Movie Poster Genre Classification and Tagline Generation based on CLIP and GPT-2

Ershov Aleksandr
7025826

Jorge Calvimontes
7024517

Qiankun Zheng
7025052

Abstract

Visual-Language Learning has become very popular in the field of Computer Vision in recent years. One of the most famous models developed in this field is the Contrastive Language–Image Pre-training (CLIP) model, which is able to associate visual concepts of images with corresponding texts. In this paper, we explore the generalization power of CLIP image and text encoders to solve the multi-label classification and text generation tasks. The results show that embeddings obtained from the CLIP model preserve enough information to solve very specific tasks at sufficient level.

1. Introduction

Nowadays, CLIP [8], a multimodal model combining visual representations with natural languages supervision, has gained great popularity due to its remarkable zero-shot ability in various tasks, such as Image Generation [5, 10], Visual Question Answering [3], and Visual Entailment [13]. The idea behind CLIP model is to train the image and text encoders on a very large dataset (over 40 million queries) to get well-represented embeddings. In the original paper, CLIP was used to solve a multi-class classification task, which yielded good results just by measuring the cosine distance between images and text embeddings for each class. Given that the generated embeddings can serve as a feature map, CLIP can also be used as a preprocessing model for other tasks. So in this project, we aim to make use of the embeddings obtained from the pretrained CLIP model and study their performance.

Specifically, we proposed two models for the task of movie poster genre classification and tagline generation. In each task, we implemented a pretrained CLIP model combined with classification and text generator head for the tasks of multi-label classification and text generation respectively. The main goal of this work is to evaluate if a pretrained CLIP model can effectively be used to learn features for the tasks proposed. Additionally, we will evaluate the performance of our proposed models on a relatively small movie dataset. The dataset contains posters, genres and taglines of over 35K movies. This dataset is chosen since it is domain-specific and contains metadata that can be used to solve various tasks. For the task of multi-label classification, we train the classification head to predict the different movie genres. For the tagline generation task, we train our model to generate a tagline based on the movie posters.

2. Dataset

We use two datasets: "Movie Genre from its Poster" [1] and "The Movies Dataset" [2]. These datasets are ensembles of data collected from IMDB and Group Queries about movies released on or before July 2017. The first dataset contains direct links to movie posters, and therefore it is used to retrieve the poster image corresponding to each movie, while the second dataset provides additional metadata about the movies, such as genres and taglines. Given that both datasets have a field of IMDB code, we combine them to form our dataset which contains all the necessary metadata about the movies and the corresponding posters.

3. Methods

3.1. Genre Classification

3.1.1 Architecture

Since a movie poster may belong to one or more genres, movie poster genre classification is a multi-label classification task. In this sense, using the classification method as the original paper did is not a good approach as not all the movie posters have the same number of labels and it only considers the embeddings with the lowest distance with respect to an image. Based on these considerations, we implemented a multi-label classification head on top of the image and text embeddings obtained from the CLIP model. Specifically, we first fed movie posters to the CLIP image encoder to get the image embeddings; as we wanted our classification head to make semantic correlations be-

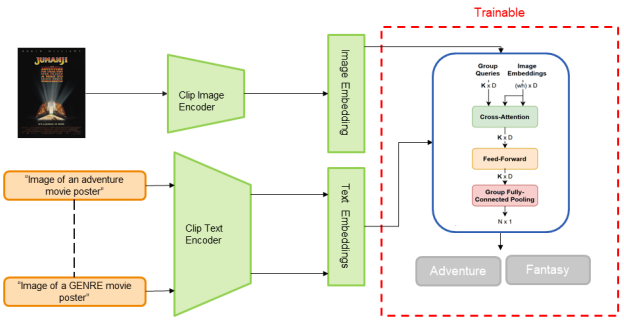


Figure 1. Model architecture for genre classification task

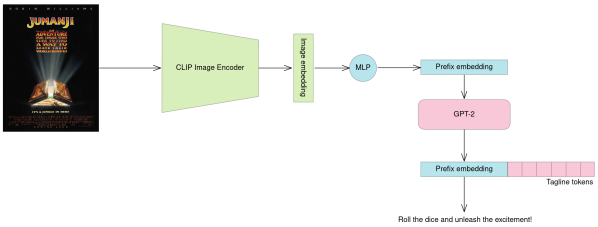


Figure 2. Model architecture for tagline generation task

tween the different genres and the image embeddings, we passed text queries corresponding to the different genres to the CLIP text encoder. For the text queries, we constructed sentences that semantically correspond to the movie genres, i.e., "This is an image of a *genre* movie poster".

Then, we used ML-Decoder model [12] as the classification head, which has good performance on multi-class and multi-label classification tasks. What attracted us to use this model was its attention-based architecture consisting of decoders, Multilayer Perception (MLP) and pooling operations on a group-based scheme. The main advantage of this model is that it is capable of receiving image and other type of embeddings and feeding those together to the decoder, whereas their contemporaries use different decoders for different types of embeddings. Given that the image and text embeddings have the same dimensionality, this classification head fits well with our task: both of the embeddings obtained from the CLIP image encoder and text encoder can be combined and fed to this model. An overview of the whole architecture is shown in Fig. 1

3.2. Tagline Generation

3.2.1 Architecture

The task of taglines generation, in contrast to the task of genres classification, requires a generative language model. We chose the GPT-2 model [9] as our backbone model, which is currently the most powerful publicly available generative language model. Due to the limited computational resources and data at hand, we used the pre-trained GPT-2 model rather than train it from scratch. The final architecture of the model used in this task is shown in Fig. 2. First, CLIP Image Encoder was used to get image embeddings for each poster. We used the most modern visual transformer ViT-B/3 [4], because it proved to be the most effective pretrained CLIP model in our first task. Then, inspired by [6], we used a simple MLP with three fully connected layers and tanh activation functions and then projected the embeddings into a new prefix embeddings space. Embeddings from this space were fed to a GPT-2 model with 12 attention modules to generated taglines. We also tried to only train the MLP module, but found that the model failed to converge. For this reason, we jointly trained the MLP module and the GPT-2 model using the cross entropy loss function and adam optimization algorithm with linear scheduler warmup.

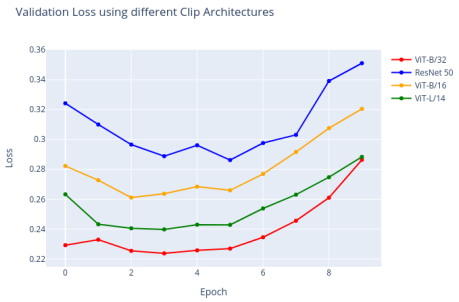
3.2.2 Evaluation

In the first place, we tried to use BLEU score [7] to assess the quality of the generated taglines, however, this score was always very low even for the reasonable generated taglines, so we recalled the definition of the BLEU score and found that this score was not very suitable for evaluating our generated taglines. Then, instead of evaluation on the word-level using the BLEU score, we measure the semantic similarities between the generated taglines and reference taglines. Specifically, we fed the generated taglines and reference taglines to another sentence-transformer pretrained on 1 billion sentence pairs dataset [11] to get the corresponding embeddings. We assumed that semantic information of sentences should be preserved in these embeddings, so we calculated the cosine similarities between the generated tagline embeddings and the reference tagline embeddings.

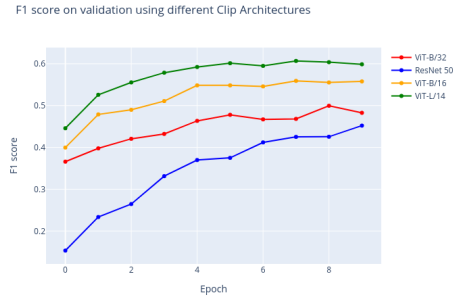
4. Results

4.1. Genre Classification

In our experiments, we first set a base configuration in which we used the standard pretrained CLIP backbone (ViT-B/32) and a classification head with the recommended configuration given by the authors, which consisted of only one layer, a MLP of 512 hidden size, a decoder of 768 size and a dropout of 0.1. On top of the base configuration, we performed further experiments with different classifier parameters and CLIP backbones. As for the loss function and the learning rate, we chose the BCE loss function and a learning rate of 0.0001 with a scheduler that reduced it by a factor of 2 when the loss stagnated. For each experiment, we trained the model for 10 epochs and kept track of the

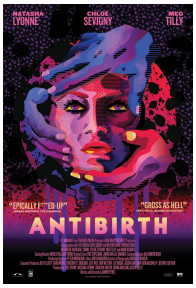


(a) Loss across epochos

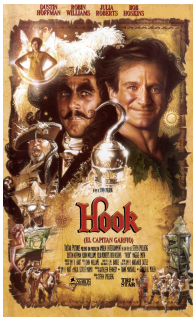


(b) F1 score across epochs

Figure 3. Clip architectures performance during validation



(a) Prediction: ['Horror', 'Science Fiction']
Target: ['Horror']



(b) Prediction: ['Adventure', 'Comedy', 'Family', 'Fantasy']
Target: ['Adventure', 'Comedy', 'Family', 'Fantasy']

Figure 4. Sampled genre predictions

	Precision	Recall	F1
Train	0.96	0.56	0.62
Validation	0.96	0.53	0.60
Test	0.92	0.50	0.56

Table 1. Best results for genre classification

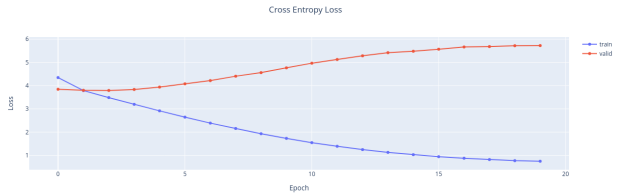
loss during training and validation phrases. We used precision, recall and F1 score to evaluate our model performance on this task.

The first thing we noticed from the base experiment was that the model was overfitting. To solve this problem, we experimented with different dropouts and found that the best dropout was 70%. Next, we experimented by changing the number of layers and the size of the MLP and decoder. Although there was some improvements with these configurations, they were not significant enough in comparison to the base configuration. For this reason, the classifier layer was kept with the same sizes as the base configuration. Keeping these parameters, we then changed the CLIP backbone. We observed the performance with different architectures as shown in Fig. 3. Finally, with the backbone that yielded the best result, we measured the precision, recall and F1 score for training, validation and test data, as presented in Tab. 1. Examples of predictions during testing can be seen in Fig. 4

Although we get a high precision, we didn't obtain satisfactory F1 score and recall. This indicates that our model is good at predicting all the true positives but does poorly on predicting the true negatives. We suspect this behaviour can be attributed to the fact that the dataset is imbalanced. For some genres, the data is very limited and our model performed poorly on these genres.

4.2. Tagline Generation

For the second task, we conducted a series of experiments to train our model. In addition to the main scenario, we also tried to train only MLP module, and use another loss function based on cosine distance of sentence-transformer embeddings, but in these experiments our model failed to converge. In the main series of experiments, we carried out the selection of hyper parameters, such as batch size and learning rate. The best parameters and final evaluation metric values for validation set are given in Tab. 2. Final test score for the best model equals 0.109 and average training time was 14 hours. Fig. 5 shows train and validation losses as well as cosine similarity scores for the best hyper parameters setup, where overfitting pattern can be observed. Despite this problem, we think the specificity of our task should be considered. The fact is that movie taglines can be characterized by their special extraordinariness, and for this reason, the generated taglines may actually be quite relevant to the content presented in the poster,



(a) Cross Entropy Loss function

(b) Cosine similarity score

Figure 5. Best model training plots.

trained modules	initial lr	batch size	score
MLP	0.000002	4	fail
MLP+GPT-2	0.00002	4	0.108
MLP+GPT-2	0.0002	4	0.105
MLP+GPT-2	0.00002	16	0.110

Table 2. Results.

but they do not exactly coincide with true tagline. Examples are shown in Fig. 6. Looking at these examples, we believe that due to the specificity of our task, the overfitting seen from the loss curve is not actually a problem that needs to be tackled.

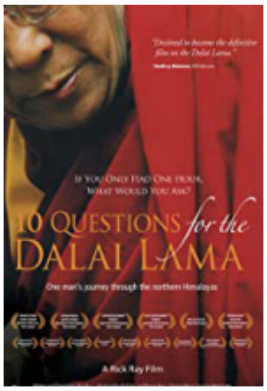
We also found that the model tends to generate identical or very close taglines for some movies, which suggests that the visual representations for these movies obtained by the CLIP model are very close in the embedding space.

5. Conclusion

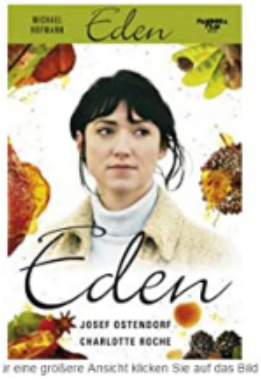
In this project, we investigated that CLIP model has sufficient generalization power to use obtained embeddings for various domain specific tasks. Despite the fact that the results we obtained were not comparable to the state-of-the-art models, we used simple neural network architectures that can be trained without using large computing resources. It can be also considered as proof of concept that large data-driven models, such as CLIP, can be used to solve a wide range of tasks. We believe that in the future, development in this direction will help with minimal effort to build models based on large CLIP-like models for specific tasks, solving them at a sufficient level.

(a) The Machines Will Rise Again

(b) Survivors will be honored



(c) One man's voice can change the lives of millions around the world



(d) Sometimes you just have to fall in love with the right person

Figure 6. Examples of movie posters and generated taglines

References

[1] Movie genre from its poster. https://www.kaggle.com/datasets/neha1703/movie-genre-from-its-poster?select=MovieGenre.csv. Accessed: 2022-06-26. 1

[2] The movies dataset. https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset/metadata?select=keywords.csv. Accessed: 2022-06-26. 1

[3] Long Chen, Yuhang Zheng, and Jun Xiao. Rethinking data augmentation for robust visual question answering. *arXiv preprint arXiv:2207.08739*, 2022. 1

[4] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale, 2020. 2

[5] Federico Galatolo., Mario Cimino., and Gigliola Vaglini. Generating images from caption and vice versa via clip-

guided generative latent space search. In *Proceedings of the International Conference on Image Processing and Vision Engineering - IMPROVE,*, pages 166–174. INSTICC, SciTePress, 2021. 1

[6] Ron Mokady, Amir Hertz, and Amit H. Bermano. Clipcap: Clip prefix for image captioning, 2021. 2

[7] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA, 2002. Association for Computational Linguistics. 2

[8] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. 1

[9] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. 2

[10] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation, 2021. 1

[11] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks, 2019. 2

[12] Tal Ridnik, Gilad Sharir, Avi Ben-Cohen, Emanuel Ben-Baruch, and Asaf Noy. Ml-decoder: Scalable and versatile classification head, 2021. 2

[13] Haoyu Song, Li Dong, Wei-Nan Zhang, Ting Liu, and Furu Wei. Clip models are few-shot learners: Empirical studies on vqa and visual entailment, 2022. 1