REPUBLIC OF TURKEY
YILDIZ TECHNICAL UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING

ESTIMATING POPULARITY OF PHOTOS SHARED ON INSTAGRAM

16011043 — Recep Furkan KOCYIGIT

SENIOR PROJECT

Advisor
Assoc. Prof. Mehmet Amac GUVENSAN

June, 2021

ACKNOWLEDGEMENTS

At the conclusion of this study, I would like to express my gratitude to Assoc. Prof. Mehmet Amac GUVENSAN, who generously shared his knowledge with me, as well as my family, who stood by me through all of my challenges and supported me throughout my life.

Recep Furkan KOCYIGIT

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

ANP	Adjectives and Nouns Pairs
CNN	Convolutional Neural Network
DNN	Deep Neural Network
IDE	Integrated Development Environment
JSON	JavaScript Object Notation
NLP	Natural Language Processing
ReLU	Rectified Linear Unit
SVM	Support Vector Machine
SVR	Support Vector Regression

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Estimating Popularity of Photos Shared on Instagram

Recep Furkan KOCYIGIT

Department of Computer Engineering
Senior Project

Advisor: Assoc. Prof. Mehmet Amac GUVENSAN

The use of social media to share content is on the rise these days. Instagram, with nearly 1 billion active users, holds a significant position in this area. The rise in popularity of social media content has become a significant social problem with numerous implications. Effective popularity prediction has a significant impact on social engineering and targeted advertising. Previous studies have shown that users' metadata have a significant impact on predicting popularity. Most of these studies divided popularity into high and low popularity and treated it as a classification problem. In this study, we consider the problem of popularity prediction as a visual content-independent regression problem and compare the predictions of popularity between regression models.

Keywords: Popularity prediction, support vector regression, polynomial regression, deep neural network regression, convolutional neural network regression

ÖZET

Instagram Üzerinde Paylaşılan Fotoğrafların Popülarite Tahmini

Recep Furkan KOCYIGIT

Bilgisayar Mühendisliği Bölümü
Bitirme Projesi

Danışman: Doç. Dr. Mehmet Amaç GÜVENSAN

İçerik paylaşmak için sosyal medyanın kullanımı bu günlerde yükselmektedir. 1 milyara yakın aktif kullanıcısıyla Instagram bu alanda önemli bir yere sahiptir. Sosyal medya içeriğinin popülaritesindeki artış, sayısız sonuçları olan önemli bir sosyal sorun haline gelmektedir. Etkili popülerlik tahmini, sosyal mühendislik ve hedefli reklamcılık üzerinde önemli bir etkiye sahiptir. Önceki çalışmalar, kullanıcıların meta verilerinin popülerliği tahmin etmede önemli bir etkisi olduğunu göstermiştir. Bu çalışmaların çoğu popülerliği yüksek ve düşük popülerlik olarak ikiye ayırmış ve bir sınıflandırma problemi olarak ele almıştır. Bu çalışmada, popülerlik tahmini problemini resim içeriğinden bağımsız bir regresyon problemi olarak ele almakta ve popülerlik tahmini için regresyon modellerini karşılaştırmaktayız.

Anahtar Kelimeler: Popülarite tahmini, destek vektör regresyonu, polinomsal regresyon, derin sinir ağı regresyonu, konvolüsyonel sinir ağı regresyonu

1
Introduction

Social media is a tool that is quite popular due to its user-friendly features. Social media platforms are giving people a chance people to share their photos, videos and other contents. Instagram, which has nearly 1 billion active users, takes a large place in this field. Estimating popularity of a photo can be effective in a variety of areas such as sociological and psychological analysis of the user, manipulating political opinion and targeted advertising.

The motivation of this study is researching which features affect the popularity of photos and whether we can predict the popularity if we have these features. Although previous studies took this problem as classification problem, we take this problem as regression problem and try to predict what percentage of a user's followers like and comment. Therefore, we create a mobile application for users to check their images before they share it.

In Chapter 2, we give details of previous studies in this subject. In Chapter 3, we check what we need to know technically, whether we have enough time, whether the study is against to any law and whether we need to pay any fees while doing this study. In Chapter 4, we analyze the main elements and methods, determine and detail the objectives, information sources and requirements to find the most suitable solution in the project. In Chapter 5, we detail what methods we use to process the dataset, what models we use to predict popularity scores, and the features of the dataset. Chapter 6 shows screenshots of the application made for this study. In Chapter 7, we show the results of the prediction models. In Chapter 8, we evaluate performance of the methods used in this study. Chapter 9 is the evaluation of the results obtained from the study carried out.

2
Related Work

In recent years, many researchers have attempt to estimate the popularity of social media content. These studies began with predicting the popularity of images on the professional photo-sharing platform Flickr[1][2][3][4][5]. Despite they define different popularity scores, they all use the same pipeline to extract different features and then use a regression model to compute the final popularity score.

In 2014, Khosla investigated effects on popularity of features such as color, GIST, LBP and the object prediction based on CNN[4]. They got encouraging results after using both user and content features. McParlane suggested combining background and context features to predict popularity but popularity measure effectiveness was decreasing when there exist no or little textual data for an image[5]. In 2015, Gelli increased number of context features and they were the first one who use visual sentiment features(ANP method)[1]. Their experiment showed some sentiments have a correlation with popularity but still smaller than user features. In 2019, Ortis came with new idea. They tried to predict popularity at time zero. Their datasets were time series and they analyzed change in number of likes and views over 30 days[3]. Ding used visual, context and content features as other researchers used. Also, they added aesthetics and intrinsic popularity score in visual features. They preferred to use DNN-based regression model to obtain the final popularity score.

Moreover, there are several studies that predict popularity of images on the Instagram[6][7]. Hu conducted the first research on predicting the popularity of photos on Instagram in 2014[6]. Their study showed the most popular 8 types of photos and 5 types of users. In 2018, Zohourian combined clustering and regression models in their research[7]. They defined 3 popularity categories(low, medium and high) and popularity score was obtained by dividing the number of likes by the number of followers. They got %90.77 accuracy for popularity class prediction.

3
Feasibility

3.1 Technical Feasibility

In this project, we use Python and Java programming languages. Python is giving developers an advantage in easier and faster writing code due to its many libraries. We used it for web scraping and preprocessing data on Jupyter Notebook and Sublime Text Editor 3. Java is one of the most used programming languages for mobile development. We used it for building the most succesful regression model and creating mobile application on IntelliJ IDE.

Since the target audience of this system is mobile phone users, the program is developed as a software that runs on the Android operating system.

3.2 Hardware Feasibility

The application made at the end of this study runs on API 28, that is, minimum Android 9.0 version.

3.3 Time Management

The Gantt diagram, which includes the distribution of tasks and schedule in this project, is given below.

	Task Name	Duration	Start	ETA
1	Looking related works	15 days	05.03.2021	20.03.2021
2	Setting up the necessary environment	5 days	20.03.2021	25.03.2021
3	Web scraping and preprocessing of data	25 days	25.03.2021	19.04.2021
4	Feature extraction	15 days	19.04.2021	04.05.2021
5	Buiding model	30 days	04.05.2021	03.06.2021
6	Analyzing and improving results	10 days	03.06.2021	13.06.2021
7	Preparing report, presentation and article	5 days	13.06.2021	18.06.2021

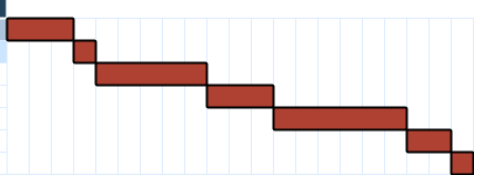

Figure 3.1 Gantt Diagram

3.4 Legal Feasibility

There is no legal obstacle regarding the licenses of the software and hardware used in this project, and the program to be realized in this project does not violate any law, patent, intellectual and industrial rights.

3.5 Economic Feasibility

Since this project was carried out by students, no staff fees were paid. Community version(free) softwares were used while project performing. There is no need for any fee for the project to be carried out.

4
System Analysis

The aim of system analysis is to reveal and identify the project's main elements and functions so that the best approach can be found. In addition, the project's goal is identified.

4.1 Goals

In this project, firstly, a dataset is created in which user metadata and photos are kept. Outliers in this dataset are corrected. Then, regression models are created and compared to predict popularity. A mobile application is designed with the model that gives the best results among these models.

4.2 Requirements

In order to achieve the goal, data collection by web scraping, processing of the collected data, extracting features from this data, creating a model to predict the popularity score, providing a useful interface to the user, receiving visual and other data from the user, processing the data, testing the data processed in the model and must be able to show the result to the user.

For these, the developer should be familiar with Python's "web scraping" and "data processing" topics, as well as the "selenium," "request," and "json" libraries. In addition, the developer should be familiar with the Maven project management tool, have a good command of the Java language, be familiar with Android Studio, and be able to use the "deeplearning4j" library.

4.3 Use Case Diagram

The use case diagram of this system is given below:

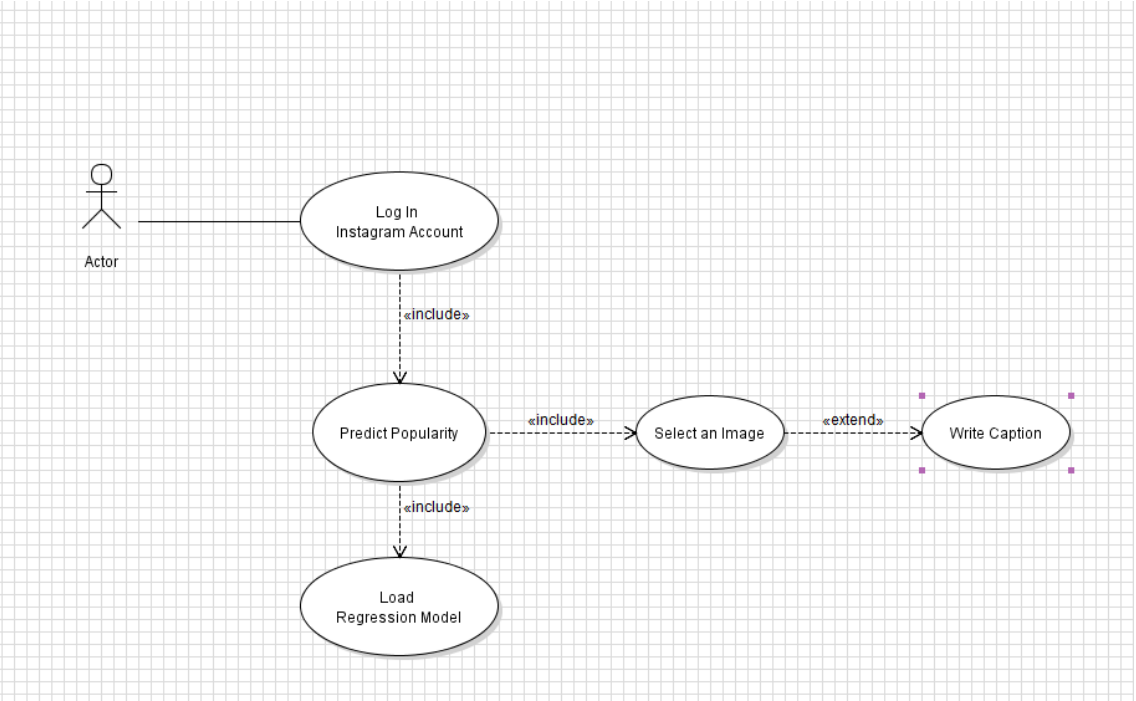

Figure 4.1 Use Case Diagram

As can seen in Figure 4.1, use case of the project for this study is not complex use case diagram. There are just two important steps in this study:

- **Log In Instagram Account :** Logging into users' accounts and getting users' data from Instagram is an important and difficult step.
- **Find Best Model To Predict Popularity :** We want to give best prediction for users. So, we need to try some models and find best model.

4.4 Performance Metrics

In this study, we treated popularity prediction problem as a regression problem. Regression analysis is a supervised machine learning sub-field. It is designed to represent the relationship between a set of features and a continuous target variable. The performance measures for evaluating a regression model are as follows:

- **Mean Absolute Error :**

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \widehat{y}_i| \tag{4.1}$$

- **Mean Squared Error :**

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y}_i)^2 \tag{4.2}$$

- **Root Mean Squared Error :**

$$RMSE = \frac{1}{N}\sqrt{\sum_{i=1}^{N}(y_i - \widehat{y}_i)^2} \tag{4.3}$$

- **Run Time :** The time difference between the model starting to predict and finishing predicting.

5
System Design

The design stages in the system that is desired to be carried out in the project will be explained in this section.

5.1 Input and Output Design

This section shows the inputs and outputs used in our study.

5.1.1 Numerical Details of Data Set

Table showing the count, mean, standard derivation, min, max and quantiles of features used in this study is given below:

	count	mean	std	min	25%	50%	75%	max
number_comments	10147.00000	275.95122	10219.56029	0.00000	0.00000	2.00000	9.00000	994700.00000
number_likes	10147.00000	11425.03765	57695.81156	0.00000	31.00000	108.00000	289.00000	1082924.00000
number_followers	10147.00000	178928.26747	1363466.57202	1.00000	386.00000	1003.00000	3883.00000	56568829.00000
number_followees	10147.00000	1126.52331	1716.89042	0.00000	238.00000	571.00000	1008.00000	7523.00000
number_highlights	10147.00000	7.50882	14.50796	0.00000	1.00000	4.00000	10.00000	310.00000
number_media	10147.00000	427.85878	1486.51260	1.00000	24.00000	88.00000	309.00000	28222.00000
is_business	10147.00000	0.29171	0.45457	0.00000	0.00000	0.00000	1.00000	1.00000
is_professional	10147.00000	0.63201	0.48228	0.00000	0.00000	1.00000	1.00000	1.00000
is_verified	10147.00000	0.06140	0.24007	0.00000	0.00000	0.00000	0.00000	1.00000
number_words_biography	10147.00000	7.80842	7.19654	0.00000	1.00000	6.00000	13.00000	35.00000
number_emojis_biography	10147.00000	1.74879	3.03350	0.00000	0.00000	1.00000	3.00000	52.00000
number_words_caption	10147.00000	22.29644	39.80535	0.00000	1.00000	7.00000	29.00000	416.00000
number_emojis_caption	10147.00000	1.35745	3.48920	0.00000	0.00000	0.00000	1.00000	54.00000
likes_divided_by_followers	10147.00000	0.13666	0.12952	0.00000	0.03824	0.11081	0.19558	1.00000
comments_divided_by_followers	10147.00000	0.00464	0.01265	0.00000	0.00000	0.00072	0.00445	0.38063

Figure 5.1 Details of Features

As seen in Figure 5.1, we have 10,147 different photographic information. The most followed user has 56,6M followers. The photo that get most likes has almost 1,1M likes. And also, the photo that get most comments has almost 1M comments. When

we look at the 3rd quantile for number of likes, comments and followers, it shows that the data of these columns seems like left-skewed because there is a big difference between maximum and 3rd quantile values of these columns. 6% of users have verified accounts and 29% of users use it as a business account. Although most users write their biographies with less words and emojis, they use more words and emojis when they share a photo.

We choose the following features as inputs:

1. number_followees : This feature shows the number of followees the user has.
2. number_highlights : This feature shows the number of highlights the user has.
3. number_media : This feature shows the number of medias the user posted.
4. is_business : This feature shows whether the user account is verified.
5. is_professional : This feature shows whether the user account is professional.
6. is_verified : This feature shows whether the user account is verified.
7. number_words_biography : This feature shows the number of words in user biography.
8. number_emojis_biography : This feature shows the number of emojis in user biography.
9. number_words_caption : This feature shows the number of words in caption that user posted the photo.
10. number_emojis_caption : This feature shows the number of emojis in caption that user posted the photo.

We use two different popularity score to calculate. First score is the number of likes divided by number of followers and second score is the number of comments that divided by number of followers. The scores we used in model are given below:

1. likes_divided_by_followers : This feature shows the score we calculated by dividing the number of likes the post has by the number of followers the user has.

$$LikeScore = \frac{numberoflikes}{numberoffollowers} \tag{5.1}$$

2. comments_divided_by_followers : This feature shows the score we calculated by dividing the number of comments the post has by the number of followers the user has.

$$CommentScore = \frac{numberof\,comments}{numberof\,followers} \tag{5.2}$$

5.1.2 Distribution of Features in Data Set

Histograms showing the distribution of features are given below:

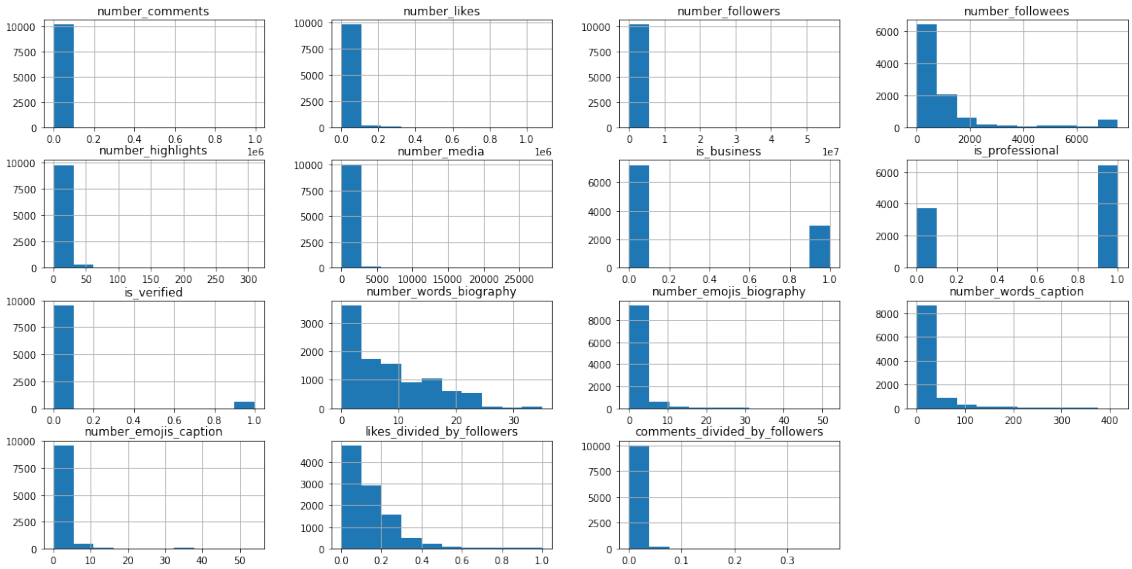

Figure 5.2 Distribution of Features

When we look in Figure 5.2, we can say that almost all numerical features are skewed to the left. This is because some users have really large values as attributes and the number of users like this is not common in our dataset. But that type of users are not outliers to remove from dataset. For example, Cristiano Ronaldo has so much followers and when he share a photo, he gets lots of likes and comments.

5.1.3 Correlation Between Features

Correlation matrix showing the correlations between features are given below:

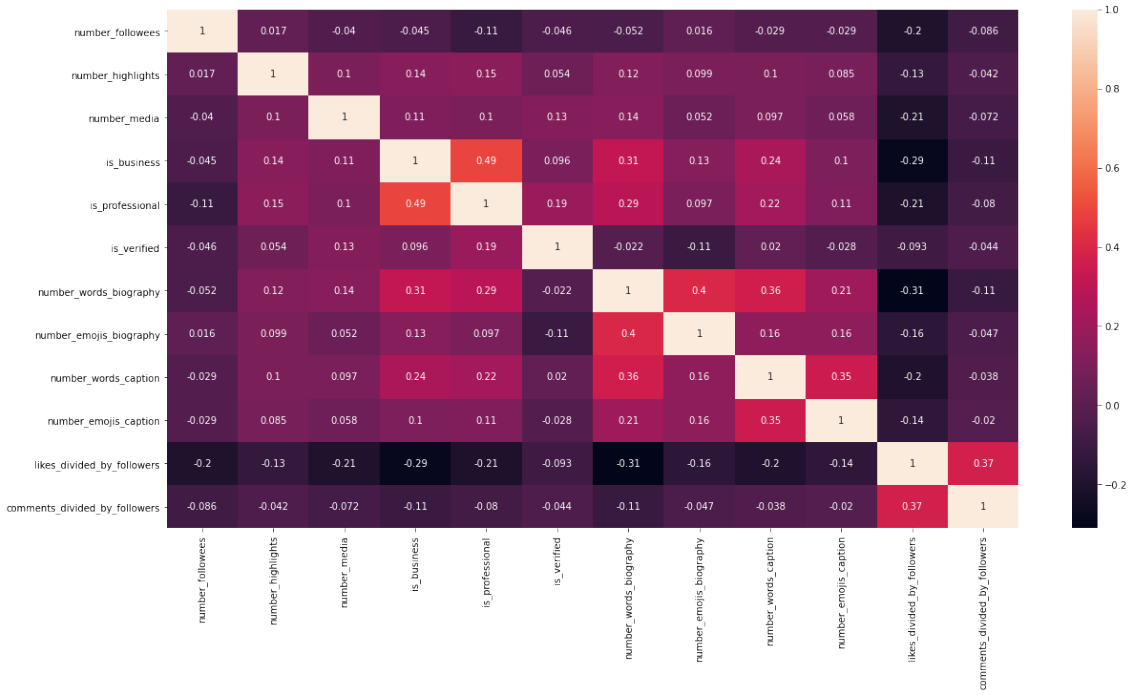

Figure 5.3 Correlation Matrix of Data Set

When we look at the correlation matrix, we can easily say that there is a strong relationship between our two outputs. There is a strong relationship between having business and professional accounts too. Also, having many words in biography can cause to get less number of likes.

5.2 Software Design

The codes are written without charge since the project will be used for an academic study "www.github.com" which will be shared by open source. Since the application will be built for academic purposes, it does not require any encryption structure and virus control is not required.

5.2.1 Web Scraping

A Python-based web scraping tool has been developed to create the necessary database for this system. We use Selenium Firefox Driver to scrape data from Instagram and we code a bot that logs into our Instagram account and gets metadata and photos from usernames given in that driver. The web scraper downloads the JSON file containing the metadata of a user. Then, it downloads each photo given the urls in that JSON file.

5.2.2 Preprocessing

We scrape 1045 JSON file and 10247 photos. In a CSV file, we combine the following features: username, biography, number of followers, number of followees, number of highlights, number of media, whether account is business account, whether account is professional account, whether account joined recently, whether account is verified account, id of the image, number of tagged users, caption, number of comments, post date and number of likes.

Then, we alter the features of the post date. We cannot use it that way because it is timestamp data. We change the data type from timestamp to integer that shows the difference between shared and scraped days. After altering the post date feature, we get number of words and emojis from biography of users and caption of photos. Finally, we combine number of faces and the name of objects features with preprocessed data.

Some users have no followers and some other followers have number of likes more than their number of followers. We clean them from our dataset. At the end, we have 1025 different users and 10147 rows including users' info and photos.

5.2.3 Prediction Models

Our popularity prediction models are built and train using linear, polynomial, support vector, random forest, DNN, and CNN regression models. These models are then compared with their success.

1. **Linear Regression** attempts to model the relationship between two variables by fitting a linear equation to observed data. In our model, Multiple linear regression is used to estimate the relationship between 10 independent variables and one dependent variable.

2. **Polynomial Regression** is a form of regression analysis in which the relationship between the independent variable x and the dependent variable y is modelled as an N th degree polynomial in x. In our study, we choose N as 4.

3. **Support Vector Regression :** Support Vector Machines are supervised learning models using learning algorithms that analyze data for classification and regression analysis in machine learning. When Support Vector Machine is used for regression, it is called as Support Vector Regression. In our study, we choose kernel as radial basis function(RBF) kernel.

4. **Random Forest Regression** is a supervised learning approach for regression that use the ensemble learning method. The ensemble learning method combines predictions from several machine learning algorithms to get a more accurate prediction than a single model. We do not give any parameters to Random Forest Regression model like max_depth.

5. **Deep Neural Network Regression :** Deep Neural Networks are commonly used to solve for classification problems. In our study, we use DNN for regression problem. We choose number of Hidden Dense layers as 6, number of neurons for each hidden layer as 64 and activation as ReLU function. Then, we train the model 500 epochs while batch size is 64. Why we chose the DNN Regression model like this is explained in the experimental results section.

6. **Convolutional Neural Network Regression :** Convolutional neural networks are mostly used for analyzing and classifying image data. In our study, we use CNN model for regression problem. We choose kernel as 2, number of Convolutional layers as 9, filters as 256 and activation as ReLU function. Then, we train the model 500 epochs while batch size is 256. Why we chose the CNN Regression model like this is explained in the experimental results section.

6
Application

The Predictor application is written in Java, a language that has become a standard in scientific computing. It is extremely useful, high-level, open source, fully compatible with Android devices, and extremely flexible.

The user can effectively apply each method and algorithm using a graphical interface and see the results. The login window of the implemented software is shown in the figure below.

Figure 6.1 Login Window of The Predictor

As can be seen in Figure 6.1, the application needs username. When users click login button, the application sends request to get data from target username. If application gets data from Instagram server, the application redirects to predict window.

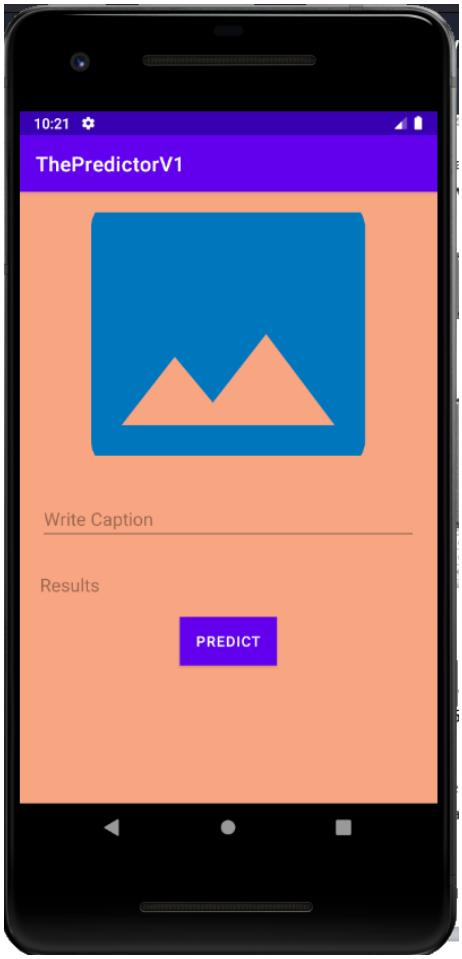

Figure 6.2 Predict Window of The Predictor

When users click on the "Gallery Icon", the application redirects users to the gallery to select an image whose popularity they want to estimate.

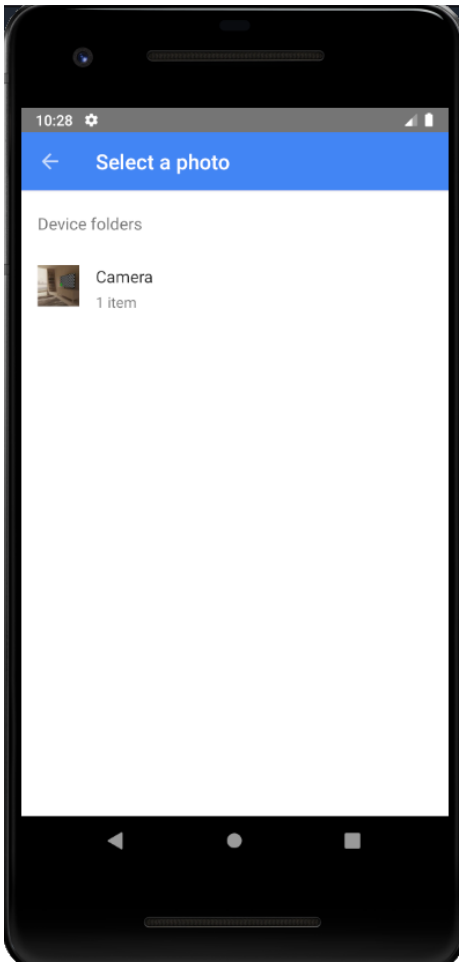

Figure 6.3 Gallery Window of The Predictor

Next, they can write caption if they want, just like on Instagram. When they click the "Predict" button, The model that we trained will return both like and comment score as popularity score. After that, the user can see how many likes and comments they will get below of the caption.

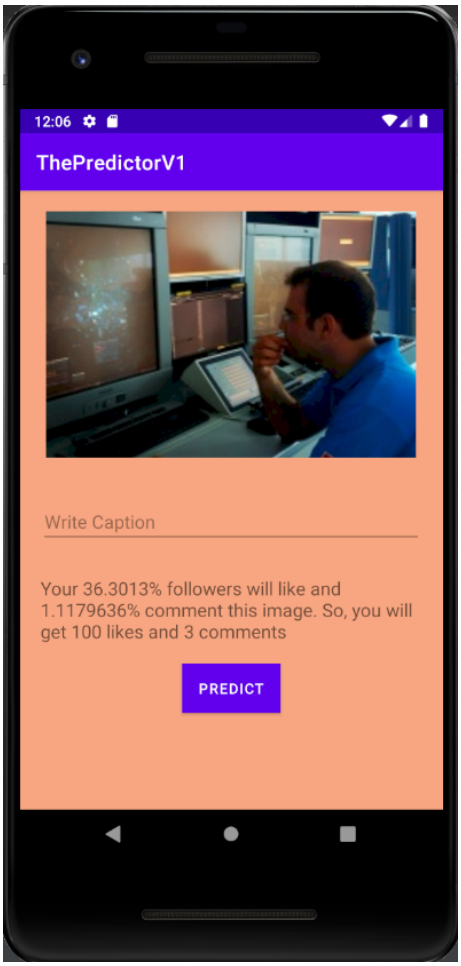

Figure 6.4 An Example of Predicting Popularity

7
Experimental Results

In this section, the results obtained by applying regression models for popularity prediction are compared according to performance metrics. To evaluate the popularity estimation process, "The Predictor" software is compiled and run on i5 processor, 16 GB DDR-4 Ram, Windows 10 OS and the results in the tables below were taken.

7.1 DNN Regression

To find best number of layers and neurons, we build and train 32 different DNN Regression models as epoch 50 and batch size 64. The results of these models are given below:

DNN Model	MAE	MSE	RMSE
2 Hidden Layers 32 Neurons	0.0684	0.0096	0.0984
2 Hidden Layers 64 Neurons	0.0664	0.0102	0.1013
2 Hidden Layers 128 Neurons	0.0693	0.0109	0.1047
2 Hidden Layers 256 Neurons	0.0925	0.0197	0.1404
3 Hidden Layers 32 Neurons	0.0605	0.0087	0.0937
3 Hidden Layers 64 Neurons	0.0607	0.0086	0.0932
3 Hidden Layers 128 Neurons	0.0693	0.0100	0.1002
3 Hidden Layers 256 Neurons	0.0870	0.0147	0.1213
4 Hidden Layers 32 Neurons	0.0574	0.0078	0.0884
4 Hidden Layers 64 Neurons	0.0551	0.0073	0.0859
4 Hidden Layers 128 Neurons	0.0587	0.0082	0.0907
4 Hidden Layers 256 Neurons	0.0609	0.0088	0.0940
5 Hidden Layers 32 Neurons	0.0560	0.0076	0.0874
5 Hidden Layers 64 Neurons	0.0553	0.0073	0.0858
5 Hidden Layers 128 Neurons	0.0551	0.0074	0.0865
5 Hidden Layers 256 Neurons	0.0581	0.0075	0.0869
6 Hidden Layers 32 Neurons	0.0557	0.0076	0.0875
6 Hidden Layers 64 Neurons	**0.0511**	**0.0065**	**0.0809**
6 Hidden Layers 128 Neurons	0.0552	0.0071	0.0842
6 Hidden Layers 256 Neurons	0.0549	0.0067	0.0823
7 Hidden Layers 32 Neurons	0.0605	0.0082	0.0910
7 Hidden Layers 64 Neurons	0.0542	0.0071	0.0847
7 Hidden Layers 128 Neurons	0.0530	0.0067	0.0822
7 Hidden Layers 256 Neurons	0.0516	0.0063	0.0795
8 Hidden Layers 32 Neurons	0.0587	0.0083	0.0914
8 Hidden Layers 64 Neurons	0.0555	0.0078	0.0886
8 Hidden Layers 128 Neurons	0.0542	0.0074	0.0864
8 Hidden Layers 256 Neurons	0.0547	0.0072	0.0854
9 Hidden Layers 32 Neurons	0.0652	0.0088	0.0942
9 Hidden Layers 64 Neurons	0.0583	0.0074	0.0860
9 Hidden Layers 128 Neurons	0.0560	0.0075	0.0869
9 Hidden Layers 256 Neurons	0.0522	0.0066	0.0816

Table 7.1 Results of DNN Models With Different Number of Layers and Neurons For Like Score

As can be seen in Table 7.1, we get the least error from the model with 6 hidden layers and 64 neurons for each hidden layer. Then, we want to check how Dropout affects our model. When we add Dropout, MAE increase from **0.0511** to **0.0691**, MSE increase from **0.0065** to **0.0103** and RMSE increase from **0.0809** to **0.1016**. Therefore, we decide that not using Dropout better for the model.

Next, we want to see the model gives less or more error when batch size changes from **64** to **128** and **256**. As can be seen in Figure 7.1, when we increase batch size, we get

slightly worse results. That is why we choose batch size 64.

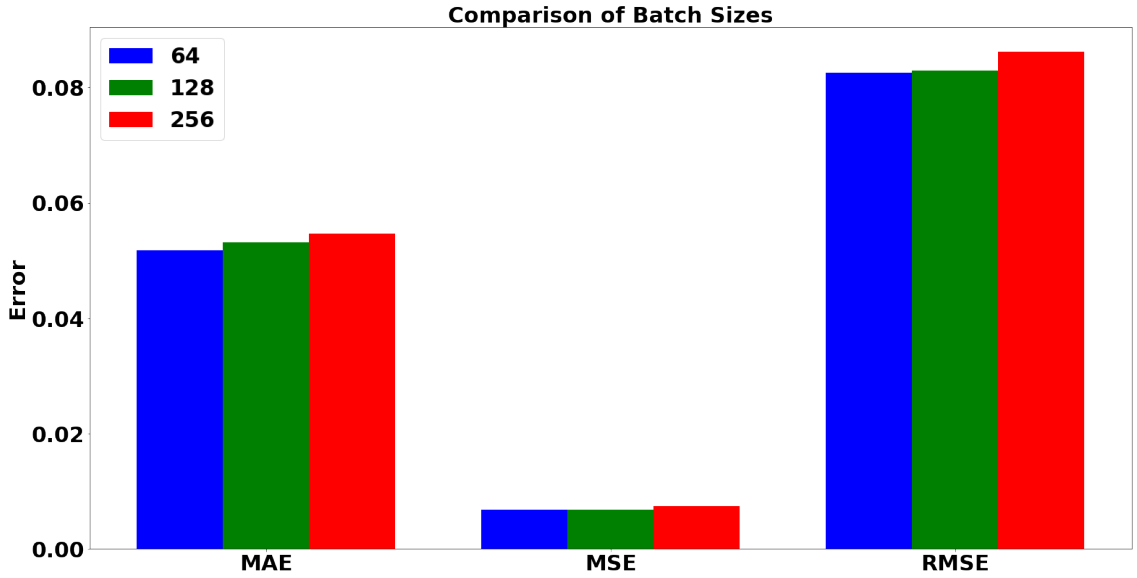

Figure 7.1 Comparison of Batch Sizes For DNN Model

Finally, we want to see how the change of epoch number affects success of the model. We train the model 100, 200, 300, 400 and 500 epochs and we compare them. We get best results with 500 epochs. We stop training model more than 500 epochs because between 400 and 500 epochs the model improve itself just two times. Learning of the model slows down and it shows little improvement.

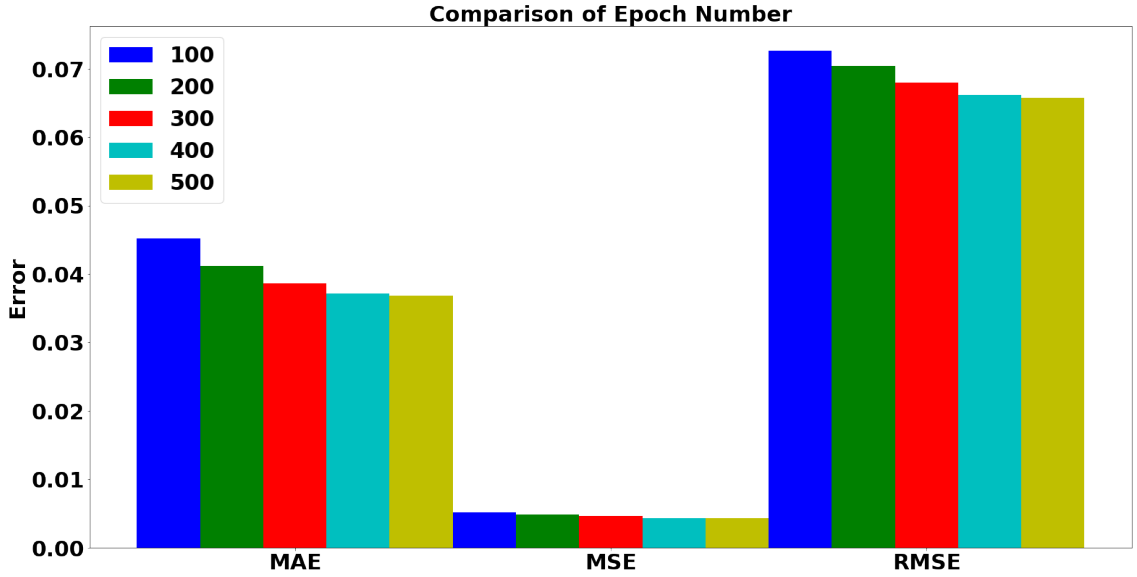

Figure 7.2 Comparison of Epoch Numbers For DNN Model

As a result, our DNN regression model is including 6 hidden layers, 64 neurons for each hidden layer. We choose batch size as 64 and train the model 500 epochs. The

architecture of DNN regression model is given below:

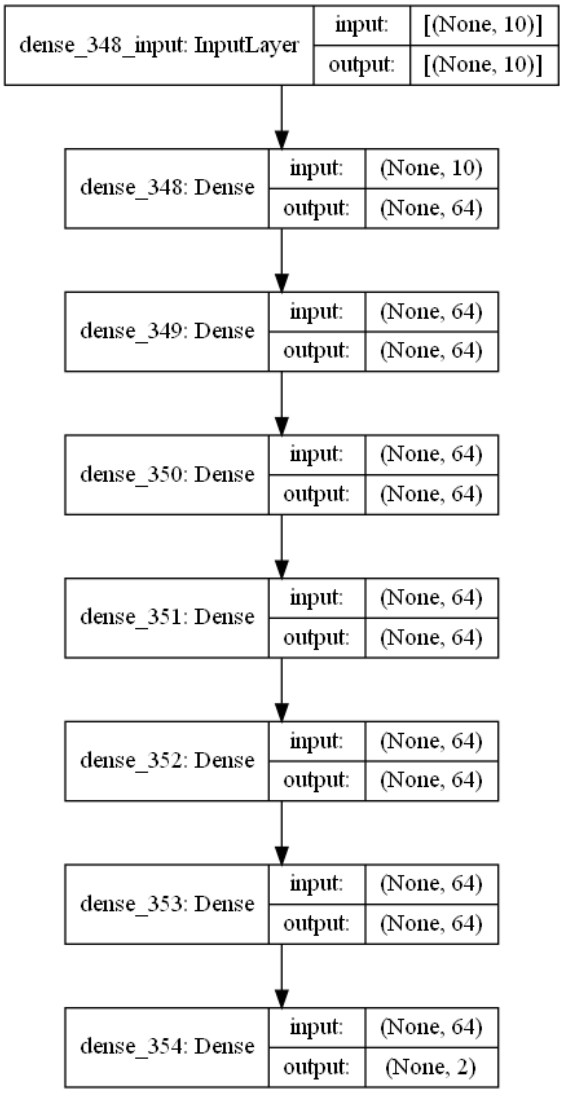

Figure 7.3 Architecture of DNN Regression Model

7.2 CNN Model

To find best number of layers and kernel, we build 11 different CNN Regression models as filters 64, epochs 50 and batch size 256. The results of these models are given below:

CNN Model	MAE	MSE	RMSE
Kernel=2, Layers=2	0.2584	0.4079	0.6387
Kernel=2, Layers=3	0.1439	0.0581	0.2411
Kernel=2, Layers=4	0.0701	0.0118	0.1090
Kernel=2, Layers=5	0.0609	0.0092	0.0963
Kernel=2, Layers=6	0.0546	0.0083	0.0912
Kernel=2, Layers=7	0.0549	0.0082	0.0907
Kernel=2, Layers=8	0.0491	0.0067	0.0823
Kernel=2, Layers=9	**0.0489**	**0.0062**	**0.0792**
Kernel=3, Layers=2	0.2273	0.1440	0.3795
Kernel=3, Layers=3	0.0700	0.0124	0.1114
Kernel=3, Layers=4	0.0612	0.0091	0.0958
Kernel=4, Layers=2	0.1782	0.0835	0.2890
Kernel=4, Layers=3	0.0658	0.0103	0.1017
Kernel=5, Layers=2	0.0885	0.0175	0.1326

Table 7.2 Results of CNN Models With Different Kernels and Number of Layers For Like Score

As can be seen in Table 7.2, when we increase the number of layers, the model gets less error. We get the least error when the model has 9 layers and kernel of the model is 2. Then, we want to check how Dropout affects our model. When we add Dropout, MAE increase from **0.0489** to **0.0665**, MSE increase from **0.0062** to **0.0103** and RMSE increase from **0.0792** to **0.1019**. Therefore, we do not add Dropout to the model.

Next, we want to see how the change of filter affects success of the model. We choose filter of the model as 64, 128 and 256. We get best results when filter is 256. But training time takes more time when we choose filter as 256.

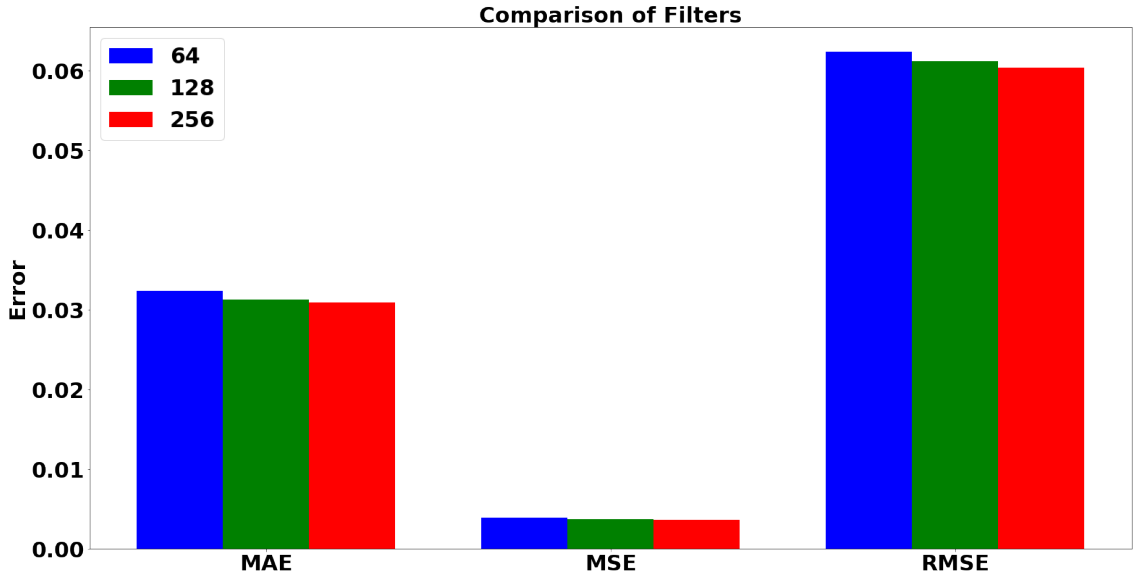

Figure 7.4 Comparison of Filters For CNN Model

Finally, we want to see how the change of epoch number affects success of the model. We train the model 100, 200, 300, 400 and 500 epochs and we compare them. We get best results with 500 epochs. We stop training model more than 500 epochs because learning of the model slows down and it shows little improvement after 500 epochs.

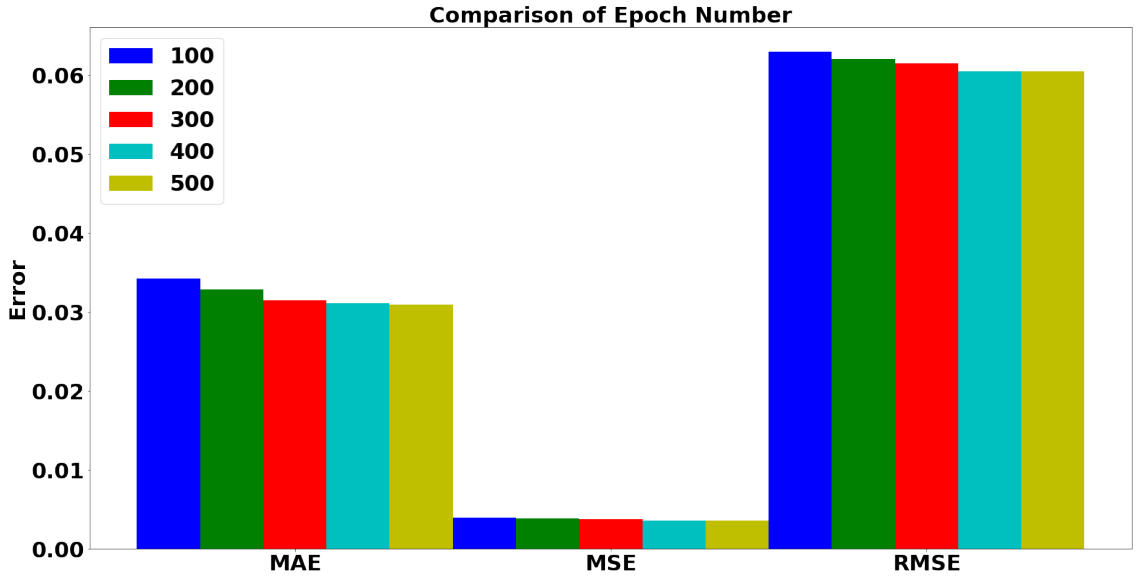

Figure 7.5 Comparison of Epoch Numbers For CNN Model

As a result, our CNN regression model is including 9 convolution layers, 256 filters for each convolution. We choose kernel as 2, batch size 256 and train the model 500 epochs. The architecture of CNN regression model is given below:

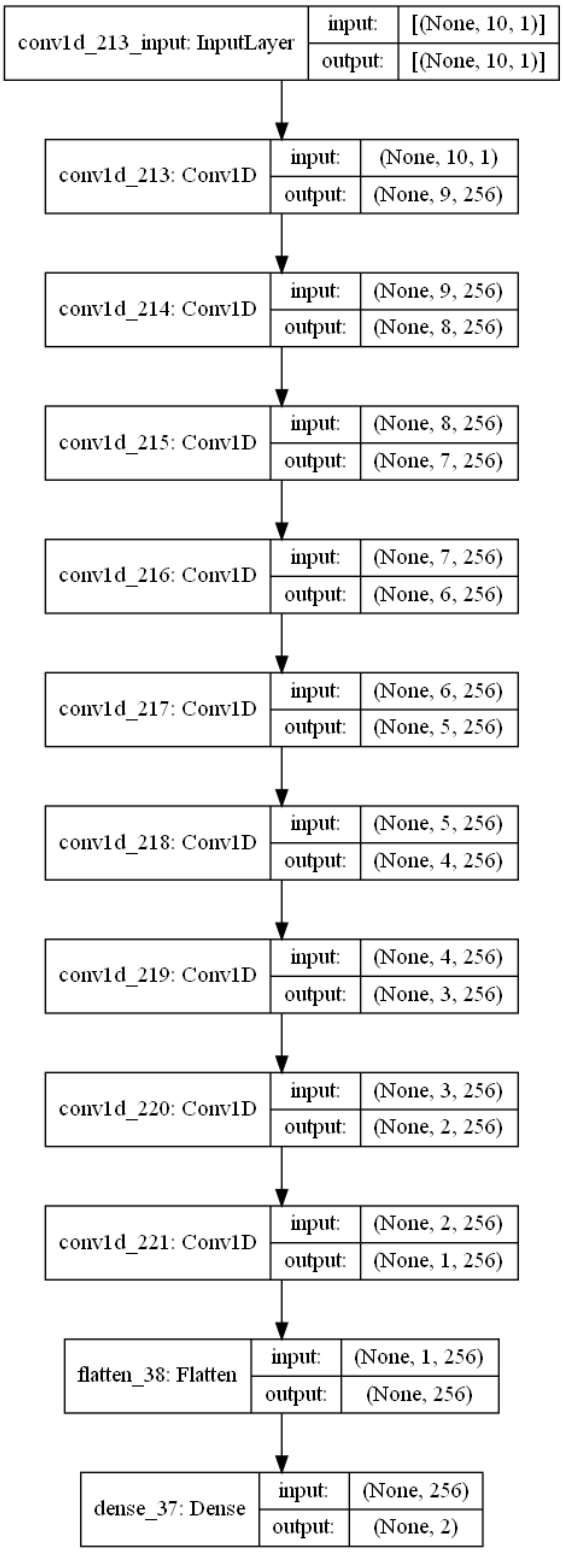

Figure 7.6 Architecture of CNN Regression Model

7.3 Results of Regression Models

In this study, we separate 80% of the data as training data and 20% as test data. Training data contains 8117 rows and test data contains 2030 rows.

The results of regression models for both like and comment scores are given below:

Regression Model	MAE	MSE	RMSE	Run Time(sec)
Linear Regression	0.0800	0.0125	0.1119	0.0051
Polynomial Regression	0.1638	1.9324	1.3901	0.0301
Support Vector Regression	0.0746	0.0114	0.1067	0.4652
Random Forest Regression	**0.0276**	**0.003**	**0.0549**	**0.1098**
DNN Regression	0.0369	0.0043	0.0657	0.3199
CNN Regression	0.0312	0.0038	0.0617	0.3947

Table 7.3 Results of Regression Models For Like Score

We get the best results with Random Forest Regression and the worst results with Polynomial Regression, as shown in Table 7.3. CNN Regression produces good results as well, although it takes nearly three times as long as Random Forest Regression.

Regression Model	MAE	MSE	RMSE	Run Time(sec)
Linear Regression	0.0056	0.0002	0.0142	0.0051
Polynomial Regression	0.0124	0.0078	0.0885	0.0301
Support Vector Regression	0.0956	0.0092	0.0962	0.4652
Random Forest Regression	**0.0035**	**0.0001**	**0.0103**	**0.1098**
DNN Regression	0.0040	0.0002	0.0135	0.3199
CNN Regression	0.0035	0.0001	0.0109	0.3947

Table 7.4 Results of Regression Models For Comment Score

We have two best models, as shown in Table 7.4: Random Forest and CNN Regressions. CNN Regression is not a good option because it takes three times as long as Random Forest Regression.

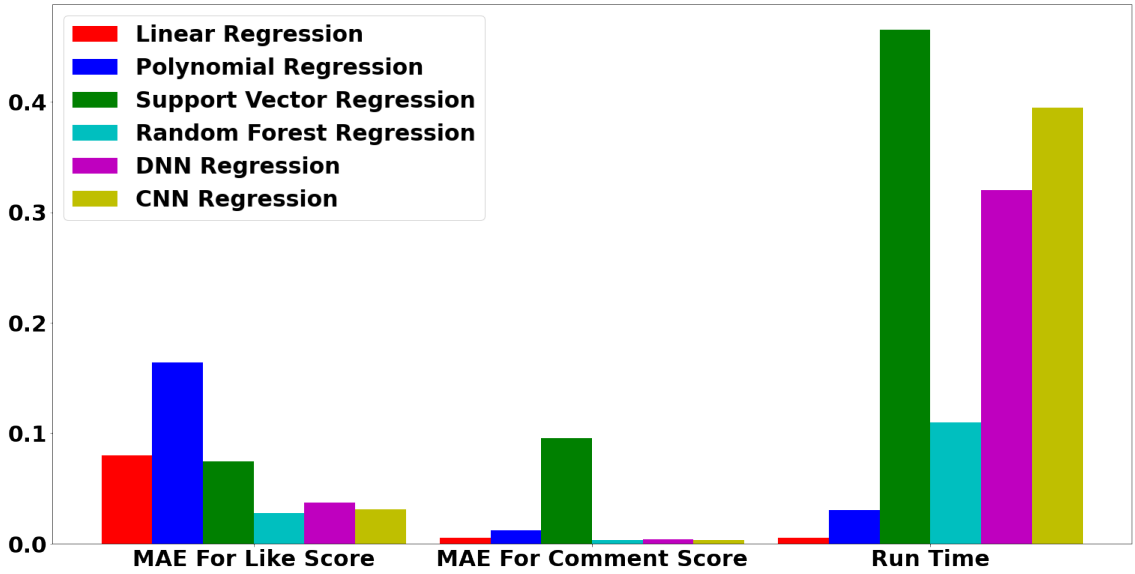

Figure 7.7 Comparison of Models

As can be seen in Figure 7.7, SVR has worst running time and worst MAE for comment score. Polynomial Regression gives worst results for like score. Random Forest Regression gives best results for both like and comment scores. DNN and CNN Regressions also produce good results, however they take significantly longer than Random Forest and they produce worse results than Random Forest Regression. Additionally, Linear Regression does not produce so poor outcomes for comment score. It can be chosen when the computing capability of the device is limited.

Figure 5.1 shows that the majority of users have less than 5,000 followers. We want to see what happens when users are divided into three groups. As a result, we split them into three categories:

1. **Low Number of Followers :** The user that has less than 5,000 followers.
2. **Medium Number of Followers :** The user that has between 5,000 and 500,000 followers.
3. **High Number of Followers :** The user that more than 500,000 followers.

7.3.1 Low Number of Followers

There is 6,308 training data for users with less than 5,000 followers among the 8,117 training data. Also, there is 1,573 test data for users with less than 5,000 followers among the 2,030 test data. As can seen from these numbers, majority of the users have less than 5,000 followers. The minimum number of likes for the photos shared

by users with less than 5,000 followers is 0 and the maximum number of likes is 1,413. The minimum number of comments for the photos shared by users with less than 5,000 followers is 0 and the maximum number of comments is 144. We decide to compare Random Forest Regression model that is built and trained for low Number of followers with Random Forest Regression model that is built and trained for all training data.

Random Forest Regression Model	MAE	MSE	RMSE
Be Built&Trained With All Data RFR Model	0.0304	0.0035	0.0590
Be Built&Trained With Low Number of Followers Data RFR Model	0.0304	0.0035	0.0591

Table 7.5 Comparison of Like Scores For Low Number of Followers

Random Forest Regression Model	MAE	MSE	RMSE
Be Built&Trained With All Data RFR Model	0.0039	0.0001	0.0090
Be Built&Trained With Low Number of Followers Data RFR Model	0.0039	0.0001	0.0090

Table 7.6 Comparison of Comment Scores For Low Number of Followers

When we compare the RFR model that trained with all data and the RFR model trained with low number of followers data, we get same error rate for comments and likes, as shown in Tables 7.5 and 7.6. For 255 out of 1573 low number of followers test data, the model successfully predicts the number of likes. Also, for 696 out of 1573 low number of followers test data, the model successfully predicts the number of comments.

Figure 7.8 One of The Best Prediction For Low Number of Followers Data

Figure 7.9 The Worst Prediction For Low Number of Followers Data

The picture shown in Figure 7.8 has 4 comments, 181 likes and the user who shared this photo has 1579 followers, 1090 followees, 10 highlights, 81 medias. This account is not business and verified account but it is professional account. The biography for this user has 5 words and 0 emojis. The caption for this photo has 123 words and

0 emojis. The model correctly predicts both number of comments and likes. On the other hand, the picture shown in Figure 7.9 has 2 comments, 972 likes and the user who shared this photo has 1145 followers, 640 followees, 1 highlights, 30 medias. This account is not business, verified and professional account. The biography for this user has 13 words and 3 emojis. The caption for this photo has 2 words and 0 emojis. The model predicts it as 88 likes and 2 comments. The model has 21.8659 MAE for prediction of likes and 2.0922 MAE for comments. In other words, when a user with less than 5,000 followers shares a photo, the model may predict 22 less or more likes and 2 less or more comments than be expected on average.

7.3.2 Medium Number of Followers

There is 1,354 training data for users with between 5,000 and 500,000 followers among the 8,117 training data. Also, there is 360 test data for users with between 5,000 and 500,000 followers among the 2,030 test data. The minimum number of likes for the photos shared by users with between 5,000 and 500,000 followers is 3 and the maximum number of likes is 125,654. The minimum number of comments for the photos shared by users with between 5,000 and 500,000 followers is 0 and the maximum number of comments is 159,409. As can seen from these numbers, The medium number of follower group has a smaller number of users than the low number of follower group. We compare Random Forest Regression model that is built and trained for medium number of followers with Random Forest Regression model that is built and trained for all training data.

Random Forest Regression Model	**MAE**	**MSE**	**RMSE**
Be Built&Trained With All Data RFR Model	0.0186	0.0016	0.0397
Be Built&Trained With Medium Number of Followers Data RFR Model	0.0188	0.0016	0.0397

Table 7.7 Comparison of Like Scores For Medium Number of Followers

Random Forest Regression Model	**MAE**	**MSE**	**RMSE**
Be Built&Trained With All Data RFR Model	0.0019	0.0002	0.0125
Be Built&Trained With Medium Number of Followers Data RFR Model	0.002	0.0002	0.0131

Table 7.8 Comparison of Comment Scores For Medium Number of Followers

When we compare the RFR model that trained with all data and the RFR model trained with medium number of followers data, we get almost same error rate for comments

and likes, as shown in Tables 7.7 and 7.8. For 14 out of 360 medium number of followers test data, the model successfully predicts the number of likes. Also, for 55 out of 360 medium number of followers test data, the model successfully predicts the number of comments.

Figure 7.10 One of The Best Prediction For Medium Number of Followers Data

Figure 7.11 The Worst Prediction For Medium Number of Followers Data

The picture shown in Figure 7.10 has 256 comments, 58,239 likes and the user who shared this photo has 454,726 followers, 168 followees, 7 highlights, 220 medias. This account is business, verified and professional account. The biography for this user has 11 words and 1 emojis. The caption for this photo has 81 words and 1 emojis. The model correctly predicts both number of comments and likes. On the other hand, the picture shown in Figure 7.11 has 499 comments, 70,773 likes and the user who shared this photo has 167,084 followers, 289 followees, 0 highlights, 30 medias. This account is business and professional but not verified account. The biography for this user has 6 words and 0 emojis. The caption for this photo has 2 words and 0 emojis. The model predicts it as 35,560 likes and 398 comments. The model has 1,260.6139 MAE for prediction of likes and 314.1 MAE for comments. In other words, when a user with between 5,000 and 500,000 followers shares a photo, the model may predict 1,261 less or more likes and 314 less or more comments than be expected on average. The reason that the MAE of the number of likes and comments is larger than that of the low number of followers group is that neither the number of likes nor the number of comments are not predicted directly. When we multiply the predicted like or comment score by the number of followers, we see that the mistake grows as the number of followers grows. If we assume that the users have a medium number of followers, we multiply the like and comment scores by a quantity between 5,000 and 500,000.

7.3.3 High Number of Followers

There is 455 training data for users with more than 500,000 followers among the 8,117 training data. Also, there is 97 test data for users with more than 500,000

followers among the 2,030 test data. The minimum number of likes for the photos shared by users with more than 500,000 followers is 62 and the maximum number of likes is 1,082,924. The minimum number of comments for the photos shared by users with more than 500,000 followers is 0 and the maximum number of comments is 994,700. As can seen from these numbers, The high number of followers group is the smallest group in dataset. We compare Random Forest Regression model that is built and trained for medium number of followers with Random Forest Regression model that is built and trained for all training data.

Random Forest Regression Model	**MAE**	**MSE**	**RMSE**
Be Built&Trained With All Data RFR Model	0.0154	0.0008	0.028
Be Built&Trained With High Number of Followers Data RFR Model	0.0144	0.0007	0.0267

Table 7.9 Comparison of Like Scores For High Number of Followers

Random Forest Regression Model	**MAE**	**MSE**	**RMSE**
Be Built&Trained With All Data RFR Model	0.0023	0.0003	0.0185
Be Built&Trained With High Number of Followers Data RFR Model	0.0022	0.0003	0.0185

Table 7.10 Comparison of Comment Scores For High Number of Followers

When we compare the RFR model that trained with all data and the RFR model trained with high number of followers data, the RFR model trained with high number of followers data showed just a little bit better performance, as shown in Tables 7.9 and 7.10. For 6 out of 90 medium number of followers test data, the model successfully predicts the number of likes. Also, for 6 out of 90 medium number of followers test data, the model successfully predicts the number of comments.

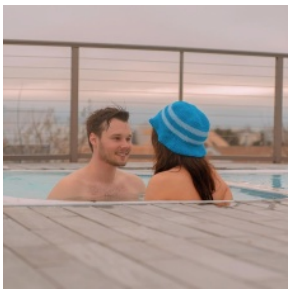

Figure 7.12 One of The Best Prediction For High Number of Followers Data

Figure 7.13 The Worst Prediction For High Number of Followers Data

The picture shown in Figure 7.12 has 354 comments, 152,401 likes and the user who shared this photo has 3,740,074 followers, 615 followees, 13 highlights, 623 medias. This account is business, verified and professional account. The biography for this user has 17 words and 0 emojis. The caption for this photo has 12 words and 3 emojis. The model correctly predicts both number of comments and likes. On the other hand, the picture shown in Figure 7.13 has 994,700 comments, 986,073 likes and the user who shared this photo has 5,382,694 followers, 728 followees, 12 highlights, 61 medias. This account is professional and verified but not business account. The biography for this user has 1 words and 0 emojis. The caption for this photo has 1 words and 0 emojis. The model predicts it as 392,668 likes and 14,832 comments. The model has 33,521.2474 MAE for prediction of likes and 10,872.701 MAE for comments. In other words, when a user with more than 500,000 followers shares a photo, the model may predict 33,521 less or more likes and 10,873 less or more comments than be expected on average. This time, although the MAE of the number of likes and comments seems to be high, the number of followers is also very high.

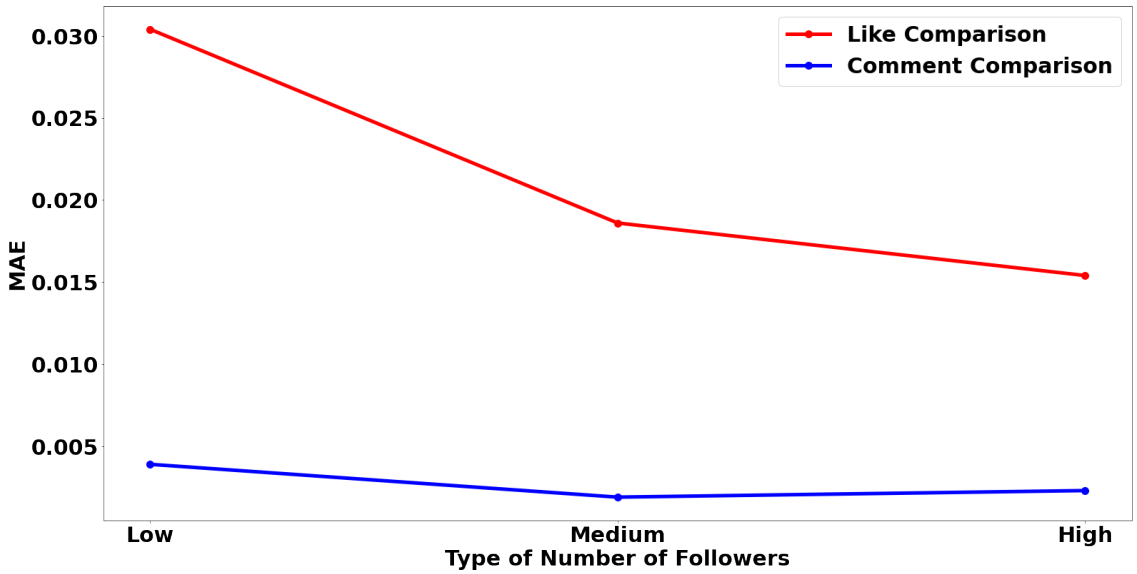

Figure 7.14 Comparison of Models

As can be seen in Figure 7.14, we get most MAE for the users has less than 5,000 followers. Because we define like and comment scores as number of likes or comments divided by number of followers. MAE seems to be more when the number of followers is low. For this reason, we get less MAE in medium and high number of followers.

8
Performance Analysis

As can be seen in Table 7.1, we get the least error for DNN Regression when the model has 6 hidden Dense layers and for each hidden layer 64 neurons. Then, we observe that effect of Dropout and batch size. At the end, we determine that we get best results when we train DNN Regression model with 500 epochs and 64 batch size.

As can be seen in Table 7.2, we get the least error for CNN Regression when we choose kernel as 2 and number of convolution layer as 9. When we add Dropout, we determine that it causes getting more error. Then we check the affects of number of filters for each convolution layer. We get least error when we choose filter as 256. At the end, we determine that we get best results when we train CNN Regression model with 500 epochs and 256 batch size.

The Random Forest Regression model yields the model with the least number of error for the scores of likes and comments, as shown in Tables 7.4 and 7.5. The reason of getting least error from Random Forest Regression is that it is a powerful and accurate model. It usually works well on a wide range of situations, including those with non-linear relationships. On the other hand, Linear Regression model gives not much errors for comment score and it works faster than other models.

If we look at Table 7.6 and 7.7, we find that we get same errors from the Random Forest Regression models trained with all data and with low number of followers data. When we look at Table 7.8 and 7.9, we deduce that we get the same results from the medium follower group as the low follower group. When we train Random Forest Regression model with high number of followers group, we get slightly better results than the model trained with all data as you can see at Table 7.10 and 7.11.

9
Conclusion

The Predictor application is developed to solve the problem of finding the popularity of shared photos on Instagram . This application produces clear and fast results for the end user.

As can be seen from all the results summarized in Chapter 8, the least error is obtained from Random Forest Regression. Therefore, we prefer to use this model in the mobile application. It predicts what percentage of followers like a photo, with an average error of 2.7 percent and what percentage of followers comment a photo, with an average error of 0.3 percent. It predicts 1221 of 2030 test data with less than 20 errors and 277 of 1221 without any error. Also, Linear Regression shows a fast and successful result for comment score. Apart from this, it has been observed that the worst model for popularity prediction is Support Vector Regression.

We use ten features to predict popularity with two popularity score in this study. The researchers who want to work on this subject can try to increase number of features, improve the CNN Regression model, define different popularity score and use features of images to get better results. Also, we do not use textual features in this study. They can use textual features with NLP methods to improve results.

References

[1] F. Gelli, T. Uricchio, M. Bertini, A. Del Bimbo, and S.-F. Chang, "Image popularity prediction in social media using sentiment and context features," in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 907–910.

[2] K. Ding, R. Wang, and S. Wang, "Social media popularity prediction: A multiple feature fusion approach with deep neural networks," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 2682–2686.

[3] A. Ortis, G. M. Farinella, and S. Battiato, "Predicting social image popularity dynamics at time zero," *IEEE Access*, vol. 7, pp. 171 691–171 706, 2019.

[4] A. Khosla, A. Das Sarma, and R. Hamid, "What makes an image popular?" In *Proceedings of the 23rd international conference on World wide web*, 2014, pp. 867–876.

[5] P. J. McParlane, Y. Moshfeghi, and J. M. Jose, ""nobody comes here anymore, it's too crowded"; predicting image popularity on flickr," in *Proceedings of international conference on multimedia retrieval*, 2014, pp. 385–391.

[6] Y. Hu, L. Manikonda, and S. Kambhampati, "What we instagram: A first analysis of instagram photo content and user types," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 8, 2014.

[7] A. Zohourian, H. Sajedi, and A. Yavary, "Popularity prediction of images and videos on instagram," in *2018 4th International Conference on Web Research (ICWR)*, IEEE, 2018, pp. 111–117.

Curriculum Vitae

FIRST MEMBER

Name-Surname: Recep Furkan KOCYIGIT
Birthdate and Place of Birth: 27.10.1998, Istanbul
E-mail: furkankocyigitfk@gmail.com
Phone: 0541 726 49 61
Practical Training: Yapi Kredi Bank, Software Engineering Intern

Project System Informations

System and Software: Windows, Python, Java, Android Studio
Required RAM: 16GB
Required Disk: 256GB