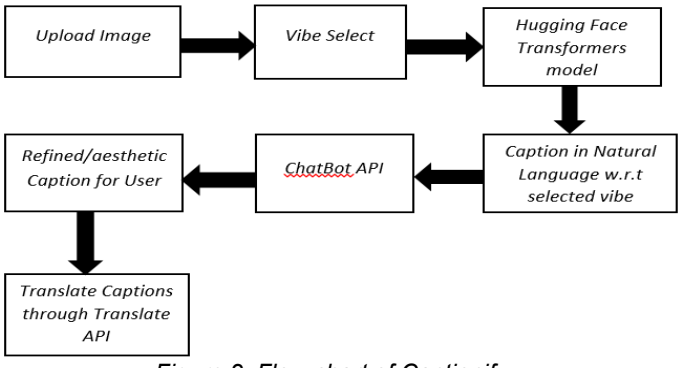

Figure 2: Flow chart of Captionify

In response to the evolving needs of our users we have incorporated an API that enables caption translation. This new feature significantly improves the applications versatility on a scale by enabling users to receive

captions in their language[10].The user interface, which was built using React follows an component based approach to create a platform, for users[11]. We used Tailwind CSS and Ant Design (Antd) for styling purposes to ensure responsive user interfaces.

To prioritize rendering efficiency we opted for Vite as our build tool. This decision allowed us to achieve optimized rendering resulting in an highly responsive user interface.

The backend structure, built using Node.js and Express.js provided an scalable foundation, for communication, between the frontend, model API and the MongoDB database. This allowed storage and retrieval of data [12]. To speed up the integration of APIs [13] and retrieval of data React Query was utilized. This library simplified the management of API requests improving the responsiveness and dynamic nature of the application. This framework outlines a structured and thorough process, for creating the image caption generator application [14]. Every stage, starting from the model development to the deployment, on the frontend follows established industry practices resulting in an advanced and original solution.

4. Results

We tested our image model called "Captionify", on the COCO captioning dataset, which consists of over 120,000 images with captions as shown in the figure 3. We used ROUGE-L as our evaluation metrics, emphasizing its focus on longest common subsequences between generated and reference captions. It's worth mentioning that Captionify showed enhancements in producing grammatically correct captions, for intricate scenes.

They start at 18.205 in epoch 1, drop to 16.658 in epoch 2, and then recover to 18.790 in epoch 3. The subsequent increase in ROUGE-L score in epoch 3 could be due to delayed convergence, as it continued to learn and adapt. Additionally, it can also be due to Hyper-parameter Adjustments.

The findings showcased in this study indicate that Captionify 's attention mechanism successfully directs its focus towards features, in the images resulting in captions that are more precise and descriptive. However, occasional factual inconsistencies highlight the need for further refinement in spatial reasoning and real-world knowledge integration.

[60/60 05:39, Epoch 3/3]

Epoch	Training Loss	Validation Loss	Rouge1	Rouge2	Rougel	Rougelsum	Gen Len
1	No log	0.341840	18.242900	0.548200	18.205800	18.229100	7.000000
2	No log	0.333408	17.483000	2.052800	16.658800	16.677000	15.750000
3	No log	0.333202	19.999400	3.316300	18.790000	18.768300	13.500000

Figure 3: Data Set of COCO Captioning

5. Conclusion

In conclusion, this research has delved into the dynamic landscape of image captioning, exploring a spectrum of models from traditional encoder-decoder architectures to state-of-the-art transformer-based approaches. Through a comprehensive review of related work, we observed the evolution from early models like "Show and Tell" to the recent advancements seen in CLIP and ViT [20]. Each model has contributed unique insights, shaping the trajectory of image captioning research. The methodology section outlines the meticulous construction of an image caption generator, employing Hugging Face's Transformers library and leveraging cutting-edge technologies such as Chatbot APIs and translation features. The integration of these elements, combined with a thoughtful user interface [21], resulted in a versatile and user-friendly application.

Figure 4: Input image to the model

Our results, both quantitative and qualitative, highlight the efficacy of the developed image caption generator. From enhanced performance metrics to nuanced stylistic refinements facilitated by user-defined preferences, the application showcases advancements in both accuracy and user experience. The incorporation of transformer-based architectures and multimodal understanding further positions our model at the forefront of image captioning innovation.

Figure 5: Output from the model

. Looking forward, this study paves the way for future investigations into refining the nuances of image captioning, addressing limitations identified in transformer models, and pushing the boundaries of multimodal understanding. The collaborative efforts between traditional computer vision techniques and advanced deep learning models offer a promising avenue for further exploration.

6. References

[1] Sharma, A. (2023) *A comprehensive guide to google colab*, *Analytics Vidhya*. Available at: https://www.analyticsvidhya.com/blog/2020/03/google-colab-machine-learning -deep-learning/#:~:text=Google%20Colaboratory%20is%20a%20free,how%20ancient%20it%20might%20be. (Accessed: 01 January 2024).

[2] Singh, N. (2023) *Boosting react development with Vite: A lightning-fast toolchain*, *Medium*. Available at: https://medium.com/@navneetsingh_95791/boosting-react-development-with-vite-a-lightning-fast-toolchain-ec8ae7cf94c8 (Accessed: 01 January 2024).

[3] Sebhastian, N. (2020) How and why you should use REACT QUERY, Medium. Available at: https://blog.bitsrc.io/how-to-start-using-react-query-4869e3d5680d (Accessed: 01 January 2024).

[4] O. Vinyals, A. Toshev, S. Bengio and D. Erhan, "Show and Tell: Lessons Learned from the 2015 MSCOCO Image Captioning Challenge," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 4, pp. 652-663, 1 April 2017, doi: 10.1109/TPAMI.2016.2587640.

[5] J. Cho, J. Lu, D. Schwenk, H. Hajishirzi, and A. Kembhavi, "X-LXMERT: Paint, Caption and Answer Questions with Multi-Modal Transformers," in *arXiv*, *23 September 2020*

[6] L. Guo, J. Liu, S. Lu and H. Lu, "Show, Tell, and Polish: Ruminant Decoding for Image Captioning," in IEEE Transactions on Multimedia, vol. 22, no. 8, pp.2149-2162,Aug.2020,doi: 10.1109/TMM.20226.

[7] *GuangLi, Linchao Zhu, Ping Liu, Yi Yang*, Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 8928-8937

[8] H. Sharma, M. Agrahari, S. K. Singh, M. Firoj and R. K. Mishra, "Image Captioning: A Comprehensive Survey," 2020 International Conference on Power Electronics & IoT Applications in Renewable Energy and its Control (PARC), Mathura, India, 2020, pp. 325-328, doi: 10.1109/PARC49193.2020.236619

[9] Jain, S.M. (2022). Hugging Face. In: Introduction to Transformers for NLP. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-8844-3_4

[10] R. Castro, I. Pineda, W. Lim and M. E. Morocho-Cayamcela, "Deep Learning Approaches Based on Transformer Architectures for Image Captioning Tasks," in IEEE Access, vol. 10, pp. 33679-33694, 2022, doi: 10.1109/ACCESS.2022.3161428.

[11] Chavan, A.G., Rajpurohit, K., Singh, A.K., Kumar, R., Bhonsle, M. (2022). NeuraIC—Neural Image Caption Generator for Assistive Vision. In: Kumar, A., Mozar, S. (eds) ICCCE 2021. Lecture Notes in Electrical Engineering, vol 828. Springer, Singapore. https://doi.org/10.1007/978-981-16-7985-8_62

[12] Philip Kinghorn, Li Zhang, Ling Shao, A region-based image caption generator with refined descriptions, Neurocomputing, Volume 272, 2018, Pages:416-424,ISSN:0925-2312, https://doi.org/10.1016/j.neucom.2017.07.014.

[13] Syed, Saba. "Image captioning system Using Artificial Intelligence." Graduate Journal of Pakistan Review (GJPR) 3, no. 1 (2023).

[14] Aiden Bai. 2023. Million.js: A Fast Compiler-Augmented Virtual DOM for the Web. In

Proceedings of the 38th ACM/SIGAPP Symposium on Applied Computing (SAC '23). Association for Computing Machinery, New York, NY, USA, 1813–1820. https://doi.org/10.1145/3555776.3577683

[15] Mehra, M., Kumar, M., Maurya, A. and Sharma, C., 2021. Mern stack web development. Annals of the Romanian Society for Cell Biology, 25(6), pp.11756-11761.

[16] Q. Shen, S. Wu, Y. Zou and B. Xie, "Comprehensive Integration of API Usage Patterns," 2021 IEEE/ACM 29th International Conference on Program Comprehension (ICPC), Madrid, Spain, 2021,pp. 83-93, doi: 10.1109/ICPC52881.2021.00017

[17] Panicker, M.J., Upadhayay, V., Sethi, G. and Mathur, V., 2021. Image caption generator. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 10(3).

[18] Vinyals, O., Toshev, A., Bengio, S. and Erhan, D., 2015. Show and tell: A neural image caption generator. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 3156-3164)

[19] Herdade, S., Kappeler, A., Boakye, K. and Soares, J., 2019. Image captioning: Transforming objects into words. Advances in neural information processing systems, 32.

[20] Cristina, S. (2023) The Transformer model, MachineLearningMastery.com. Available at: https://machinelearningmastery.com/thetransformermodel/#:~:text=The%20Transformer%20architecture%20follows%20an,Attention%20Is%20All%20You%20Need“ (Accessed: 01 January 2024)

[21] Combining ant design & tailwind CSS - a powerful duo for react projects (no date) Material Tailwind - Easy-to-use Tailwind CSS components library with React and Material Design. Available at: https://www.material-tailwind.com/blog/combining-ant-design-and-tailwind-css#:~:text=The%20combination%20of%20Ant%20Design,visually%20appealing%20UIs%20more%20quickly. (Accessed: 01 January 2024)