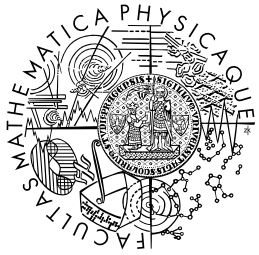


FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Shubham Shubham

Image popularity prediction

Department of Theoretical Computer Science and Mathematical Logic

Supervisor of the master thesis: Mgr. Martin Pilát, Ph.D
Study programme: Computer Science
Study branch: Artificial Intelligence

Prague 2023

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In ..... date ....

Author's signature

Title: Image popularity prediction

Author: Shubham Shubham

Department: Department of Theoretical Computer Science and Mathematical Logic

Supervisor: Mgr. Martin Pilát, Ph.D, Department of Theoretical Computer Science and Mathematical Logic

Abstract: In this thesis, we compare deep learning models for the purpose of predicting the popularity of social media posts. We curated a comprehensive dataset from a renowned social media platform, encompassing a rich variety of features including images, text captions, and social attributes. Each model's performance was evaluated based on Mean Squared Error, Mean Absolute Error, and Spearman's rank correlation coefficient. Our model, integrating convolutional neural networks for visual inputs, transformer-based models for text, and layers for social inputs, achieved a higher composite score across all evaluation metrics in contrast to the baseline model. Enhancements such as the addition of a caption network, sentiment analysis, and the removal of scaling further boosted the performance. This study illuminates the potential of deep learning in improving the precision of popularity prediction for social media posts.

Keywords: Deep Learning, Convolutional Neural Networks, Language models, Sentiment Analysis

Contents

1. Introduction

1.1 The Rise of Social Media

Technology has advanced significantly in the past few decades, leading to the rise of social media. Social media has become an integral part of over 4.74 billion people's lives around the globe. Social media's ability to connect people, to share information and experiences with ease is the reason behind it's advancement. This has allowed anyone with an internet connection to share their ideas and experiences with the world, creating new opportunities for businesses, organizations and individuals to reach and connect with others which has lead to it becoming a $94 billion industry and have had a significant role in shaping today's society. From the earliest forms of social networking platforms such as Friendster and MySpace to the modern-day-behemoths like Instagram, Reddit, Facebook they have completely transformed the way we communicate, share information and interact with each other.(Wikipedia Contributors: Social Media [2023])

Reddit is one of the most popular social media platforms today, amassing a total of 430 million monthly active users making over $423 million yearly. It was founded in 2005 by Steve Huffman, Alexis Ohanian and Aaron Swartz. Unlike other social media platforms, Reddit is not designed for users to share content with their friends or followers. Instead, its organized into thousands of subreddits, each focused on a specific topic. Users can join these subreddits to connect with others who share their interests and participate in discussions and share content. It has allowed Reddit to create a highly active user base. This unique way of content curation and community building can be attributed to making Reddit a worldwide success.(**?**)

1.2 Deep Learning

In recent decades Deep Learning (Goodfellow et al. [2016]) have revolutionized the field of Artificial Intelligence (AI). Deep Learning is a branch of machine learning that consists of various models that compose of the creation and training of multiple layers of neural networks. Deep neural networks (DNNs) are the back bone of deep learning. These models can be used to obtain relevant high level information from raw input data. They are subset of machine learning algorithms that are inspired by structure and function of human brain. They have achieved significant breakthroughs in image analysis and have been able to achieve many state-of-art results on image analysis benchmarks.

One such benchmark is Image-Net Large Scale Visual Recognition Challenge (ILSVRC) (Image-Net [2023]), this is a competition held annually from 2010 to 2012, it evaluates algorithms for object detection and image classification using large dataset of images. In 2012, a DNN architecture named AlexNet (Alom et al. [2018]) beat the previous year's winner by achieving a top-5 error rate of 15.3% which is a significant improvement over the previous year winner's top-5 error rate of 26.2%, this marked the beginning of revolution of Deep learning in image

analysis. Since then, many DNNs have been proposed and have achieved even better accuracy on Image-net benchmark. DNNs have also achieved impressive results in other image analysis tasks such as image captioning. In 2015, a DNN architecture called Show and Tell (Vinyals et al. [2015]) was proposed for the task of image captioning, which involved generating natural language descriptions of images. Its architecture was a combination of a Convolutional neural network (CNN) for image feature extraction and a Recurrent neural network (RNN) for language modeling, it achieved state-of-art performance on several benchmarks. These benchmarks show the impressive performance of DNNs in image analysis and highlight their potential for wide range of applications they have in the field of health-care, transportation, entertainment and social media.

1.3 Image Popularity Prediction

Image popularity prediction refers to predicting the popularity an image will get on a social media platform. Due to the popularity gained by social media platforms like Facebook, Instagram, Reddit it has become extremely significant to be able to predict popularity of an image on these platforms, understanding which images are likely to become more popular on these platform can help the content creators and marketers to optimize their content to gain more traction and engagement and advertisers to target the right audience, which ultimately leads to more revenue.

Deep learning techniques have proven to be particularly effective for the task of image popularity prediction. CNNs have been efficient in extracting visual features from images. These features can be extracted using a combination of convolutional and pooling layers which can then be fed to fully connected neural network to predict the popularity of the image.
Language models, such as RNNs and transformers can be used to extract textual features from captions or texts associated with the image. These features can provide more context about the image and its contents which can be used to further improve the popularity prediction of the image.

Combining both visual and textual features by employing a mutlimodal deep learning approach can further improve the accuracy of the image popularity prediction models.

1.4 Goal of the thesis

In this thesis we aim to predict the popularity of images on Reddit by employing deep learning techniques. We will create our own dataset by scraping data off Reddit. We will use convolutional neural network to extract the visual features from the image and language models to extract the textual features from the caption associated with the image. We will then combine these features to predict the score, our popularity prediction metric, of the image.
The goals of this thesis are:

G1 Create a dataset of Reddit images and metadata.
G2 Analyze the dataset to gain insights related to popularity of images.
G3 Implement deep learning techniques,including CNNs and language models, to extract image features and caption emotions.
G4 Combine the extracted features to predict the score of an image on Reddit.
G5 Evaluate the performance of the model.

By achieving the goals stated, we strive to make a contribution to the growing research on social media analysis using deep learning techniques. The results of this thesis can be further used to understand the factors that influence the popularity of images on social media platforms and provide insights for content creators and businesses.

2. Related work

In this chapter we first give a brief background on the core theoretical concepts and their working, upon which the later approaches build up.
We define Convolutional neural networks and how they can be trained to analyze and extract features of an image and how these features can be further processed to make predictions. Later, we discuss pre-existing models that utilize these concepts.

2.1 Convolutional neural network

CNNs (Goodfellow et al. [2016]) in particular are well suited for working with images as the built-in CNN layers are more efficient at reducing the higher dimensionality of images without losing information they contain, compared to other deep learning models.
The basic building block of a CNN is the convolutional layer followed by pooling layer, the convolutional layer employs the input image with a set of learnable filters. Every single filter is a small matrix (e.g. 3x3 or 5x5) of weights, these matrices convolve with the input image to produce a feature map i.e. a 2D array where the values correspond to the locations in image where that feature was detected. In small steps the filter slides over the image producing a new feature map, which shows presence of local patterns and features, at each location of the image.
The following layer in a CNN is the pooling layer, they are used for down-sampling the feature maps produced by the convolutional layer i.e. reducing the spatial dimensions of the feature maps without losing important information about the image. There are various pooling layers for CNNs but the most commonly used one is max pooling, which takes the maximum value of each patch of the feature map, it also makes the model more robust to small variations in the input image.
After several convolutional and pooling layers, the output is flattened and passed through one or more fully connected layers, which then learns to classify the image or make predictions based on the features extracted by the previous layers.

2.2 Intrinsic image popularity assessment

Intrinsic image popularity means the popularity of an image solely based on the contents of the image itself. They (Ding et al. [2019]) predict the popularity of images on Instagram, they start by creating a dataset of image-popularity discriminable pairs(PDIPs) by lowering the effects of non visuals factors. First, they propose a probabilistic method to generate PDIPS at a low cost with high accuracy. The metric they use for image popularity prediction is log scaled number of likes on the image, N, and make two assumptions:

- N follows a normal distribution with the mean μ and standard deviation (std) σ.

- The intrinsic image popularity E is a monotonically increasing function of μ.

Considering the assumptions and applying Bayes theorem they obtain :

$$P(E_X \geq E_Y | N_X, N_Y) = \Phi(\frac{N_X - N_Y}{\sqrt{2}\sigma})$$

where $\Phi(.)$ is the standard normal cumulative distribution function.

$$P(E_X \geq E_Y | N_X, N_Y)$$

implies the probability of image Y being intrinsically less popular compared to image X.
Practically, they chose a threshold,

$$P(E_X \geq E_Y | N_X, N_Y) > T$$

, that is large enough to facilitate discernible distinctions in popularity among Popularly Differentiated Image Pairs (PDIPs). However, identifying pairs of images that conform to this threshold might not fully capture the nuances of popularity since factors beyond visual characteristics could also be at play. Therefore, they have chosen to incorporate the three most impactful non-visual factors from Instagram for a more comprehensive analysis., i.e.

- **Upload time:** For the likes to stabilize, images selected are those that were posted at least a month ago.
- **User statistics:** The number of followers a user has can influence the number of likes an image receives, creating a proportional relationship.
- **Caption:** Images with trending captions and hashtags gain more exposure, potentially leading to increased popularity.

. Their dataset consists of 200 million images forming 2.5 million PDIPs which obey all the constraints mentioned.
They utilize Pairwise learning-to-rank approach, whose objective is to reduce the amount of inaccurately ordered pairs, it makes the assumption that the relative order between two instances is known (or can be inferred). They use a Siamese architecture for they learning of their model, it consists of two streams of inputs and outputs, an RGB image is the input to both the streams and the predicted intrinsic popularity score is their output. Both streams have the same model architecture and share the weights while training and testing. The predicted score difference is computed and converted into a probability by utilizing a logistic function. When the training is finished an optimal predictor, l^*, is learned from either of the two streams.
During testing for a test image A, a standard forward pass is performed in order to obtain the predicted intrinsic popularity score

$$E_A = l^*(A)$$

As their base Deep neural network(DNN) architecture they employed ReNet-50, replacing only the last layer to a fully connected layer which gives one output i.e.

the predicted intrinsic popularity score. The initial weights are inherited from models pre-trained for object recognition on ImageNet, except for the last layer that is initialized by the method of (He et al. [2015]). The parameters governing the reliability of the PDIPs generation are the threshold T and standard deviation σ and are set to 0.95 and 0.3, respectively. While both training and testing, the input to the DNN is an input image of dimensions $224 \times 224 \times 3$, which is cropped randomly after re-scaling the original image to 256 x 256 pixels. Throughout the training the cross entropy function is optimized by using Adam optimizer alongside an l_2 penalty multiplier of 10^4 and batch size of 64 and learning rate for the pre-trained DNN layers is set to 10^{-5} and for the last layer is set to 10^{-4}. A decay by the factor 0.95 is introduced to the learning rate after every epoch.

2.3 Neural Networks and Regression analysis

They (Qian et al. [2017]) predict the popularity of images of Instagram, their dataset consists of 3,411 images of different landscapes. They use landscape images to reduce the non visual bias, i.e. images of good looking women or cute animals gets more attention when compared to images of planets or galaxies, that might affect the popularity of the images. They designed a web scrapper developing on existing Instagram downloaders which are written in Python, this web scrapper uses GraphQL to directly send requests onto Instagram's web servers and then processesing the responses in JSON, eliminating the need to use Instagram API key, then it saves the metadata into a csv file. Although the scrapper was restricted by Instagram's web server to processing only a couple of hundred requests in an interval of 5-10 minutes. To further remove the bias they chose images that were most popular and had similar shape, i.e. square, and they utilized #scenery_lovers so that bias from the amount of traction images get from use of hashtags can be reduced.
The metric they use for image popularity prediction is the like-to-follower ratio, i.e. the ratio between the number of likes on the image to the number of followers the account that posted the image, this metric is used to minimize the bias in the popularity based on the number of followers different accounts have. The non-visual features they take into account are the location at which the image was captured, number of comments,the amount of time passed since the image was posted, the number of hashtags in the caption, the length of the caption, the number of followers of account posting the image, the amount of posts and activity the posting account has and height and width of the image.
For the base DNN architecture they utilize a state-of-art model, developed by Google, know as Inception-v3 (Szegedy et al. [2016b]), the architecture of the model consists of over 30 layers and multiple paths throughout, enabling it in achieving high accuracy. They keep the default weights and settings and retrain the last layer of the network.

2.4 Multimodal Deep Learning Framework

They (Abousaleh et al. [2020]) utilize 432,000 images from Flickr as a dataset and analyze the internal features (color, texture, hue count, brightness contrast,

gist, color entropy, composition geometry, background simplicity) and external features (user information, post metadata, timeline of the post). They utilize a virtual-social convolutional neural network (VSCNN) model. The architecture of the model can be divided into two parallel phases.
Phase 1 deals with the extraction of internal features, high and low level visual features, from the images. To achieve this they employ a CNN known as VGG19 (Simonyan and Zisserman [2014]), this network consists of 19 layers and has been pre-trained on 1 million images from the ImageNet dataset, followed by the integration of the extracted features which results in the output of a vector with 4,710 dimensions. This vector describes various visual features of the image, then Principal component analysis(PCA) is used to lower the dimensionality of this vector from 4,710 to 20 dimensions, giving a vector, denoted A, which constitutes of 20 most relevant features from the image and at last they normalize the values of A, so they're all contained in same scale.
Phase 2 deals with the extraction of external features, meta and non-visual features, from the metadata of the corresponding images. Using similar process from phase 1, they extract an output vector and after applying PCA it results in a vector of 14 dimensions, denoted B.
The resulting vector outputs, A and B, from both phases then act as inputs to the proposed VSCNN model in-order to make the predictions on the popularity of the corresponding posts. The architecture of the VSCNN model consists of two independent CNNs, these CNNs are applied in-order to draw out the structural and discriminating depictions of both the visual (A) and social (B) vector's features and are classified as visual network and social network, respectively. The architecture of both these networks consists of three layers each being a 1-dimensional convolutional layer, utilizing the Rectified Linear Unit(ReLu) as the activation function to each of the layers. Followed by a Fusion network which is employed to bring together the two networks, visual and social, by combining their outputs and turning these to separate networks into a unified network. The architecture of this network includes a merge layer which is followed by two fully connected layers. The merge layer functions as a way of concatenating the outputs from the visual and social network, by taking them as an input and producing outputs which act as the inputs for the following fully connected layer. The outputs from the second fully connected layer are summed up at a final node and represent the predicted popularity of the network. This predicted popularity is then cross-checked against the actual probability during supervised training and Mean Square Error (MSE) is computed, then they employ back propagation up to the final node in-order to reduce the MSE and achieve the highest accuracy. The Figure 2.1 displays the full architecture of the model.

2.5 Sentiment and Context Features

The proposed architecture (Gelli et al. [2015]) is derived from two important characteristics that contribute to the popularity of an image. First one being the visual sentiments portrayed in the image and the description of the image, i.e captions, tags etc, these features are then fed to their prediction model and the prediction score is obtained. For the popularity metric they consider the number of views an image gets on Flickr and apply the log function to the division of the

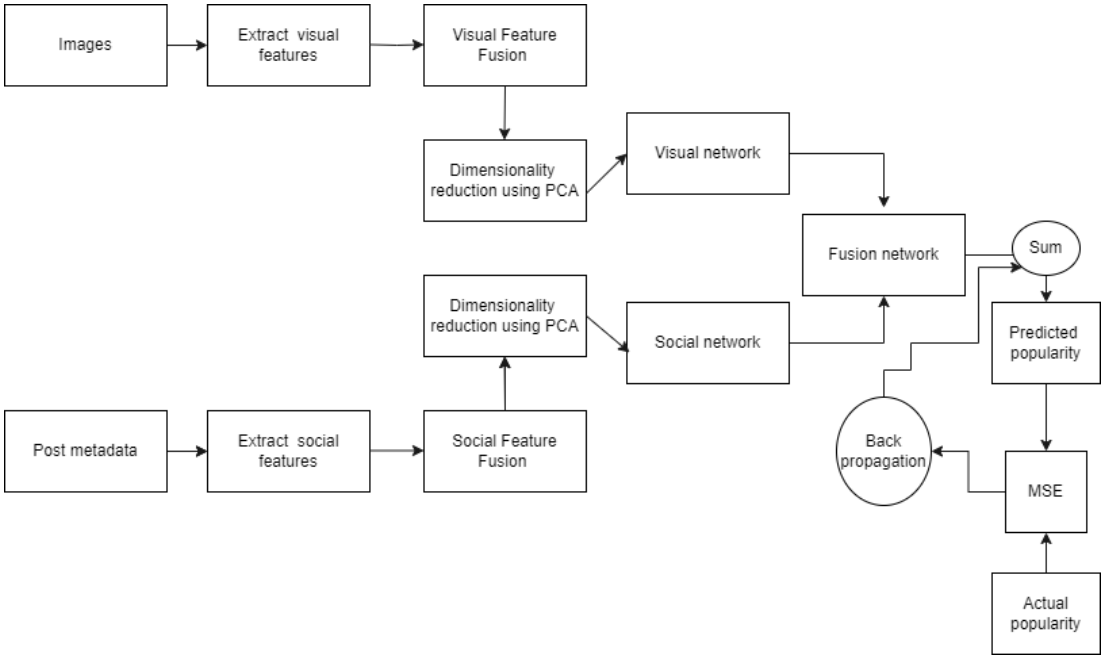


Figure 2.1: Architecture of the model

number of views with the amount of time spent between the retrieval of the image and the when the image was uploaded in order scale the popularity metric and better deal with the variations in the number of views.

For the purpose of finding out which emotion is triggered by the visuals of an image a classification known as Visual Sentiment Ontology (VSO) (Columbia University: Digital Video Multimedia Lab [2023]) is utilized. VSO is incorporated with 3,244 Adjective-Noun-Pairs (ANPs). Specifically, they used a model known as DeepSentiBank, a CNN which is finetuned for the classification of images on 2,096 ANPs, a subset of the above ontology. For each image they extracted two descriptors which they named SenANPs and FeatANPs which are extracted from the prediction layer of 2,096 dimensions and 7th fully connected layer of 4,096 dimensions. To extract the object features they use a 16 layer CNN, which for each image extracted 1,000 objects along with 4,096D representations of the 7th rectified fully connected layer. Extraction of context features is further divided into two parts. First, extraction of features from the Tags on the image for which they utilize an ontology known as Freebase. It consists of millions of topics which are interconnected. For a tag from the image, a search is performed for Freebase topics associated with that tag, and the most popular topic is then selected based on the popularity ranking of the topic in the ontology. The tag whose matches were not found in the ontology were ignored. After the retrieval of the topic a different query is performed to obtain the type and its domain. Then 100 types with most frequencies were nominated as the their knowledge base. This process is executed for each image through which they obtained a 100 dimension histogram, which they termed as TagType. Then they took the 78 pre-defined by the ontology and matched them with the tags, they counted the matches, thus obtaining a 78 dimension histogram. Second, extraction of emotions from the caption of an image. For this they utilize a CRF-based language model in order to carry out the process called Named Entity Recognition(NER). They obtained a dimension

feature through implementing a 7-class model which they termed NER_7. To further reduce the bias from the users uploading the images, they extracted user features as the mean of the number of views on the images from the user.

Entity extraction from description is performed using a well known CRF-based language model to perform Named Entity Recognition (NER). They used the pre-trained 7-class model for MUC that is able to recognize Time, Location, Organization, Person, Money, Percent, Date. They count the occurrences for each class and build a 7D feature that we term NER_7. They utilize a support-vector-machine as their model for the popularity prediction. To this model they apply L2 regularized L2 loss support vector regression from LIBLINEAR package (SVR), reason being it's scalability, in contrast to the kernelized version, over a large sparse data and very high number of instances.

3. Dataset

In this chapter, we discuss the specifics of the dataset employed for the evaluation of our deep learning model. The dataset, gathered in-house, is sourced from the globally renowned social media platform, Reddit. Initially, we provide an overview of the Reddit platform to familiarize the reader with its operational dynamics. Later, we outline the detailed procedure of data extraction from Reddit, detailing the nuances of the web scraping process. We further discuss the steps involved in data preprocessing, leading to the final dataset.

3.1 Reddit platform

Reddit, a prominent American social media platform, hosts an array of specialized forums, referred to as subreddits, that cater to a variety of interests ranging from politics and games to memes. Registered users can engage with these subreddits by subscribing to those that pique their interest and by posting content that aligns with the respective subreddit's theme. Once a post is shared on a subreddit, it becomes visible on the home timeline of all the subreddit's subscribers, though users may also directly view the post within the subreddit. Interactivity on Reddit is fostered through mechanisms like upvoting, which indicates a liking of a post, and downvoting, signifying disliking. Users can also comment on posts or share them, further enhancing the dynamics of engagement on the platform. Moreover, Reddit provides several filtering options to tailor the user's browsing experience. These categories include "Hot", "Top", "Controversial", "New", "Rising", and "Best", each serving a unique function to streamline content in accordance with the user's preference. This customization makes Reddit a versatile platform that fosters vibrant communities.(**?**)

- **Hot:** These posts have the highest number of upvotes in recent time.
- **Best:** Posts in this category have the highest difference of upvotes to downvotes, meaning the highest score. The score can be calculated as:

$$\text{score} = \text{number of upvotes} - \text{number of downvotes}$$

- **Top:** These are the posts with the highest number of upvotes regardless of the downvotes. 'Top' posts can be further categorized into subcategories such as all-time, past year, past month, past week, past 24 hours, and past hour.
- **Controversial:** These posts are typically not seen as suitable for the subreddit for various reasons and thus have a very low score.
- **Rising:** These posts are relatively new and are still gaining a lot of traction and upvotes.
- **New:** These posts are sorted by the time of their posting, regardless of their traction on the platform.

Apart from home timeline, users can also browse Reddit via popular, this timeline contains most recently trending posts around the App irrespective of the subreddits the user follows. There is also an option known as 'Discover' where Reddit suggested posts that the user may like based on their interests and interactions.

3.2 Scraping data from Reddit

Collecting data from Reddit involves several methods, each with its own set of challenges. Initially, we tried scraping data using the Reddit API with a Python script but encountered Error Code 429 ("too many requests") repeatedly. Adjustments such as setting a sleep time between requests did little to resolve this issue, resulting in inefficient collection of data.

We then explored using Pushshift API (Pushshift [2023]) with both Pushshift API Wrapper (PSAW) and Python Pushshift.io API Wrapper (PMAW), but while data collection was efficient with PSAW, it isn't actively maintained by developers, raising reliability concerns. PMAW, on the other hand, failed to provide accurate scores during scraping, making the gathered data quite unreliable.

We then considered an alternative method of using large Reddit data dumps, however, for a requirement of only 2000 data points per dataset, downloading tens of thousands of data points seemed highly inefficient.

Our optimal choice became Python Reddit API Wrapper (PRAW) with the Reddit API, despite its limit of a maximum 1,000 data points per request. We overcame this limit by accessing posts during specific time frames by using the built-in PRAW filters, such as 'Hot', 'Top', etc. 'Top' posts were further filtered by time frames such as 'All time', 'Past year', 'Past month', etc. This sequential retrieval helped us bypass the limit and achieve a sufficient number of posts.

Our final data, however, faced duplicate rows and broken URLs issues due to overlapping posts during retrieval and dead URLs respectively. To resolve this, we created functions to eliminate duplicate rows and delete rows with broken URLs. After trimming the dataset, and excluding non-image posts, we finally obtained a dataset of 2000 posts along with their metadata from two comparable subreddits: 'pics' and 'earthporn'.

The resulting dataset contained 21 features including post id, title length, author details, number of author's posts, flair text, post time, URL, post flair, original content flag, distinguished post flag, self-post flag, caption of the post, spoiler flag, adult content flag, stickied post flag, edited post flag, locked post flag, number of comments, upvote ratio, post score, and log-scaled post score. In conclusion, data collection and cleaning demanded an iterative approach to ensure the quality and relevance of the dataset for subsequent model training and evaluation.

This dataset is further processed. This initially entails downloading images locally from their URLs to prevent potential future disruptions due to 'dead' URLs. Later, we streamline our data by eliminating less informative attributes such as self-post flag, stickied post flag, edited post flag, spoiler flag, flair text, post id, and distinguished post flag. These features tend to be sparsely populated and exhibit negligible correlation with the metric for predicting popularity.

Additionally, we also transform our timestamp data, following the methodology presented in (Eryk Lewinson [2023]). As we prepare the dataset for modelling, we eliminate the URL feature, which becomes redundant once we have access to the images. We also exclude features such as comment count and upvote ratio, considering that they are indeterminable at the time of posting. Through this data preparation, we aim to provide clean and effective datasets to maximize the efficiency and performance of our predictive models.

3.3 Pics dataset

This dataset is taken from subreddit r/pics. This subreddit has 29 million users, it contains images with interesting story behind them, there is no concrete theme for the images that are posted here so it contains very diverse images from selfies of people to funny signs to paintings and sketches. The story in the caption by the author is often of more importance than the image features itself. Some examples of the images are below:

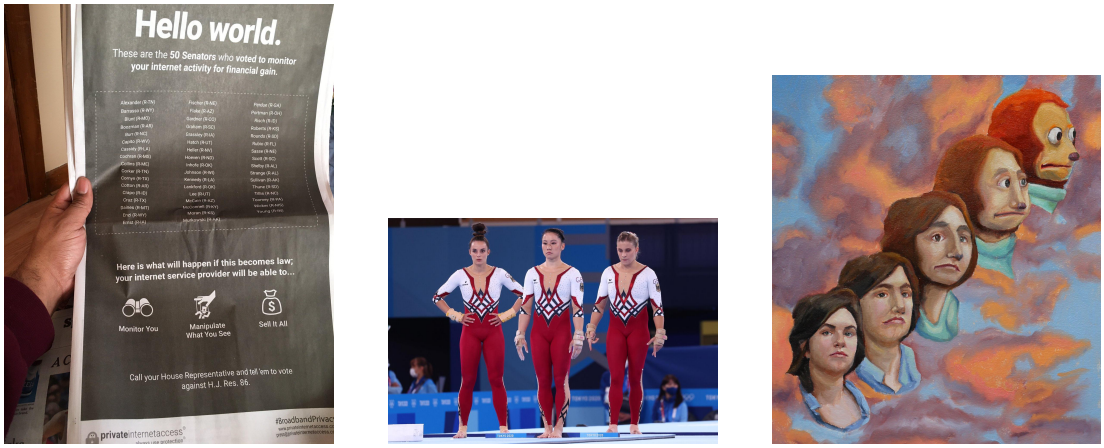


Figure 3.1: Images from pics dataset

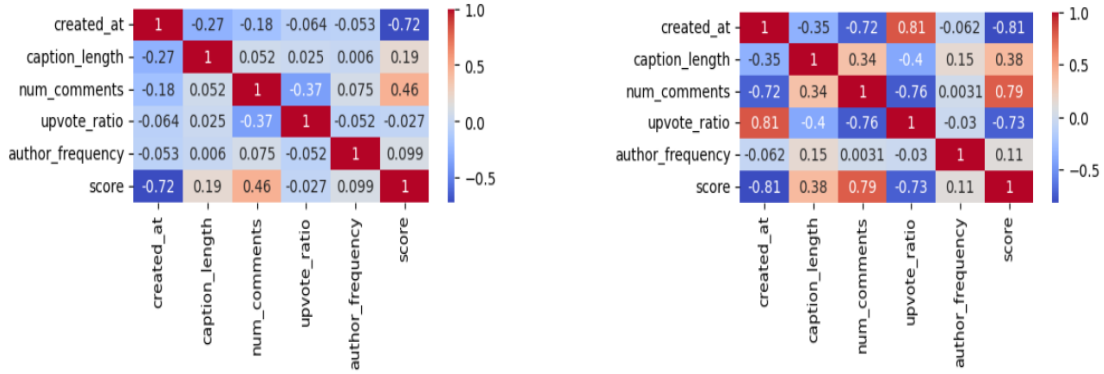
3.4 Earth dataset

This dataset is taken from subreddit r/earthporn, this subreddit has 23 million users, it contains images of nature and beautiful scenic places on the planet. The images posted are usually high quality and of professional caliber. The caption of the image is not as important as the high level features of the image itself. All the images have a concrete theme and do not diverge from it. Some examples are below.

Figure 3.2: Images from earth dataset

3.5 Analysing the dataset

We carry out some analysis on the datasets above to understand it better and extract information about some patterns it follows or characteristics it exhibits.

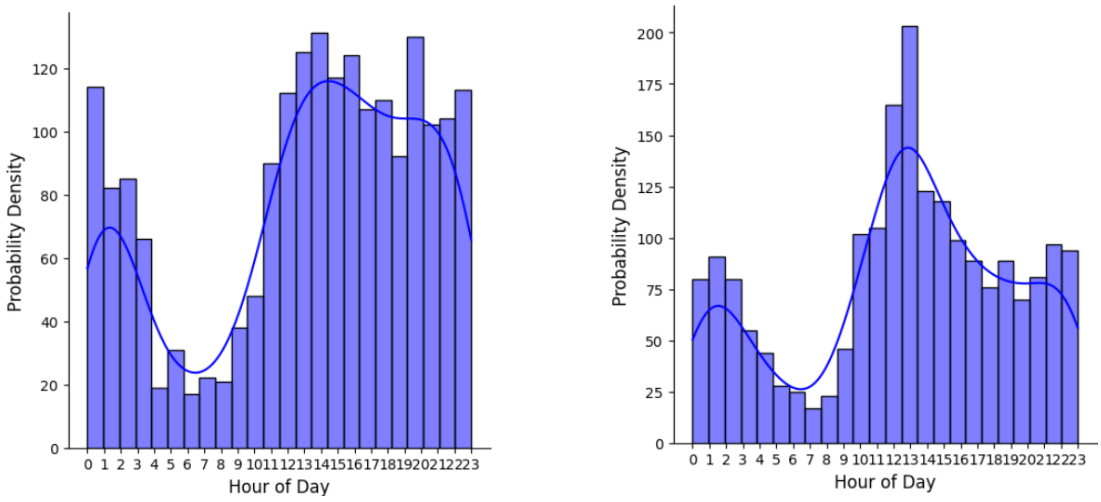


Pics dataset Earthporn dataset

Figure 3.3: Correlation matrix

We notice in Figure 3.3 that number of comments are quite correlated to the score in both datasets which is expected but it is interesting to see a negative correlation between upvote ratio and score, it indicates the upvote ratio of a post is not a significant indicator of the score a post has in pics dataset, as it happens that a post with a score of 100 with upvote ratio of 0.7 is still more popular than a post with score of 50 and upvote ratio of 0.90, but in earth dataset the negative correlation is quite significant which is unexpected. We can also notice a high negative correlation between 'created at', which constitutes the time of creation of the post, and score, meaning that posts made during a certain time during the day tend to have a higher score. We notice a positive correlation between the number of posts made by authors on the subreddit, although it is quite low in pics dataset indicating the number of posts author submits is quite insignificant for the score a post will get, but in earth dataset it does have some influence on score. We can notice positive correlation between the length of the caption provided by

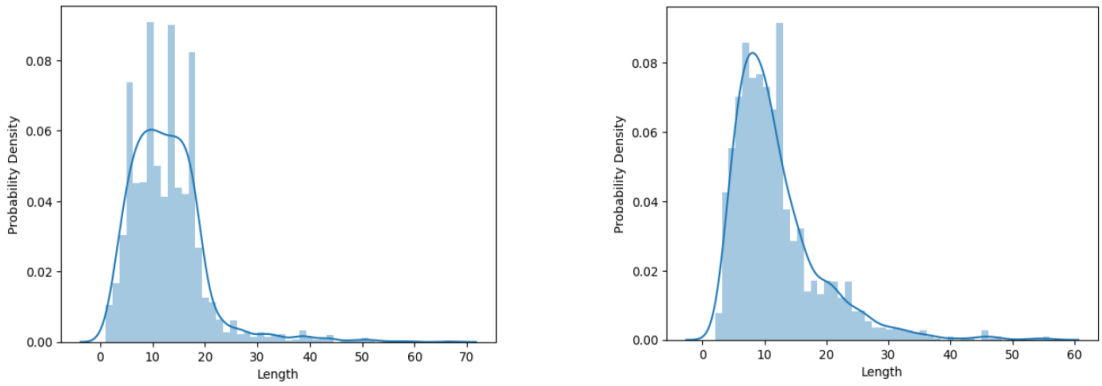
the author and the score indicating that a meaningful caption has influence on the score but it is unexpected to see the correlation being higher in earth dataset compared to pics, as we have made the assumption from observing the subreddits that in r/pics caption is significant for score.



Pics dataset Earth dataset

Figure 3.4: Frequency of post submission by hour (UTC)

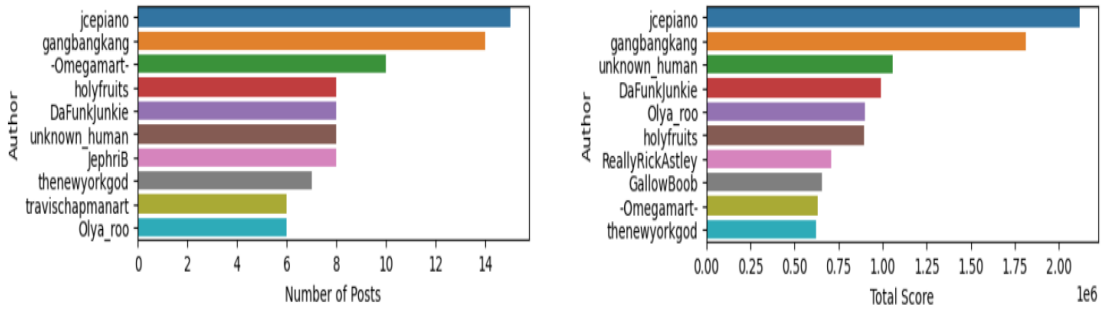
The graphs in Figure 3.4 show the number of posts made to each subreddit by the hour of the day. We can notice that Pics dataset has more frequency meaning number of posts submitted to the pics subreddit during a day is typically higher in comparison, which could possibly be due to higher number of joined users on r/pics. Also, from 04:00 to 09:00 there aren't much posts submitted to either subreddit, this indicates that there are less active users during that time the reason behind this is because during that time period it's early morning in Europe and night in America. The frequency of posts peaks during evening time, which is to be expected as a lot of people get off work and browse and relax during that time. This also provides some evidence in the support of the correlation discovered earlier, as most posts are submitted during evening time implying most active users during that time and posts during that time, based on this we can hypothesize that post submitted during this time have a better score.



Pics dataset Earth dataset

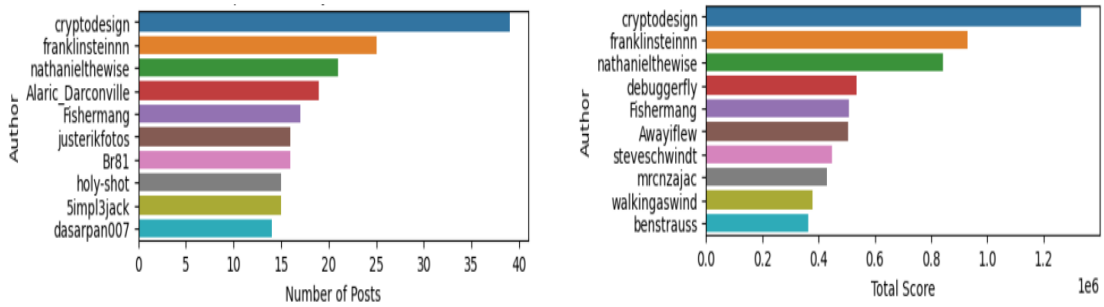
Figure 3.5: Caption length

We can conclude from the graphs in Figure 3.5, that likelihood of a caption length occurring is more uniform in Earth dataset, we can assume that more captions in Earth dataset are well formed which gives evidence for the correlation we discovered, i.e. a well formed title does have positive influence on score and also gives us evidence explaining the unexpected, as more captions are well constructed sentences in Earth dataset. This explains why the correlation of caption is slightly lower in pics.



10 Most active authors 10 most popular authors

Figure 3.6: Pics dataset



10 Most active authors 10 most popular authors

Figure 3.7: Earth dataset

We observe from figures 3.6 and 3.7, that only 2 of the top 10 authors actually show a positive correlation between frequency of posts by an author and score in

pics dataset while 4 authors show this relationship in earth dataset confirming the results of our correlation matrix. The reason behind these results are due to celebrities, like Ricky Astley and Travis Scott, posting on pics subreddit, as celebrities don't usually post that often but their post get a lot of traction compared to general public.

Pics dataset Earth dataset

Figure 3.8: Most Frequent words

The Figure 3.8, shows most frequent words used in the captions of each dataset, the font size of the of the words are indicative of their frequency. We can notice the words used in pics dataset are more diverse and commonly used, like today, year, day, friend, dad etc, while the words in earth dataset all stick to a theme of scenic high quality photography, like Mountain, USA, Fall etc.

4. Experimental Analysis and Enhancements

In this chapter, we delve into an exhaustive exploration of our experimental analysis, focusing on the enhancements we made to achieve a better performance and various experiments we performed aiming to understand these improvements more intuitively. We start by a thorough examination of the underlying baseline model, building this state of art model and then laying out its inherent limitations. This is subsequently followed by a comprehensive clarification of the improvements introduced to address and overcome these identified deficiencies. Thereafter, we shift our focus towards the empirical aspect of our study, encapsulating the experimental setup. This includes, different experiments we perform as a way to more intuitively understand our model performance. Concluding our study, we put forth a meticulous analysis and discourse on the experimental results.

4.1 Establishing the baseline model

For the purpose of this study, we propose the baseline model from Abousaleh et al. [2020], it involves the extraction and analysis of low-level features, high-level features, deep learning features all in combination defined as visual features, and social context features of an image which are later used in our model as well. Followed by feeding them to a visual network for visual features and social network for social features and in the end these networks were combined into another network defined as fusion network. We adapt this as our baseline model to predict image popularity.

4.1.1 Framework of the Baseline Model

The first step in establishing our baseline model is extracting data from our datasets. We extract a very diverse range of features, the features extracted from the image itself are defined as visual features, these features are further divided into low level features, high level features and deep learning features. We also extract features that are quite relevant in the popularity of an image, defined as social features, these features are extracted from the metadata we scraped for images.

Incorporating the visual features

Low-Level Features: We begin by extracting low-level features such as color, texture, and gist. These visual attributes are rudimentary yet critical components of any image.

- **Color:** A color histogram descriptor is used, which results in the 32-dimensional vector capturing the color distribution within an image. This method is further described in (Hassanien and Abraham [2008].)

- **Texture:** For texture analysis, we use the uniform local binary patterns (LBP) descriptor from skimage, which results in a 10-dimensional vector instead of a 59-dimensional one as described in (Abousaleh et al. [2020]). LBPs are a type of visual descriptor used in computer vision for texture classification. They work by comparing each pixel with its surrounding neighborhood of pixels. Certain binary patterns are referred to as 'uniform' because they contain at most two transitions from 0 to 1 or vice versa when traversed circularly. These patterns often signify regions of uniform texture or constant intensity in image analysis.
  - The pattern '00000000' implies all the 8 neighboring pixels have the same intensity as the center pixel. This might be seen in a clear sky in an image, where every pixel in the neighborhood is of the same color.
  - Patterns like '00011111' or '00001111' have exactly two transitions (from 0 to 1 and from 1 to 0), and they might signify an edge in an image. This could occur, for example, at the boundary of two contrasting regions in an image, like the edge of a building against the sky.
  - On the other hand, '01010101' has 8 transitions which makes it non-uniform as it indicates that the pixel intensity is changing rapidly or irregularly in the local neighborhood. This might indicate a high-frequency pattern in the image, possibly some type of noise or texture that rarely appears.
- **Scene Description:** The GIST descriptor is used to get a rough description of the scene by incorporating gradient information for different parts of an image, resulting in a 512-dimensional feature vector. This technique is described in more detail in (Oliva and Torralba [2001]).

High-Level Features: Next, we extract high level features which are more abstract and pertain to the quality and aesthetic appearance of an image, they are derived by separating the subject of an image from it's background and help in assessing the visual quality of the image.

- **Clarity Contrast:** Measures the high-frequency components in an image to determine its sharpness. This is described in more detail in (Luo and Tang [2008]).
- **Hue Count:** Analyzes an image's color simplicity and vibrancy by calculating the number of different hues present. This is explained further in (Ke et al. [2006]).
- **Brightness Contrast:** Quantifies the difference in illumination between the subject area and the background of an image, as described by (Tang et al. [2013]).
- **Color Entropy:** Differentiates natural images from drawings by calculating the entropy of RGB and LAB color space components. A 56 dimensional feature vector is extracted (28 each for the original image and down-sampled image) using the technique described in (Chen et al. [2019]) and their code from (jacob6 [2023]).

- **Composition Geometry:** Assesses the quality of a photo based on its adherence to the rule of thirds, a principle of photographic composition. This concept is explored further in (Tang et al. [2013]).
- **Background Simplicity:** Evaluates the dispersion of color in the background, with simpler backgrounds having less color variation as explained by (Luo and Tang [2008]).

Deep Learning Features: We also incorporate pre-trained deep learning model. The Visual Geometry Group-19 (VGG19) model is employed to extract high-dimensional deep features from images. It is a 19-layer deep neural network and is part of the VGG family of models. These models were among the earliest to showcase the effectiveness of depth in neural network performance. The features are extracted from the model's last layer, namely 'fc2', before the classification layers. The output of this layer is a 4,096-dimensional feature vector that serves as a robust representative of the image's deep learning features.(Abousaleh et al. [2020])

This model has proven to be highly effective in image feature extraction due to the depth of the network and the use of small and stacked convolutional filters. However, it's important to note that VGG19 is more computationally intensive and uses significantly more memory than architectures with similar performance, such as the EfficientNet and ResNet families.

Incorporating Social Context Features

The visual features are not enough to predict the popularity of an image on social media, as many user-centric features also play an important role, some examples of this are the time of day the image was posted, the user that posted the image etc. These features are divided further into user features, post metadata and time but as we are using a completely different social media platform from the one used for the baseline model, Flickr, we cannot include features such as average views, group count, member count, tag count and tagged people, simply because Reddit does not have these features. We deliberately excluded the comment count as a feature - there are two major reasons for it. Firstly, the comment count is inherently unpredictable at the time of posting an image, as it would not be a realistic or practically useful feature in a model meant to predict popularity based on data available at the time of posting. Secondly, comment count can exhibit a high correlation with an image's popularity, thus acting as a 'cheat' feature and including it might inflate the model's apparent performance while not genuinely contributing to its predictive capacity. By excluding it, we maintain the integrity of the model and ensure that the predictive power comes from genuine correlations in the data, rather than from direct indicators of popularity like the comment count. The social features we include are:

- **Caption Length**: The length of the captions that accompany the image. This might give us insight into the level of detail or complexity in the post's content.
- **Author Frequency**: This measures how frequently the author posts on the subreddit. Regular posters may have different engagement levels than infrequent ones.

- **Flair**: This feature acts as a tag and lets users know what category the post belongs to.
- **Original Content Indicator**: A boolean feature that indicates if the posted content is original or not. This may affect how users engage with the post.
- **Over 18 Indicator**: A boolean feature that tells the users if the content is for people over 18. Content with this tag may include some violence or inappropriate themes.
- **Locked Post Indicator**: A boolean feature that tells users if the post is locked, meaning no new comments or changes can be made to the post. This can have implications on user engagement.
- **Time Encoding**: The encoding of the time of the post in the form of sin and cos functions, as described in (Eryk Lewinson [2023]). The timing of a post can significantly impact its visibility and engagement.

After all the features have been collected and preprocessed for use in model, we are left with an 8 dimensional vector.

Feature integration and model training

Once we have all the visual and social features preprocessed and ready for the model. We use Principal component analysis (PCA) (Jolliffe and Cadima [2016]) to reduce the dimensionality of the visual features from 4,711 to 20 and then we feed it to the visual network where it goes through a series of Conv1D layers, each layer is followed by a dropout layer for regularization and batch normalization layer for stabilizing the learning process and then it ends with a flatten layer to convert the multi-dimensional tensor into a 1D tensor. We then feed the social features to the social network, which consists of a sequence of Conv1D layers, mirroring the structure of the visual network, and then we concatenate these values and feed them to a fusion network. The fusion network comprises of two dense layers, each followed by dropout for regularization. It works as a way to blend the information from both visual and social aspects of the data.

We scale the target values i.e. scores by taking a log of the score and then normalizing it using Minmax scalar before feeding it to the networks for the training. For training we make a 80-10-10, train-validation-test split in our dataset. Then model is trained by employing the Adam optimizer along with a learning rate scheduler callback which decreases the learning rate by a factor of 0.1 after every 10 epochs. The model's performance is evaluated on a validation set during training, and the weights that yield the minimum validation loss were saved by employing a model checkpoint callback. The model was trained for a total of 50 epochs with a batch size of 20. The 4.1 describes the model architecture of the baseline model described in (Abousaleh et al. [2020]). In both the visual and social networks, every Conv1D layer is paired with a Dropout layer and a Batch Normalization layer. The Dropout layer, set with a parameter of 0.1, serves as a regularization technique to prevent overfitting by randomly setting a fraction

of the input units to 0 during training. The Batch Normalization layer, on the other hand, normalizes the activations of the layer at each batch by adjusting and scaling the activations. This aids in improving the speed, performance, and stability of the neural network. The fusion network, the first Dense layer, namely FC1, is followed by the Dropout layer set to 0.1 and the second Dense layer, namely FC2, is followed by a dropout layer set to 0.2, this is followed by the Dense layer, namely output layer, with 1 neuron.

Network	Layer	Kernel	Activation Function	Number of Neurons
Visual	Conv1D 1	3	ReLU	32
	Conv1Dv2	3	ReLU	64
	Conv1Dv3	3	ReLU	128
Social	Conv1D 1	2	ReLU	32
	Conv1DS2	2	ReLU	64
	Conv1DS3	2	ReLU	128
-	Merged Layer	-	-	4736
Fusion	FC1	-	ReLU	1024
	FC2	-	ReLU	500
-	Output Layer	-	-	1

Table 4.1: Detailed Layers of the Model Architecture

Evaluation and Limitations of the Baseline model

The Multimodal presented exhibits a unique approach as it utilizes both visual and social data in predicting the popularity of online images. The evaluation metrics used, namely Mean Squared Error (MSE), Mean Absolute Error (MAE), and Spearman's rank correlation coefficient also known as Spearman's rho, give a comprehensive idea of the model's performance.

The model's ability to separately process the visual content and the social context of each post and then merging the results together allows it to capture a broader spectrum of features that can be potentially influential in predicting the popularity of images on social media platform, as in the context of social media, the popularity can be influenced by a variety of factors ranging from the visual appeal of the post to the social influence of the poster. It also provides insights such as, the popularity of an image is closely related to the popularity of the user uploading it, and the trends that are going on in the world for example the image with highest upvotes in both datasets is uploaded by Rick Astley, a famous singer and songwriter, from the set of "Never gonna give you up", the song was a major hit in the 80s and also started trend of "Rick Rolling" which lasts still to this day on social media specially Reddit. This aligns with our intuitive understanding of social media, where followers of a popular user are more likely to interact with their posts, leading to higher popularity.

Even though there are some promising aspects to the baseline model, it is not without its limitations. Firstly, this model assumes that social and visual features can be processed independently before merging. This may not always be the case as there might be complex interactions between these two types of features that the model fails to capture. For example, a visually appealing image posted by a less popular user may receive fewer interactions than the same image posted by a more popular user.

Secondly, the baseline model does not consider a major part of any images posted online which is the captions that accompany them, these captions can play a very important role in the popularity of the image so not considering them might handicap the baseline model's capability to get precise predictions.

Lastly, the baseline model's reliance on the availability of both visual and social data might limit its applicability. In scenarios where one type of data is missing or insufficient, the model may not perform as well. Furthermore, the model's performance can be influenced by the quality of the social data, which can vary significantly across different social media platforms and user demographics.

4.2 Enhancements to the model

4.2.1 Pretrained Models

In an effort to enhance the visual feature extraction process, we choose to explore a variety of other pretrained models. These include models from EfficientNet family, namely EfficientNetB0 and EfficientNetB3, also models based on Inception architecture, namely Inceptionv3 and InceptionResNetv2. Finally model from Residual Network family, namely, ResNet50. These models, unlike VGG19 which was used in the baseline model, consists of different architectural optimizations and advancements in the field of deep learning, offering a potentially wider and more nuanced extraction of the visual features of the image.

EfficientNet B0 and B3

EfficientNet (Tan and Le [2019]) is a family of models that are scaled versions of a base model. EfficientNetB0, the base model, was developed through a systematic approach called compound scaling - optimizing the depth, width, and resolution of the network together. The idea is that while increasing the depth of a network generally improves performance but there comes a point after which additional layers offer diminishing returns and can even hurt performance. The same holds true for increasing the width and resolution. Thus, compound scaling offers a balanced approach to scaling all three dimensions together. EfficientNetB3 is a larger variant of the base model, scaled using the compound coefficient determined through a grid search on the base model. The B3 version is significantly larger than B0 and can extract more detailed features from the images, potentially leading to better performance.

Inceptionv3 and InceptionResNetv2

The Inceptionv3 model from (Szegedy et al. [2016b]), based on the Inception architecture, is known for its efficiency in terms of computational resources. It introduces modules of varying filter sizes in the same layer, which allow for multi-level feature extraction from the images, capturing different types of information. These features can range from simple edges to complex shapes. The InceptionResNetv2 model from (Szegedy et al. [2016a]) is an advanced model that combines the benefits of the Inception architecture with residual connections. The use of residual connections helps alleviate the vanishing gradient problem, making it easier to train deep networks.

ResNet50

ResNet50 (He et al. [2016]), a 50-layer deep network it is part of the Residual Network (ResNet) family that was introduced to handle the vanishing gradient problem. The central innovation in ResNet is the use of "skip connections" or "shortcuts" that allow the gradient to be directly back-propagated to earlier layers. This allows the model to learn identity functions and ensures that the additional layers do not degrade the performance of the network. This model can learn more complex features and offer improved performance over shallower networks.

4.2.2 Sentiment Analysis of Captions

Understanding that social context plays a crucial role in the popularity of an image. We decided to extend our social features by incorporating sentiment analysis of the captions associated with the images. This is rooted in the understanding that user's reactions to an image can be profoundly influenced by the emotional connotations carried by the accompanying captions. Thus, quantifying these sentiments can significantly bolster our model's predictive abilities.

To accomplish this, we employ two sentiment analysis tools: the Valence Aware Dictionary for Sentiment Reasoning (VADER) lexicon and Flair. Both of these tools are renowned for their effectiveness in measuring the sentiments expressed in text data of captions, but each of them bring unique capabilities to our analysis.

VADER

VADER (Hutto and Gilbert [2014]) is specifically designed for social media text. It considers both the polarity (positive vs negative) and the intensity of emotion in a given text. VADER is lexicon and rule-based, which means it uses a list of lexical features (i.e. words, emojis) associated with sentiment scores and combines them based on a set of predefined rules or heuristics. One of the main strengths of VADER is its understanding of the context, as it can handle intensifiers, such as "very", and diminishers, such as 'not', as well as understand when 'but' is used to change the sentiment of a statement.

Flair

Flair (Akbik et al. [2019]) is a powerful and flexible framework for state-of-the-art Natural Language Processing (NLP). Unlike VADER, Flair uses a combination of bidirectional LSTM (Long Short-Term Memory) models for contextual string embeddings to understand the sentiment of a sentence. This means that it understands the order and the semantic meaning of words, thereby providing it with a broader contextual understanding of the sentence. In doing so, it can provide more accurate sentiment analysis results when the context is important for understanding the sentiment of a sentence.

We utilized VADER to generate an initial sentiment score for each caption, effectively transforming the qualitative text data into a quantifiable numeric representation of the underlying sentiment. We also employed Flair for its ability to analyze sentiment in a context-dependent manner, which can potentially capture the more nuanced emotional tone in a caption that could escape VADER's analysis.

This integration of rule-based and contextually-aware sentiment analysis tools positions our model to more accurately predict the popularity of images shared on social media platforms.

4.2.3 Language Models

We extend our analysis by experimenting with the use of Language Models (LMs) on the captions. While the sentiment score provides an understanding of the overall emotional tone of the caption, an LM could provide a more nuanced understanding of the text content, potentially leading to improved prediction accuracy. We experimented with several state-of-the-art LMs, including BERT (Bidirectional Encoder Representations from Transformers), GPT-2 (Generative Pretrained Transformer 2), and XLNet. Each of these models has their own unique architecture and capabilities.

BERT

BERT (Devlin et al. [2019]) is developed by Google, it is transformer-based and designed for NLP. It is pre-trained using a large collection of text data, then fine-tuned for specific tasks. The bidirectionality of this model is what sets it apart from the rest. Models that came before like OpenAI's GPT were trained in a unidirectional manner meaning either from left-to-right or right-to-left but BERT, is trained to understand the context of each word based on all of its surrounding words, to both its left and right, leading to a deeper understanding of the language.

GPT-2

GPT-2 (Radford et al. [2019]) is developed by OpenAI, it is a large-scale unsupervised language model that excels in tasks that require generating long sequences of text. Its scale and performance have established it as a leading model in the NLP space. GPT-2 models the probability of a word given all the previous words in a sentence, and is thus a left-to-right unidirectional model. It uses the transformer model's attention mechanism to weigh the importance of words in a sentence.

XLNet

XLNet (Yang et al. [2019]) is a generalized autoregressive pretraining method that allows learning bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order. Unlike BERT, which corrupts the input by replacing some tokens with masks, XLNet keeps all the tokens and predicts each of them in all possible permutations. This approach overcomes the limitations of both BERT's masked language modeling and traditional autoregressive methods.

4.2.4 Modified Model Architecture

The basis of enhancements to our model are grounded on the hypothesis that visual, social, and linguistic (captions) aspects have individual and combined significance in predicting the popularity of a post on social media. Therefore,

the model could benefit from understanding these independent predictions before integrating them. To incorporate the linguistic aspect, we introduced a new network, which we define as the "captions network". This network makes use of different language models to extracts their embeddings on the captions and then feeds them into the captions network. The output from the visual, captions, and social networks are then fed to the fusion network, creating a more informed basis for final predictions.

Independent predictions

By allowing each network to independently predict the outcome, the model takes into account the unique predictive power of each data source. For instance, the visual network may find patterns in the images that are strongly correlated with the score, whereas the social network may discover that certain social features that hold the predictive power and and the captions network might identify sentiment or cues that impact traction gained by the post. This approach respects the individuality of each data source and attempts to learn the unique patterns present in each of them.

Merging results

Once the individual predictions are made, they are merged into a single vector that will be fed into the fusion network. This merging step is key, as it allows the model to learn how the predictions made from the visual, social and linguistic data interact with each other. Some patterns may only emerge when considering all the data sources together. For instance, an image's aesthetic quality might significantly contribute to the final popularity only if it's combined with certain linguistic or social aspects.

Captions network

The captions network plays a pivotal role in our model by processing the textual information contained in the post captions. Captions may provide context to the image that carry meaningful information that can influence user engagement. For this network, we leverage several language models, including BERT, GPT-2, and XLNet, extracting their embeddings as a representation of the captions. This network learns to predict the popularity score based on the semantics and sentiment encoded in the captions.

Fusion network

The final network layer, the fusion network, is fed with the combined data from merging layer. It learns to take these individual predictions and further refine them into a final, integrated prediction. This layer's main function is to handle the higher-level interactions between the visual and social network's outputs, taking into account the possible dependencies between them.
This architecture mirrors our own intuitive understanding of how these different types of information might contribute to the target variable; they each hold valuable insights on their own, but their predictive power is likely to be highest when they're considered together.

Network	Layer	Kernel	Activation Function	Number of Neurons
Visual	Conv1D 1	3	ReLU	128
	Conv1D 2	3	ReLU	128
	Conv1D 3	3	ReLU	64
	Conv1D 4	3	ReLU	64
	Conv1D 5	3	ReLU	32
	Dense	-	Linear	1
Social	Conv1D 1	2	ReLU	128
	Conv1D 2	2	ReLU	128
	Conv1D 3	2	ReLU	64
	Conv1D 4	2	ReLU	32
	Dense	-	Linear	1
Caption	Conv1D 1	3	ReLU	128
	Conv1D 2	3	ReLU	128
	Conv1D 3	3	ReLU	128
	Conv1D 4	3	ReLU	64
	Conv1D 5	3	ReLU	64
	Dense	-	Linear	1
Fusion	Merged Layer	-	-	3
	Dense 1	-	ReLU	32
	Dense 2	-	ReLU	16
	Dense 3	-	ReLU	8
	Output	-	-	1

Table 4.2: Detailed Layers of the Model Architecture

Model Optimization and Tuning

We removed the scaling of the target value completely and it improved the performance of the model, as evident from Table 4.3. We add Early stopping with a patience of 10 to avoid overfitting as we noticed the training loss was considerably lower than validation loss.

		MSE	MAE	Spearman's rho
Baseline Earth	Scaled	241,008,867.22	11,765.44	0.483
	Unscaled	187,071,900.46	11,028.77	0.524
Baseline Pics	Scaled	3,693,914,277.58	46,687.55	0.223
	Unscaled	2,558,580,110.58	42,699.86	0.404

Table 4.3: Comparison of MSE, MAE, and Spearman's rho for Scaled and Unscaled Baseline Models ('Pics' and 'Earth').

4.3 Model Experimentation and Baseline Comparison

4.3.1 Setup

Hardware

Operating system - Windows 11 Home 64-bit
Processor - Intel i7-9750H
GPU - NVIDIA GeForce RTX 2070
RAM - 16 GB

Software

IDE - Visual Studio, Code - editor - Visual studio code
Language - Python 3.9.2
Tensorflow gpu - 2.9.0
CUDA - 11.8, CuDnn - 8.6

All the experiments are run three times and the best performance is taken, this is to reduce the randomness in model performance.

4.3.2 Experiments

We have extracted features from five different Imagenet models(vgg19, EfficientNetB0, EfficientNEtB3, ResNEt50, Inceptionv3, IncetionResNetv2) and combined these deep learning features with the low level and high level visual features mentioned in the previous section. We have also extracted embeddings from last layer before the classification layers of three different language models(BERT, GPT-2, XLNet). The various combinations of these features have led to the construction of 18 distinct models, each presenting a unique blend of high-level and low-level visual features, along with language model embeddings.

Each model has been trained independently and evaluated, allowing us to study the efficacy of different combinations of visual and language model features in predicting our target value. The various combinations of these features not only test the performance of the individual ImageNet and language models but also examine the potential of interactions that might result from the fusion of different feature sets.

The performance of these features models is then thoroughly evaluated and compared, forming a comprehensive study of the interplay between different types of image and text-based features in our prediction task. The models with the most promising results are further compared against the baseline model and some other regression models.

The following Table 4.4 and Table 4.5 show the performance of these different combinations of models on Pics and Earth datasets respectively.

ImageNet Model	Language Model	MSE	MAE	Spearman's Rho
VGG-19	XLNet	1895550501.77	35142.13	0.570
EfficientNetB0	XLNet	1870303297.745	34502.675	0.567
EfficientNetB3	XLNet	1954050566.855	35979.835	0.548
InceptionV3	XLNet	2034789256.455	37472.135	0.518
InceptionResNetV2	XLNet	1927947048.48	36370.01	0.532
ResNet50	XLNet	1988879540.86	36175.89	0.526
VGG-19	GPT-2	2299886242.705	39827.865	0.436
EfficientNetB0	GPT-2	2305111900.645	39284.615	0.418
EfficientNetB3	GPT-2	2482492402.795	41062.995	0.331
InceptionV3	GPT-2	2490070048.13	41357.52	0.321
InceptionResNetV2	GPT-2	2335512467.705	40371.675	0.367
ResNet50	GPT-2	2453295671.795	41070.245	0.350
VGG-19	BERT	2223135356.065	38916.875	0.449
EfficientNetB0	BERT	2356174648.16	39566.02	0.401
EfficientNetB3	BERT	2599499174.67	41328.7	0.318
InceptionV3	BERT	2530238035.04	41216.15	0.339
InceptionResNetV2	BERT	2355004858.365	39825.225	0.391
ResNet50	BERT	2354279647.645	39809.085	0.419

Table 4.4: Performance of different models on Pics dataset

ImageNet Model	Language Model	MSE	MAE	Spearman's Rho
VGG-19	XLNet	196221382.57	11325.56	0.474
EfficientNetB0	XLNet	202608016.765	11560.985	0.460
EfficientNetB3	XLNet	196819957.37	11353.8	0.474
InceptionV3	XLNet	184643776.915	11154.725	0.513
InceptionResNetV2	XLNet	201088429.485	11335.195	0.478
ResNet50	XLNet	193458682.01	11237.24	0.483
VGG-19	GPT-2	177914669.395	10812.625	0.525
EfficientNetB0	GPT-2	182213023.25	11152.09	0.513
EfficientNetB3	GPT-2	172718770.655	10986.875	0.561
InceptionV3	GPT-2	173348683.11	10961.15	0.558
InceptionResNetV2	GPT-2	182195662.395	11049.965	0.535
ResNet50	GPT-2	176231450.81	10774.11	0.522
VGG-19	BERT	183983294.76	10825.27	0.542
EfficientNetB0	BERT	183086740.57	10693.26	0.549
EfficientNetB3	BERT	178200376.19	10726.78	0.554
InceptionV3	BERT	179863771.295	10770.325	0.556
InceptionResNetV2	BERT	190602553.72	10844.46	0.534
ResNet50	BERT	173759530.37	10616.72	0.573

Table 4.5: Performance of different models on Earth dataset

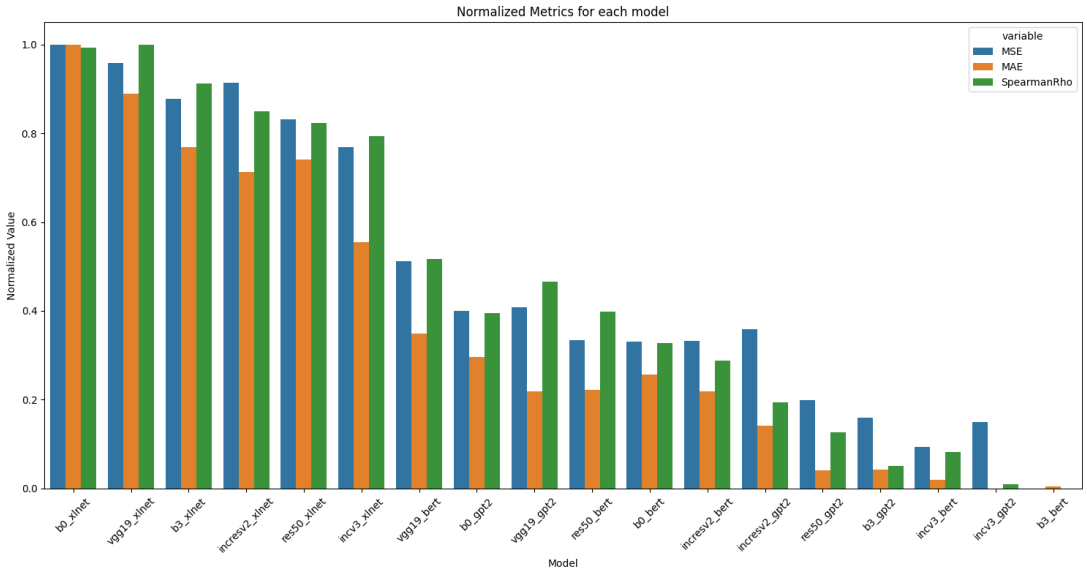
Model	combined_metric
res50_bert	2.965179
b3_bert	2.529647
b3_gpt2	2.495525
incv3_gpt2	2.472614
incv3_bert	2.444620
b0_bert	2.354177
res50_gpt2	2.266074
vgg19_gpt2	2.187756
vgg19_bert	2.121681
incresv2_gpt2	1.887227
incresv2_bert	1.813440
b0_gpt2	1.586815
incv3_xlnet	1.495747
res50_xlnet	0.855633
vgg19_xlnet	0.589969
b3_xlnet	0.537625
incresv2_xlnet	0.447538
b0_xlnet	0.000000

Table 4.6: Performance on Earth Dataset

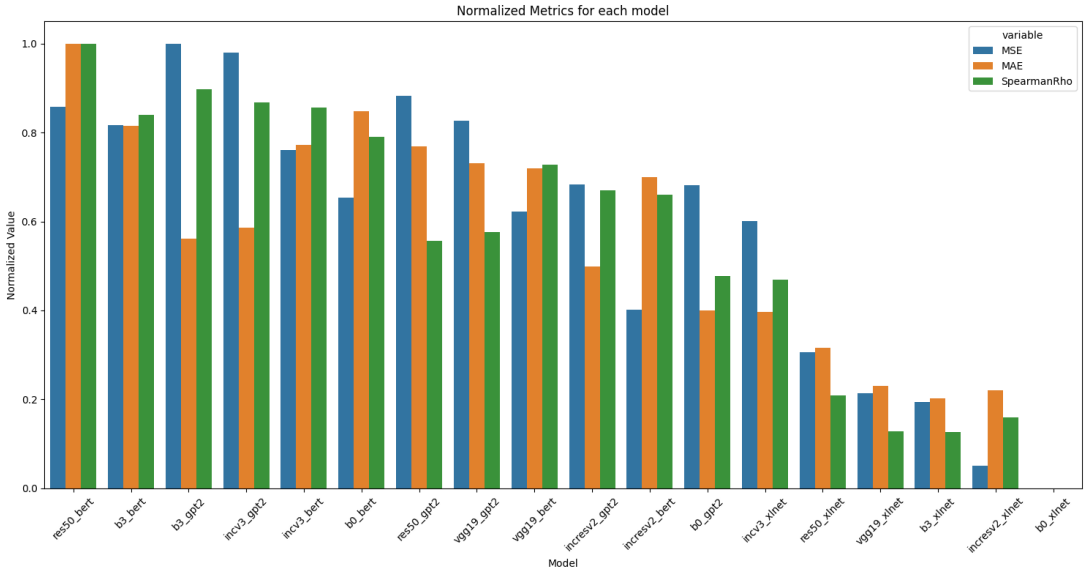
Model	combined_metric
b0_xlnet	2.986179
vgg19_xlnet	2.872092
b3_xlnet	2.581080
incresv2_xlnet	2.497465
res50_xlnet	2.416583
incv3_xlnet	2.135000
vgg19_bert	1.389237
b0_gpt2	1.100939
vgg19_gpt2	1.099266
res50_bert	0.960056
b0_bert	0.922213
incresv2_bert	0.846475
incresv2_gpt2	0.699710
res50_gpt2	0.369659
b3_gpt2	0.254877
incv3_bert	0.198291
incv3_gpt2	0.159965
b3_bert	0.004204

Table 4.7: Performance on Pics Dataset

The above Table 4.6 and Table 4.7 show how well each model did one the datasets, based on the combined ranking of mse, mae and spearman's rho. Each metric was scaled between 0 and 1 during the calculations of the combined ranking.



(a) Pics dataset



(b) Earth dataset

Figure 4.1: Comparison of Normalized Performance Metrics for Each Model

The Figure 4.1 consists of two sub figures Figure 4.1a and Figure 4.1b they illustrate the normalized performance metrics, meaning the values for the metrics MSE, MAE, and Spearman's Rho. To acquire the plots, the maximum value of MSE and MAE is subtracted from each value to "invert" the metrics. So, a lower original value (which was better) will now have a higher transformed value (which is considered better in the transformed system), then all metrics are normalized to bring them on the same scale. So, the transformed MSE and MAE values are now comparable to the Spearman's Rho values. These figures provide a comparative insight into how well each model performed in predicting popularity on different datasets. By comparing the heights of the bars for each metric, we can determine which models had the lowest error and the highest correlation, indicating better performance.

As we can observe, the model that performs the best on pics dataset is a combination of EfficientNetB0 and XLNet and the model that performs best on earth dataset is a combination of ResNet50 and BERT.

4.3.3 Experimental setup for analysis and comparison of models

Our analysis employs multiple graphical and numerical methodologies to evaluate the performance and properties of our predictive models. We focus on understanding the model's overall predictive performance, its precision in various circumstances, and the nature of errors that it commits.

Scatter Plot

This plot is fundamental to understanding the overall predictive accuracy of our model. Each point on this scatter plot represents an observation, with the X-axis displaying the model's predicted score and the Y-axis displaying the actual score. The black line indicates a perfect prediction. The closeness of points to this line reflects the accuracy of the model meaning the closer the points are to the line, the better the model's predictions. The spread of points around the line also provides insights into variability in the model's prediction accuracy, for example if the points are largely concentrated above the line it indicates that model underestimates.

Histogram of Prediction Errors

This histogram reveals the distribution of prediction errors, i.e., the difference between the actual scores and the predicted scores. The average prediction error is marked with a dashed line. This graph can provide a more precise insight into whether the model tends to overestimate or underestimate the score.

Least Accurate Predictions

We select the top three least accurate predictions. We define accuracy in this context as the absolute difference between the actual and predicted score. Inspecting least accurate predictions can give us an understanding of where the model performs worst and what might be the reason behind it.

Accuracy per Score Difference Bin

We divide the score differences between pairs of images into several bins and calculate the accuracy for each bin. This allows us to understand how well our model predicts scores for various ranges of score differences. Moreover, the ratio of each bin's accuracy to the first bin's accuracy provides insights into the relative performance of the model for different score difference levels. Such an analysis can provide essential insights for improving the model. If the model performs poorly for certain score difference ranges, it might be beneficial to investigate why this is the case and consider changes to the model or the training data to improve performance for these ranges.

In combination, these visualizations and metrics provide a detailed view of our model's predictive performance, revealing its strengths, weaknesses, and potential areas for improvement. They allow us to assess the model not only in terms of overall accuracy but also in terms of how this accuracy varies across different score ranges and prediction errors.

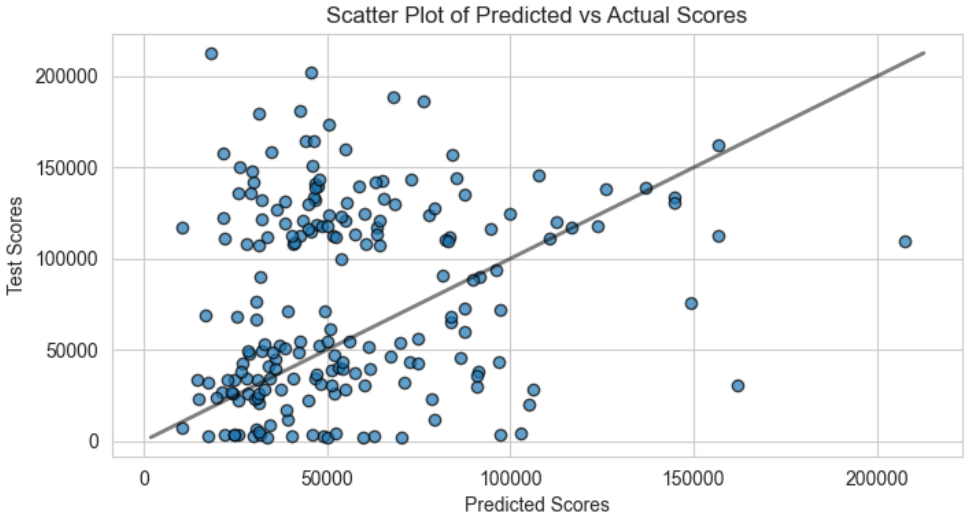
4.3.4 Baseline model analysis

We run the baseline model on pics and earth dataset individually and observe its performance.

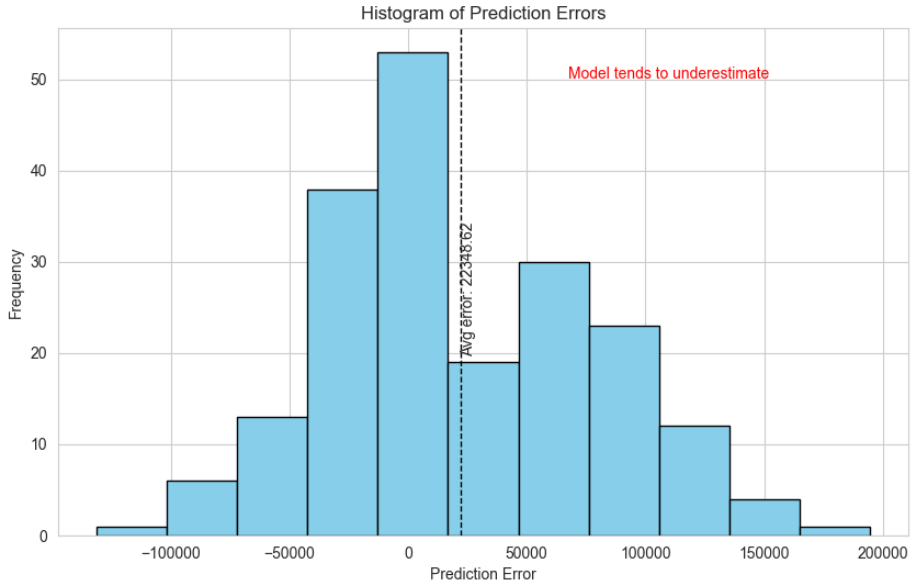
On Pics Dataset

The range of scores (difference between maximum and minimum scores) in the dataset is a substantial 437,069, highlighting a vast disparity between the most and least popular post. Evaluating the baseline model, we get the MSE of 3,693,914,277.58, this high MSE can largely be due the outliers as MSE tends to punish them, and MAE of 46687.55 which is around 10.69% of the range which suggest a decent performance but also indicates a significant room for improvement. The model also displays a weak increasing relationship between predicted and actual scores with a Spearman's rho of 0.223. This could be due to missing crucial features, the model's struggle to effectively manage the wide range of scores and the wide range of scores might be adding a complexity that the model is struggling to capture.

We can observe from the scatter plot in 4.2a of 4.2, most of the points seems to be above the line, this would indicate that our model is underestimating this is further confirmed by the plot in 4.2b. In the scatter plot we notice that a lot of points are predicted far from the ideal line for scores above 100,000 this could indicate the model simply being too simple to capture the complexities of the dataset.



(a) Scatter plot



(b) Histogram of Prediction errors

Figure 4.2: Plots

Figure 4.3: Least accurately predicted images

The Figure 4.3 shows the three images which were predicted most inaccurately, we can see that the model have underestimated the prediction in all three cases this can be highly because of models incapability to understand contexts and trends around the world.

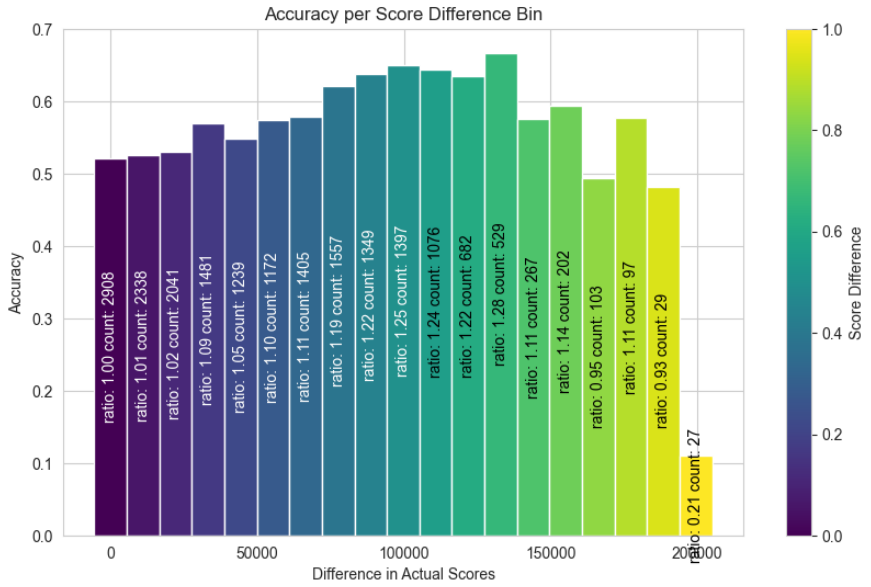
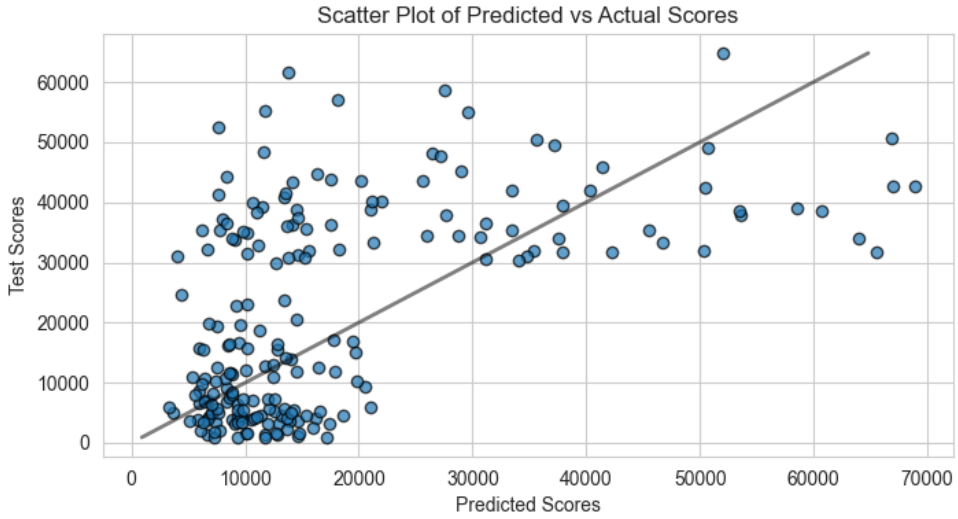


Figure 4.4: Accuracy per score difference bin

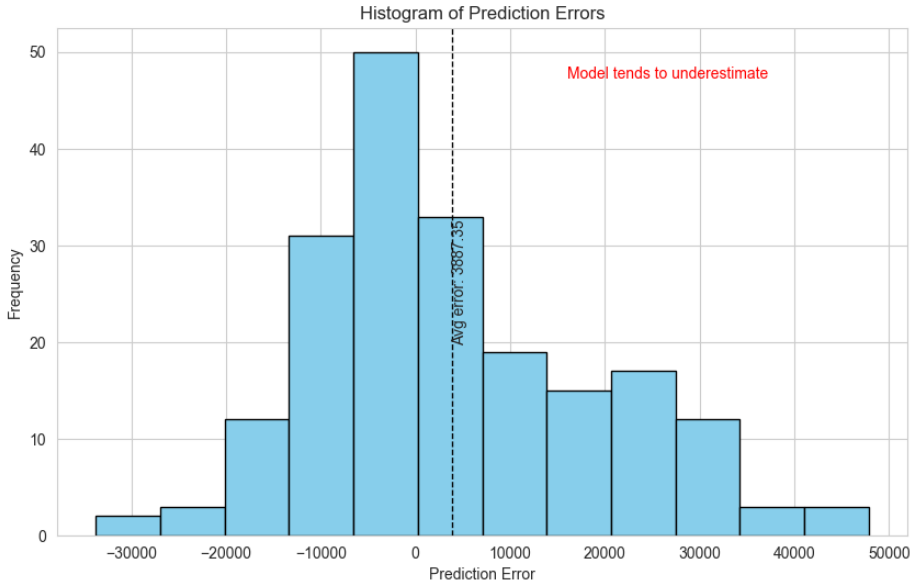
In 4.4 we notice that the accuracy increases upto a certain difference in score values but then it decreases, this indicates the models incapability for handling outliers, it further confirms that the model does not have enough complexity and it might struggle to capture the more complex patterns associated with larger score differences. Although this decrease can also be associated to the low number of score pairs for a high score difference range bins.

Earth dataset

The range of scores in the earth dataset is a substantial 103,533. The model's MSE is at 241,008,867.22, indicating substantial deviation in some predictions as MSE is sensitive to larger errors. This is underscored by the MAE at 11,765.44 which is 11.37% of the score range, a considerably lower value, signifying that on an average, the model's predictions are relatively proximate to the actual values. This becomes evident in the Spearman's rho value for the model, which is at a modest 0.478. As a rank correlation measure, this points to a moderate positive correlation between predicted and actual scores, indicating the model's limited predictive capability in correctly ranking the images based on their scores. In summary, the model, despite an average prediction error significantly smaller than the range of scores, exhibits substantial scope for improvement, both in reducing prediction errors and enhancing its ranking ability.



(a) Scatter plot



(b) Histogram of Prediction errors

Figure 4.5: Plots

Similar to the plots in 4.2 for the Pics dataset, we can see the scatter plot in 4.5a alot of points above the line indicating underestimation which is further confirmed by 4.5b. In the scatter plot we also notice that alot of points are underestimated for test scores above 30,000 this could indicate the model simply being too simple to capture the complexities of the dataset.

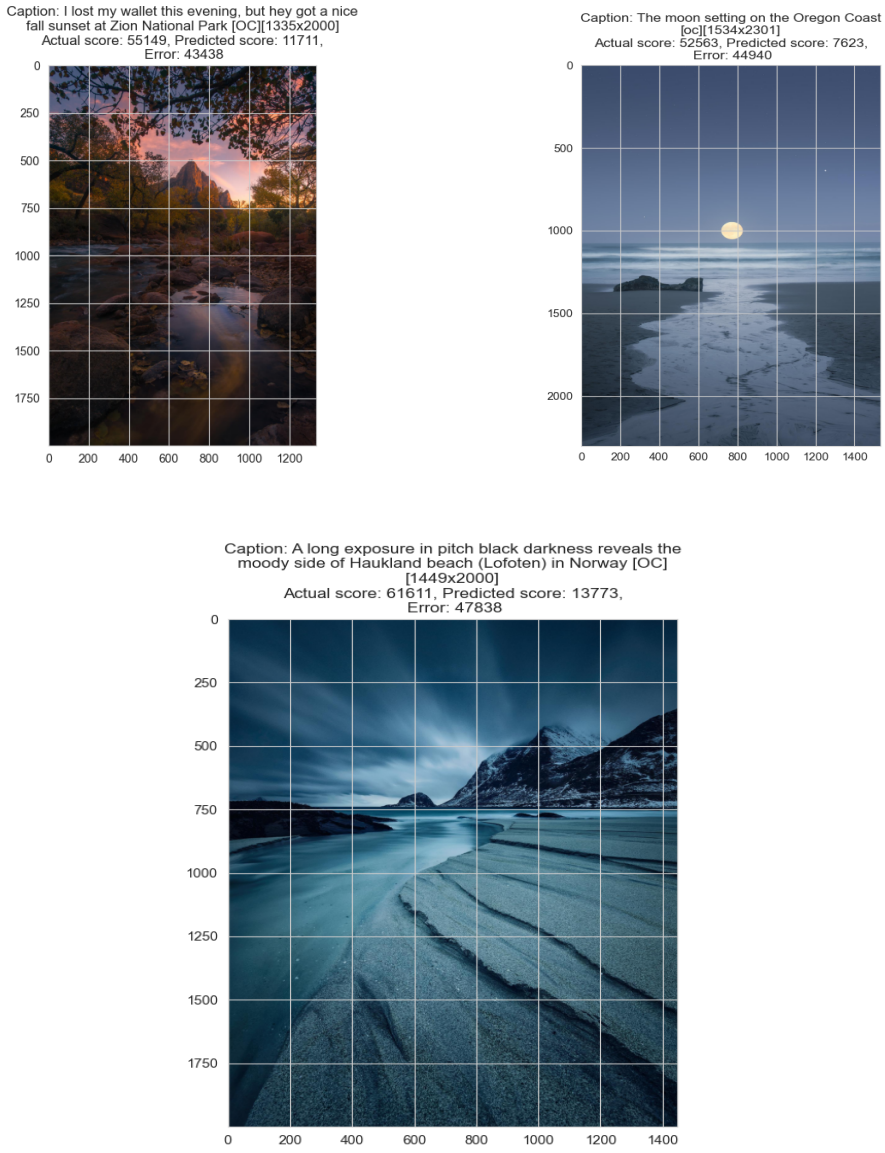


Figure 4.6: Least accurately predicted images

The images in 4.6 indicate the models lack of understanding of the context as it does not consider the captions as a feature, and after looking at the captions we can intuitively say that these associated captions were probably the reason of the high score these images have.

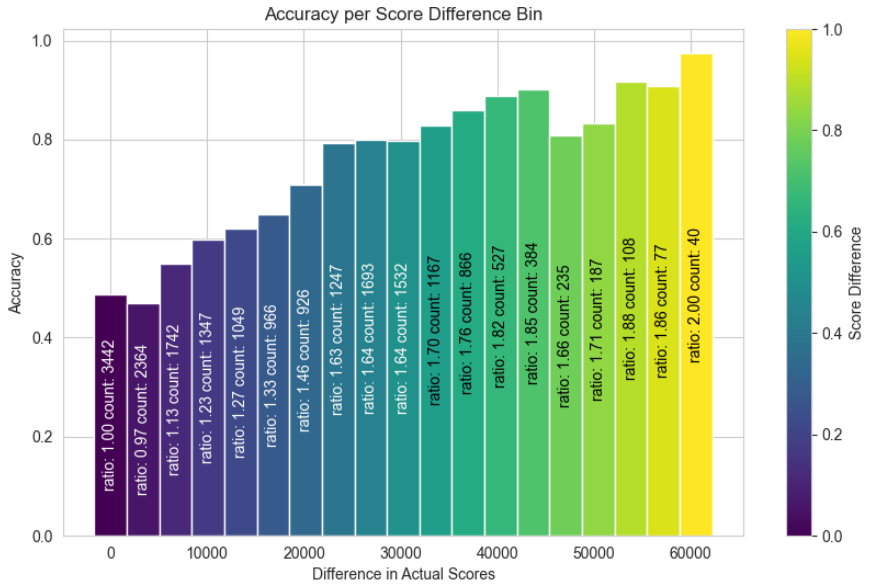


Figure 4.7: Accuracy per score difference bin

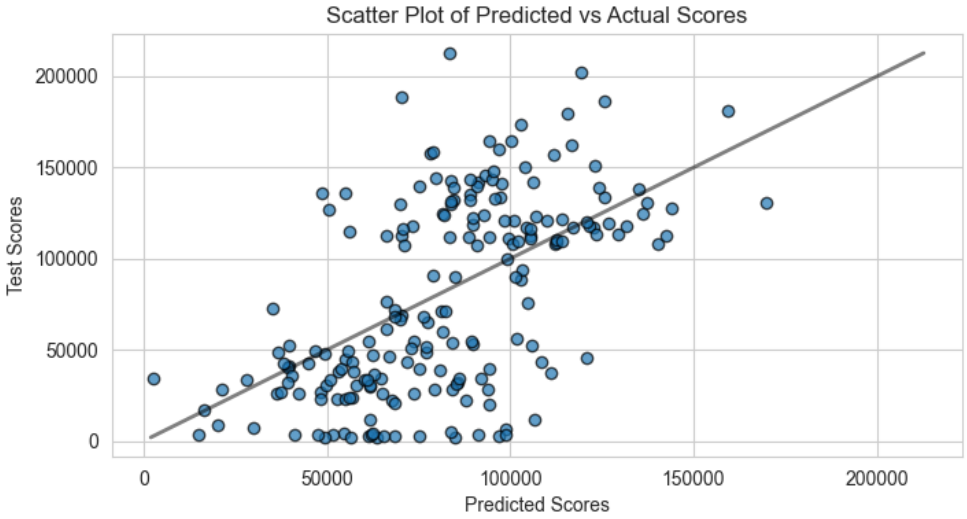
In 4.7 we notice the accuracy of the model increases with increase in the score difference this is intuitive and expected.

4.3.5 Analysis of best model combination

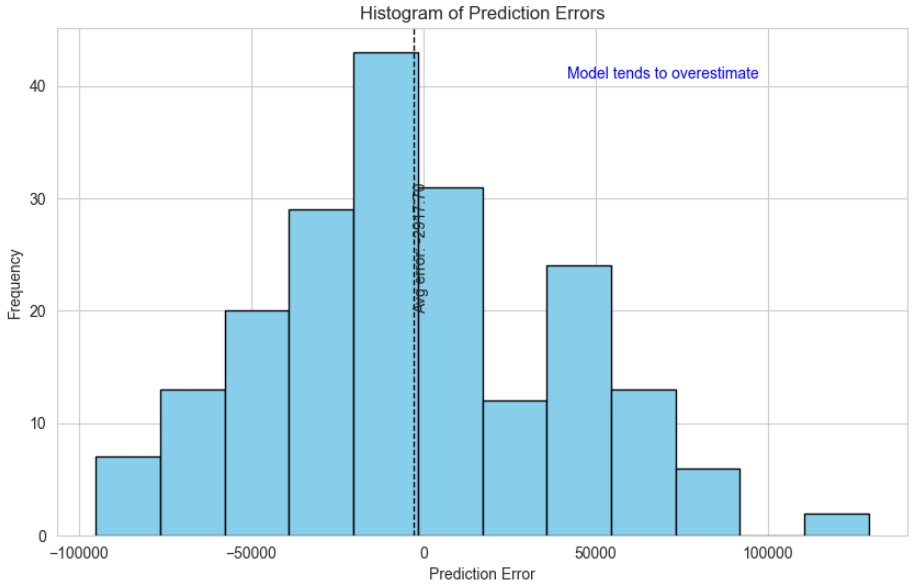
The combination of EfficientNetB0 and XLNet performed the best on Pics dataset and combination of ResNet50 and BERT performed the best on Earth dataset, so we analyse these models on the associated datasets and then compare them to the baseline model.

EfficientNetB0 and XLNet on Pics dataset

The model's performance on a dataset with a score range of 437,068 reveals nuanced insights. With an MSE of 1,870,303,297.745 and an MAE of 34502.675, the model, on average, presents reasonably accurate predictions - the MAE accounting for approximately 7.89% of the total score range. However, the large MSE value indicates the model's struggle with outlier predictions, possibly due to large errors. Meanwhile, the Spearman's rho of 0.567 reveals a moderate positive correlation between the model's predictions and actual scores. This demonstrates that the model generally maintains the relative ordering of images similar to the actual ranking, making it potentially useful for tasks emphasizing relative ordering over exact score prediction. Therefore, while the model exhibits decent performance on average, its handling of outliers or specific challenging cases could be improved.



(a) Scatter plot



(b) Histogram of Prediction errors

Figure 4.8: Plots

In 4.8 we can see all the point distributed equally on both sides of the scatter plot on 4.8a meaning we need further investigation to understand if the model overestimates or underestimates, we find that the model slightly overestimates in 4.8b. We also notice that histogram error prediction skips a bin on the right side, this may suggest that there are fewer instances where the error falls within the range of the skipped bin and then a resurgence of errors within the range of the final bin but as there are only two samples in that bin so it is hard to make any conclusions.

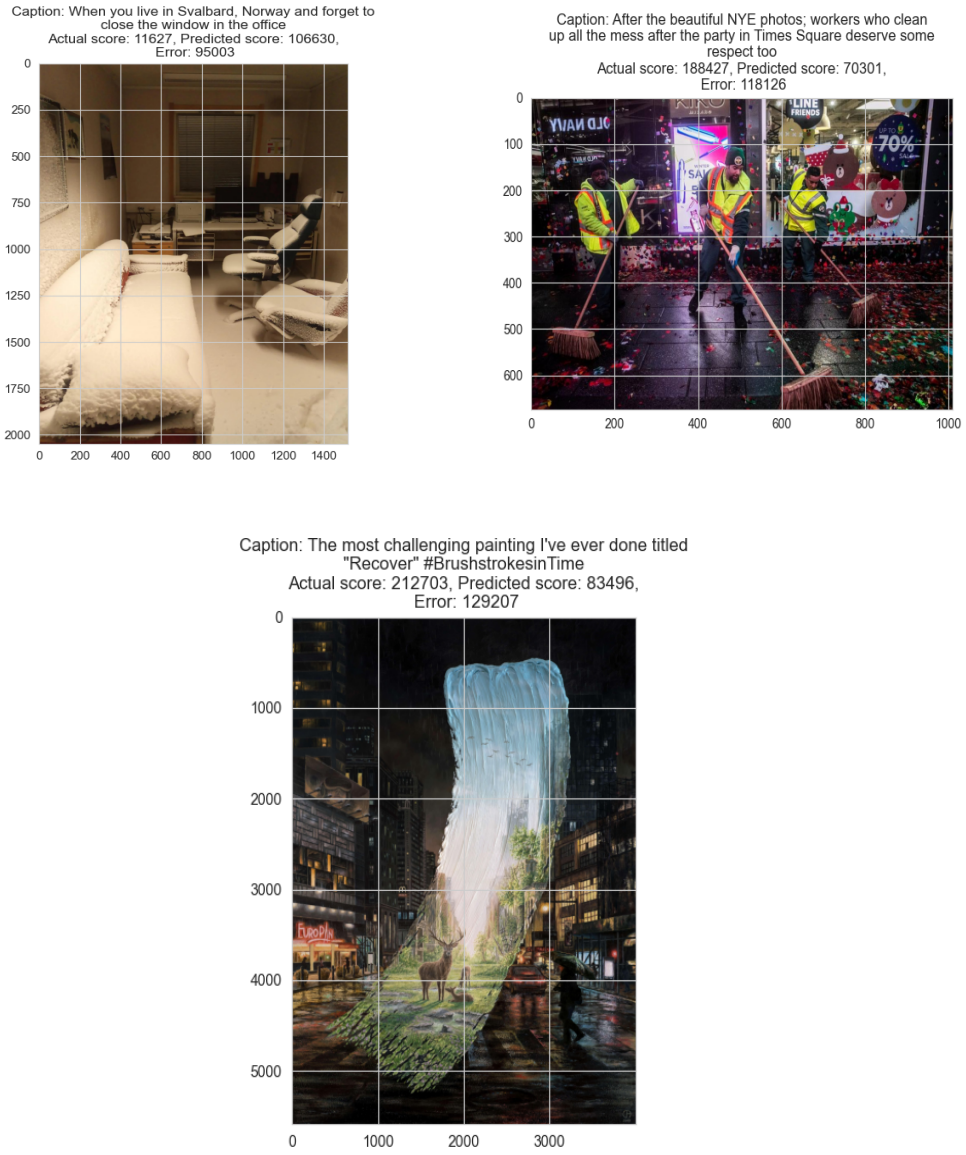


Figure 4.9: Least accurately predicted images

We notice in 4.9 that our model's least accurate prediction is the same as the least accurate prediction of the baseline model. This is largely due to the model's incapability to capture the emotion the painting ignites in humans, which is understandable.

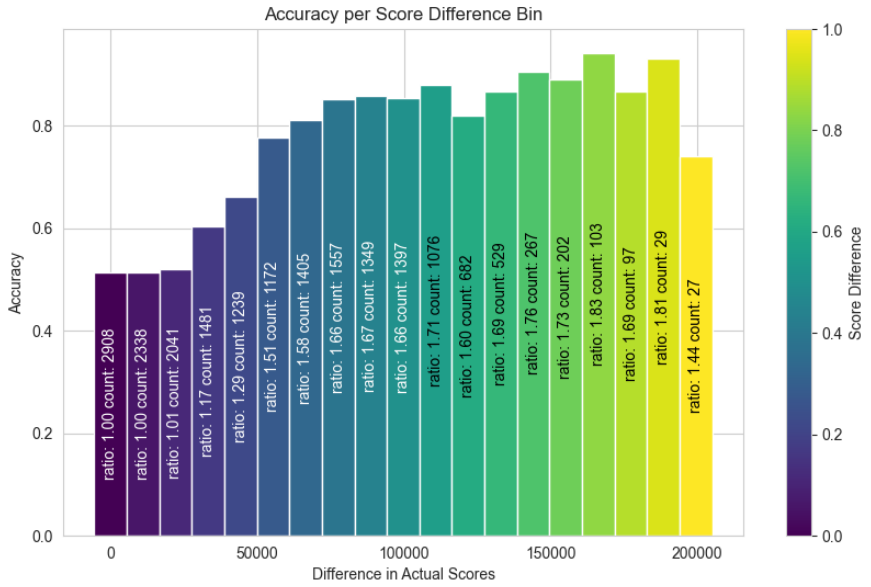


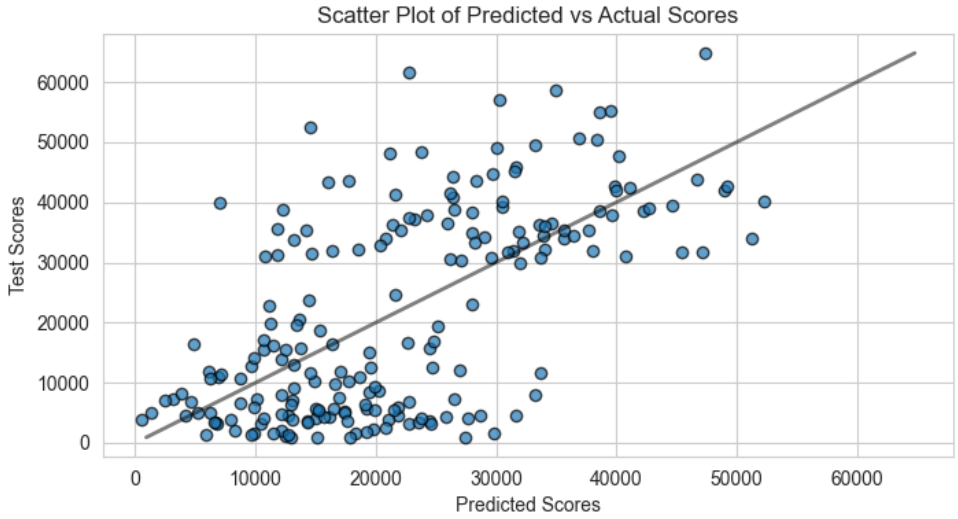
Figure 4.10: Accuracy per score difference bin

In 4.10 the model is less accurate when the difference between the predicted and actual scores is at its greatest this may mean that larger score difference might be associated with more complex or nuanced features that our current model isn't sophisticated enough to capture but since the number of sample pairs in the last bin is low we cannot be certain that is the case.

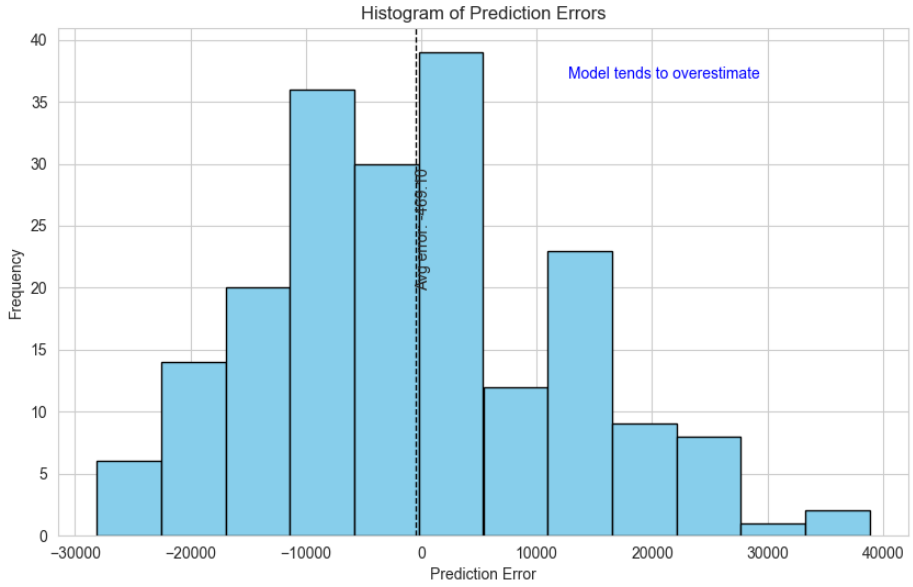
ResNet50 and Bert on Earth dataset

The model performance on a dataset with a score range of 103,533 exhibits an MAE of 10,616.72, constituting about 10.23% of the total range, and a MSE of 173,759,530.37, indicating some substantial error instances. However, the Spearman's rho value of 0.573 signifies a moderate positive correlation, suggesting that while the model may not predict the exact score, it generally retains the correct order of rankings. Despite decent performance, improvement areas exist, particularly around larger errors for enhanced accuracy.

The sub-figures of 4.11, the scatter plot in 4.11a show more points above the line meaning the model is underestimating but the histogram in 4.11b suggests otherwise this is because the model tends to underestimate more severely for high-value targets and overestimate slightly for lower-value targets leading to the scatter plot showing a pattern of underestimation due to the larger magnitude of errors for high-value targets.



(a) Scatter plot



(b) Histogram of Prediction errors

Figure 4.11: Plots

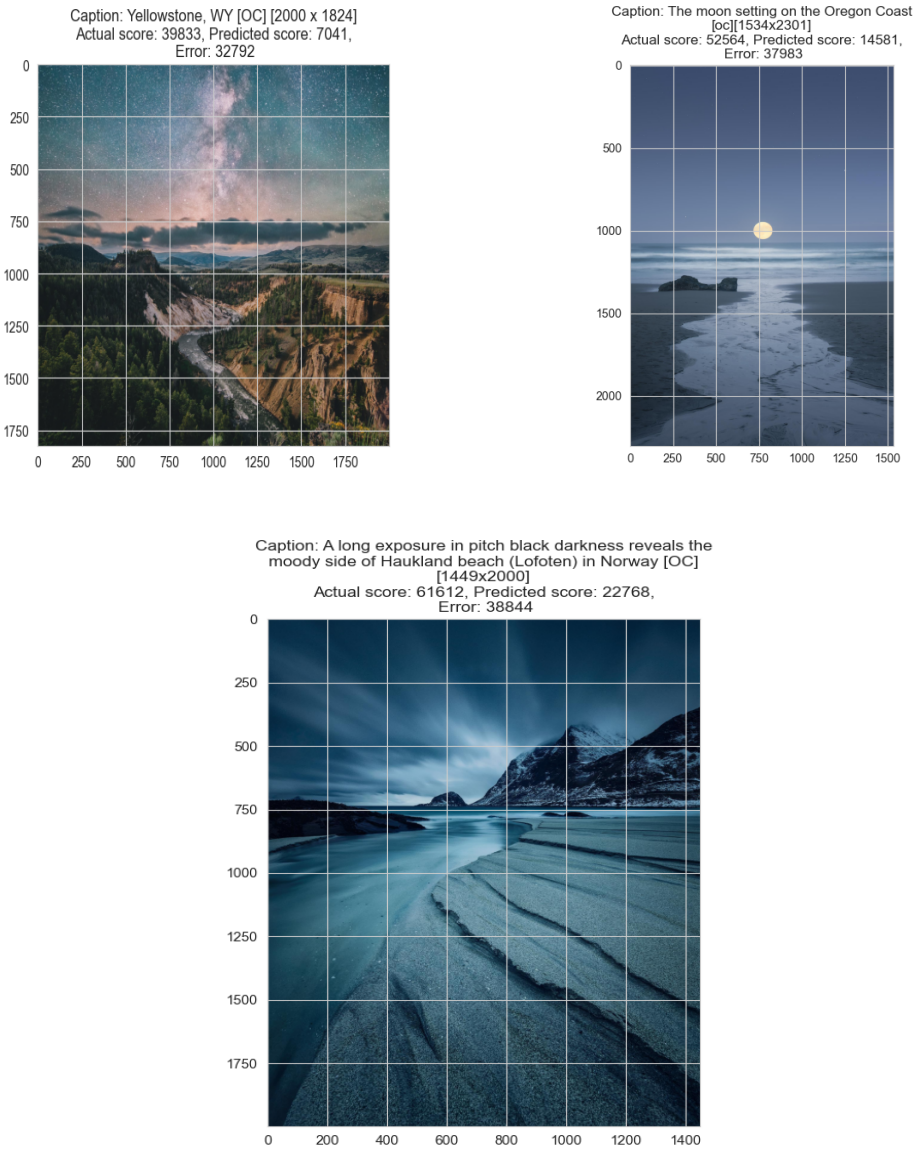


Figure 4.12: Least accurately predicted images

The least accurate images in 4.12, we can see that the maximum error is significantly less than the ones in baseline model indicating a significant improvement. The reason why the model is not having more accurate predictions is the same as previously mentioned, meaning the context although this time they are not nearly as clear as before.

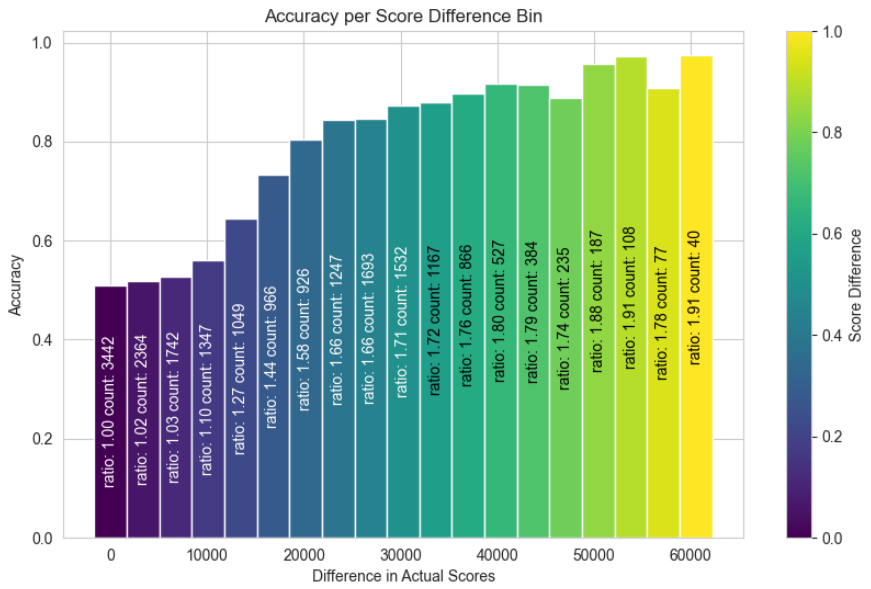


Figure 4.13: Accuracy per score difference bin

In 4.13 we can the the accuracy of the model increases as the difference in the scores increases, this is intuitively what we expect from the model- as the difference between actual scores of post's increases the model's accuracy also increases.

4.3.6 Comparison of baseline model to best models

In examining the results, the noticeable improvement in model performance over the baseline for both the 'pics' and 'earth' datasets becomes clear. For the 'pics' dataset, the implementation of the EfficientNetB0 and XLNet model leads to significant enhancements in predictive accuracy. The substantial reduction in the MSE, from 3,693,914,277.58 down to 1,870,303,297.74, highlights a decreased disparity in predicted scores, signaling a reduction in the variability of prediction errors and a better handling of the outliers. The MAE further supports this by decreasing from 46,687.55 to 34,502.67, indicating that the average distance between the predicted and actual scores has been reduced, thereby implying an increase in prediction accuracy. Finally, the substantial increase in Spearman's rho value from 0.223 to 0.567 signifies the model's strengthened ability to maintain the ordinal ranking of predictions, thereby preserving the relative order of score differences. In the case of the 'earth' dataset, there are also improvements with the introduction of the ResNet50 and BERT model. The MSE reduces significantly from 241,008,867.22 in the baseline model to 173,759,530.37, which suggests a notable decrease in the variability of prediction errors and similarly to pics dataset, a better handling of the outliers. The MAE too reduces from 11,765.44 to 10,616.72, implying an enhancement in the model's accuracy. This is reinforced by the jump in Spearman's rho value from 0.482 to 0.573, which emphasize the model's improved ability in retaining the ordinal ranking of predictions. These results suggest that advanced machine learning models like EfficientNetB0 and

XLNet and ResNet50 BERT can indeed provide substantial improvements in prediction accuracy over the baseline model.

Model		MSE	MAE	Spearman's rho
Pics	Baseline	3,693,914,277.58	46,687.55	0.223
	B0 XLNet	1,870,303,297.74	34,502.67	0.567
Earth	Baseline	241,008,867.22	11,765.44	0.482
	Res50 BERT	173,759,530.37	10,616.72	0.573

Table 4.8: Comparison of model performances

The improved results corroborate the beneficial impact of the enhancements incorporated into the model design and training process. The introduction of early stopping with a patience of 10 serves as a form of regularization, aiding in preventing overfitting by halting the training process if the model's performance on the validation set does not improve for 10 consecutive epochs. This ensures that the model does not waste computational resources nor lose its generalization power by continuing to learn patterns that do not improve its performance. Removing the scaling mechanism from the data preprocessing stage can also be beneficial as evident from Table 4.3. While scaling is generally a good practice for most machine learning models, in this case, maintaining the original range of scores have provided the model with more meaningful information that scaling could have distorted or eliminated. Modifying the architecture of the model allowed it to better accommodate the specific structure and nature of the data. By adjusting layers, activation functions, the model could capture more complex patterns and relationships within the data, thereby boosting its predictive capabilities. Adding a rule-based and context sentiment analysis gave the model a better understanding of the sentiment expressed in the captions of the posts. As the sentiment of a post can significantly impact its popularity, incorporating this factor into the model greatly enriched its interpretative power. Finally, the addition of a caption network added another dimension of information for the model to learn from. By considering the captions in conjunction with the image features, the model had a broader perspective of the post, leading to a more comprehensive and accurate prediction of the popularity.

4.3.7 Comparison of the best models

Now that we have established that our top-performing models significantly outshine the baseline model in predicting image popularity within their respective datasets, the natural next step is to determine which model is more robust and generalizable across diverse datasets. Therefore, we aim to evaluate their performances on a broader spectrum of data. To facilitate this, we merge both the 'pics' and 'earth' datasets into a larger, more diverse dataset. This unified dataset is characterized by its increased dispersion and variability, better representing a wide range of scenarios a model might encounter in a real-world setting. In theory, a model that performs well on this combined dataset is likely to be more generalizable, given its ability to handle a wide array of data characteristics and still maintain high performance. Therefore, this combined dataset provides an excellent platform to assess and compare the generalizability of our models, helping us identify the model

that performs best across different types of data, rather than being optimized for one specific dataset. Upon analyzing Tables 4.6 and 4.7 and Figure 4.1, we notice that the top three models exhibit performance metrics that are closely matched. Given their demonstrated strengths, we decide to further evaluate these top three models from each dataset on our combined dataset. We hypothesize that while their performances are relatively similar within their specific datasets, subtle differences might become more evident in a more diverse and generalized data environment, potentially revealing which models are more versatile and capable across varying dataset.

ImageNet Model	Language Model	MSE	MAE	Spearman's rho
Resnet50	BERT	1,296,020,053.12	26,224.40	0.719
EfficientNetB0	XLNet	1,288,698,285.85	25,439.00	0.709
EfficientNetB3	BERT	1,326,953,471.72	26,606.76	0.703
VGG19	XLNet	1,248,373,281.715	25,082.095	0.728
EfficientNetB3	GPT2	1,320,212,151.90	26,323.80	0.707
EfficientNetB3	XLNet	1,317,306,942.76	25,617.60	0.707

Table 4.9: Performance of top models on combined dataset

Model	combined_metric
VGG19_XLNet	3.000000
B0_XLNet	1.473126
Resnet50_BERT	1.270265
B3_XLNet	0.936649
B3_GPT-2	0.426304
B3_BERT	0.000000

Table 4.10: Combined Ranking of Models

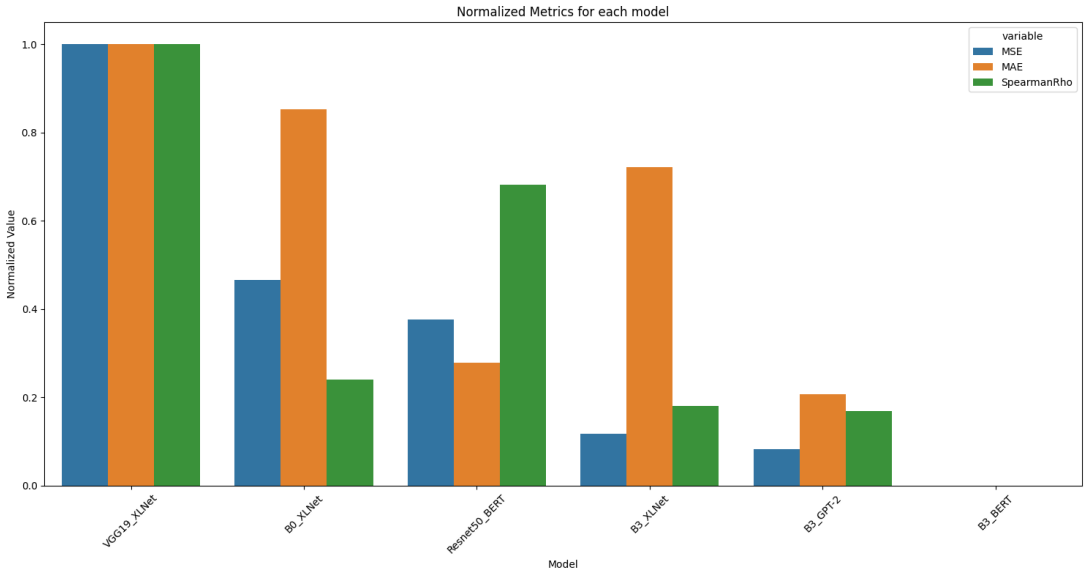
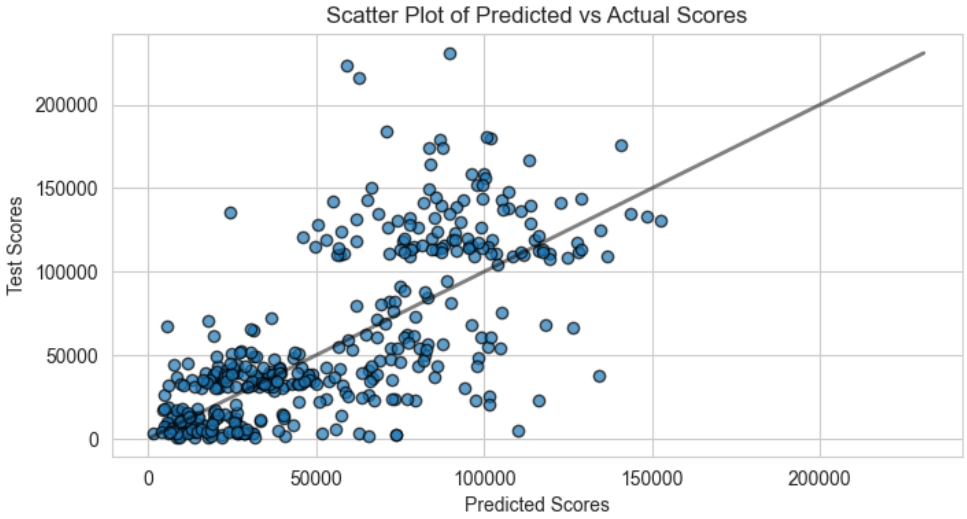


Figure 4.14: Comparison of Normalized Performance Metrics for Each Model

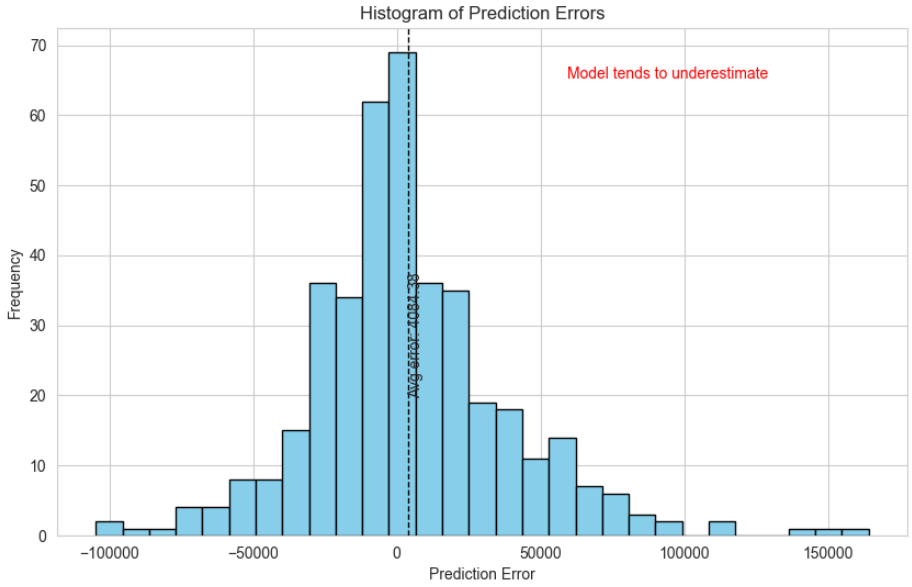
As evidenced by the combined dataset ranking and performance metrics displayed in 4.10 and 4.14, the most effective model combination is VGG-19 and XLNet. This conclusion substantiates our previous speculation that close competitors on each dataset, such as the models using VGG19/XLNet and EfficientNetB0/XLNet, could exhibit varying adaptability when confronted with a broader, more heterogeneous dataset. In this case, VGG19/XLNet takes the lead, demonstrating superior capacity to generalize and provide reliable predictions across diverse data. The reason the top 2 models on combined dataset are also the top two models on 'pics' dataset is due the high diversity of pics dataset.

4.3.8 Analyzing the best model

The best performing model on the combined dataset is the one using the VGG19 architecture for image feature extraction and the XLNet for linguistics feature. The range of the scores in the combined dataset is a substantial 438,065. The model's MSE on this dataset is 1,248,373,281.715, which is significantly lower than the baseline's MSE and the best models MSE on individual dataset, indicating that the model's predictions are less volatile and more concentrated around the true values. The model's MAE is 25,082.095. Compared to the range of the scores, the MAE represents around 5.72% of the range. This indicates that, on average, the model's predictions are within this percentage of the actual scores, which showcases an appreciable level of accuracy in the context of the overall variability in the dataset. Furthermore, the model's Spearman's rho is 0.73, suggesting a strong monotonic relationship between the predicted and actual scores. This highlights that the model is capable of understanding and preserving the order of the data points effectively. Overall, the superior performance of this combined model illustrates the power of leveraging diverse architectures, each tailored to capture different aspects of the data. The VGG19 model, with its depth and small-sized filters, is well-suited for extracting intricate details and patterns in the images. On the other hand, the XLNet model, with its ability to understand the full context of a sentence in any direction, helps in effectively processing and deriving meaning from the post captions. Thus, by harnessing the strengths of these two architectures, the model achieves an impressive performance in predicting the popularity of Reddit posts. In Figure 4.15, we observe several notable characteristics of our model's performance. Firstly, the scatter plot 4.15a shows a dense concentration of points in close proximity to the line of perfect fit, indicating that the model generally yields accurate predictions. Interestingly, we see a higher number of points above the line, suggesting a tendency of the model to underestimate the scores this is further confirmed by the histogram in 4.15b. The histogram distribution with a peak at 0, provides additional evidence of the model's strong performance. This peak signifies that most predictions fall near the true values, thus indicating the model's high accuracy. In conclusion, the various visualizations collectively attest to the effectiveness of the model in predicting the popularity of Reddit posts. Despite its occasional tendency to underestimate, the model demonstrates a high degree of accuracy overall.

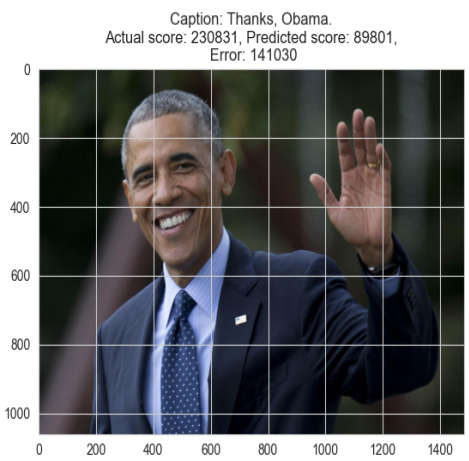


(a) Scatter plot

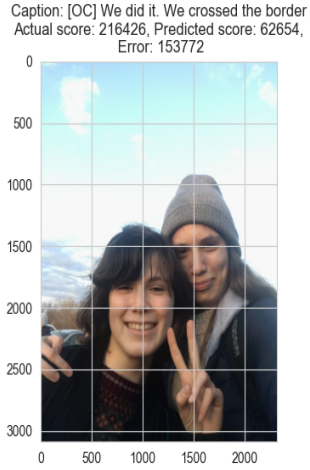


(b) Histogram of Prediction errors

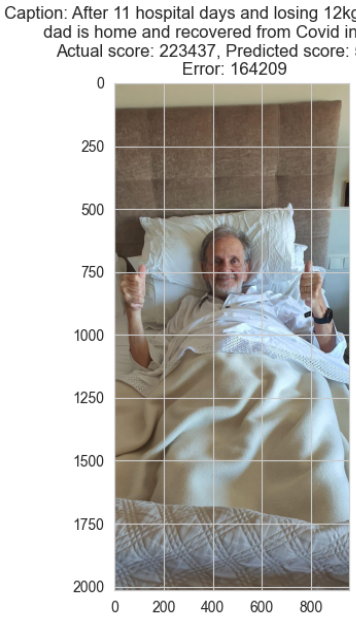
Figure 4.15: Plots



(a) 3rd least accurate prediction



(b) 2nd least accurate prediction



(c) 1st least accurate prediction

Figure 4.16: Least accurately predicted images

Figure 4.16 presents the least accurate predictions produced by our model, offering intriguing insights into the challenges of this task. For instance, consider the images in 4.16a and 4.16b. To accurately capture the popularity of these posts, a deeper understanding of world events would be necessary, an understanding beyond the scope of our model's training data. The simple captions do not provide sufficient context or clues about the image's potential popularity. Similarly, accurately predicting the popularity of the post in 4.16c would require a nuanced understanding of the societal impact of COVID-19, which goes beyond the capabilities of our model. These are the reasons why the model underscores all three images. Interestingly, while prior models trained on the 'pics' dataset had the same image for the top least accurate image as can be seen in 4.9 and 4.3, none of the least accurate images for this model overlap with the prior models. This could be indicative of a shift in the model's understanding of the dataset

nuances, potentially brought about by exposure to a larger and more diverse training set. This reinforces the notion that accuracy in prediction tasks like this one can often be significantly influenced by the depth and variety of context the model has been trained on.

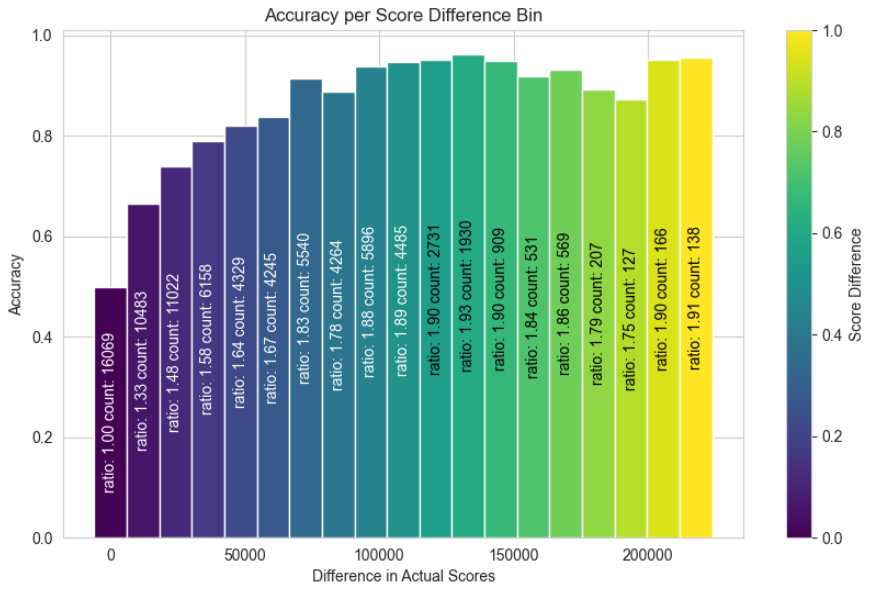


Figure 4.17: Accuracy per score difference bin

As seen in Figure 4.17, there's a clear trend demonstrating an increase in the model's accuracy as the difference between scores increases. This intuitive correlation validates the effectiveness of our model in discerning more noticeable variations in image popularity. When the difference in scores is larger, it generally means there's a more substantial disparity in the popularity of the posts, which our model appears to be accurately capturing. This not only validates the model's performance but also reinforces the inherent value of using machine learning for predicting image popularity based on given dataset.

Conclusion

The primary motivation of this thesis was to study and develop a model capable of predicting the popularity of images posted on Reddit. To achieve this, we harnessed the power of deep learning techniques. In line with the outlined goals, we first built two comprehensive datasets of images posted on Reddit, along with their associated metadata, establishing a rich platform for our subsequent analysis (Goal G1). Following this, we carried out an exhaustive analysis of the datasets, aiming to uncover key insights into the intricate dynamics that drive image popularity on this platform (Goal G2). Armed with these insights, we adopted a dual approach involving state-of-the-art deep learning models (Goal G3). This approach is comprised of a blend of features derived from images and the emotions encapsulated in their captions, with the ultimate goal of predicting the 'score' or popularity of an image on Reddit (Goal G4). Each of the constructed models underwent a rigorous evaluation process, enabling us to gauge their performance and discern their strengths and weaknesses (Goal G5). Thus achieving all the goals of the thesis. The culmination of our work offers persuasive evidence of the efficacy of deep learning techniques in the realm of social media analysis. Our findings indicate that the combination of the VGG-19 model, renowned for its image feature extraction capabilities, and XLNet, a powerful language model, delivered the most accurate predictions. The enhancement in prediction accuracy achieved with this model blend is a significant leap towards the goal of decoding the complex factors that determine image popularity on social media platforms. In conclusion, the current study validates the potency of modern deep learning methodologies in understanding and predicting the popularity of social media content. It contributes valuable insights and offers a robust framework that future research can build upon to further unravel the intricacies of social media dynamics. Through the lens of deep learning, we have moved one step closer to capturing the formula of popularity of a post on social media platforms like Reddit.

Future Work

While the current thesis provides a robust foundation for predicting image popularity on Reddit, there is substantial scope for further improvement and exploration.

Exploring More Advanced Techniques: With the rapid progress in deep learning, newer and more sophisticated models are constantly emerging. Future work could explore leveraging advanced models for feature extraction and prediction.

Transfer Learning: Given extensive computation resources the models could be further refined by applying transfer learning, utilizing pre-trained models on larger and more diverse datasets, and fine-tuning them on the specific dataset before extracting the features.

Increasing the dataset: In future the size of the datasets used on the models can be increased making model more accurate in understanding the different nuances of the data.

Real-time Analysis: Another potential avenue for future research could be the development of a system capable of real-time popularity prediction. Such a system could provide immediate feedback to users about the potential popularity

of their posts.

Comprehensive statistical analysis: Performing a comprehensive statistical analysis to further evaluate and understand the impact of the enhancements made to our model architecture, such as integration of caption network, elimination of scaling, and adding a sentiment analysis component to our model. We anticipate these modifications will significantly improve the model's performance. However, the actual impact and the interactions of these enhancements remain to be systematically assessed and quantified. This in-depth evaluation will not only validate our current work but also illuminate potential avenues for further optimization and refinement of our model.

Through these potential enhancements, we anticipate that future work can make an even greater contribution to the understanding and prediction of social media content popularity.

Bibliography

F.S. Abousaleh, W.-H. Cheng, N.-H. Yu, and Y. Tsao. Multimodal deep learning framework for image popularity prediction on social media. *IEEE Transactions on Cognitive and Developmental Systems*, 13:679–692, 2020.

Alan Akbik, Tanja Bergmann, Duncan Blythe, Kashif Rasul, Stefan Schweter, and Roland Vollgraf. FLAIR: An easy-to-use framework for state-of-the-art NLP. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 54–59. Association for Computational Linguistics, June 2019. doi: 10.18653/v1/N19-4010. URL `https://aclanthology.org/N19-4010`.

Md Zahangir Alom, Tarek M Taha, Christopher Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Brian C Van Esesn, Abdul A S Awwal, and Vijayan K Asari. The history began from alexnet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164*, 2018.

X. Chen, Q. Zhang, M. Lin, G. Yang, and C. He. No-reference color image quality assessment: From entropy to perceptual quality. *EURASIP Journal on Image and Video Processing*, 2019(1):77, 2019.

Columbia University: Digital Video Multimedia Lab. Visual Sentiment Ontology, 2023. URL `https://www.ee.columbia.edu/ln/dvmm/vso/download/vso.html`. Accessed: 023.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, volume 1, pages 4171–4186. Association for Computational Linguistics, 2019.

Keyan Ding, Kede Ma, and Shiqi Wang. Intrinsic image popularity assessment. *ACM International Conference on Multimedia*, pages 1979–1987, 2019.

Eryk Lewinson. Three Approaches to Encoding Time Information as Features for ML Models, 2023. `https://developer.nvidia.com/blog/three-approaches-to-encoding-time-information\-as-features-for-ml-models`.

F. Gelli, T. Uricchio, M. Bertini, A. Del Bimbo, and S.-F. Chang. Image popularity prediction in social media using sentiment and context features. *Proceedings of the 23rd ACM International Conference on Multimedia*, pages 907–910, 2015.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016. `http://www.deeplearningbook.org`.

A.-E. Hassanien and A. Abraham. Computational intelligence in multimedia processing: Recent advances. 96, 2008.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

C. J. Hutto and E. Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 8, pages 216–225, 2014. URL `https://doi.org/10.1609/icwsm.v8i1.14550`.

Image-Net. Image-net large scale visual recognition challenge, 2023. URL `https://www.image-net.org/challenges/LSVRC/`.

jacob6. Eniqa, 2023. `https://github.com/jacob6/ENIQA`.

Ian T Jolliffe and Jorge Cadima. Principal component analysis: a review and recent developments. *Philosophical transactions of the royal society A: Mathematical, Physical and Engineering Sciences*, 374(2065):20150202, 2016.

Y. Ke, X. Tang, and F. Jing. The design of high-level features for photo quality assessment. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1:419–426, 2006.

Y. Luo and X. Tang. Photo and video quality evaluation: Focusing on the subject. *European Conference on Computer Vision*, pages 386–399, 2008.

A. Oliva and A. Torralba. Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, 42(3): 145–175, 2001.

Pushshift. Pushshift's API Documentation, 2023. URL `https://github.com/pushshift/api`. Accessed: 023.

C. Qian, J. Tang, M. Penza, and C. Ferri. Instagram popularity prediction via neural networks and regression analysis. *IEEE Transactions on Multimedia*, pages 2561–2570, 2017.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8): 9, 2019.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Christian Szegedy, Sergey Ioffe, Vincent Vanhoucke, and Alex Alemi. Inception-v4, inception-resnet and the impact of residual connections on learning. *arXiv preprint arXiv:1602.07261*, 2016a.

Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016b.

Mingxing Tan and Quoc V Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning (ICML)*, pages 6105–6114, 2019.

X. Tang, W. Luo, and X. Wang. Content-based photo quality assessment. *IEEE Transactions on Multimedia*, 15(8):1930–1943, 2013.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3156–3164, 2015.

Wikipedia Contributors: Social Media. Social Media, 2023. `https://en.wikipedia.org/wiki/Social_media`.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32, 2019.

List of Figures

List of Tables

A. Attachments

The repository accompanying this thesis contains all the scripts, notebooks, and datasets used in the research.

Folder Structure and Contents

- `Source_code`: This folder includes Python scripts and Jupyter notebooks that were used for the research.
  - `baseline_model_support.py`: Python script that provides support to the baseline model by extracting visual features.
  - `baseline_model_deep.py`: Python script for defining, training, and evaluating the baseline deep learning model.
  - `my_model.py`: Python script for defining, training, and evaluating the enhanced model described in the thesis.
  - `language_model.py`: Python script that generates embeddings from a language model used in the research.
  - `combined_dataset_model.py`: Python script for the combined model that utilizes both earth and pics datasets.
  - `data_scraping.ipynb`: Jupyter notebook that details the data collection process from the respective sources.
  - `performance_of_18_models.py`: Python script for comparing and analyzing the performance of the 18 models utilized in this research.
  - `featureExtract56.mat`: MATLAB file used for feature extraction, specifically color entropy. Additional .mat files provide support to this.
  - Raw data: `earth.csv` and `pics.csv`, which are used to train the models.
  - Processed data: `earth_final_model.csv` and `pics_final_model.csv`, these are the datasets after preprocessing and feature extraction stages.

We did not include the images due to the potential of copyright issues that may arise in future and all the feature that we extracted as they are straight forward to extract from the mentioned scripts.