# CHAPTER IV
# ANALYSIS AND DESIGN

## 4.1. Analysis

### 4.1.1. Classes Determination

Naive Bayes algorithm is an algorithm which usually used in classifying something. Therefore, determine the classes in this project is very important. In this project, the video data will be categorized into two classes, namely class for video that can be recommended for user and class for video that cannot be recommended for user (Recommended and Not Recommended).

### 4.1.2. The Attributes

In Naive Bayes algorithm, the attributes need to be determined. The attributes that will be used in this project are category (music, sports, food, and etc), YouTube account (Agnes Mo Official, GMS official, Veevo, and etc), duration (<10minutes, 10-60minutes, >1hour), date published, and view count (<500 viewers, 500-5.000 viewers, 5.000-50.000 viewers, 50.000-500.000 viewers, 500.000-5.000.000 viewers, > 5.000.000 viewers).

### 4.1.3. Prepare The Data

The Naive Bayes algorithm requires two kinds of data, the training data and testing data. Training data in this project consists of a set of video data which has been watched by the user and a set of video data which has never been watched by the user. To get the video data which has been watched by the user, a list of video needs to be shown to the user. In this project, the Home page will display a list of popular videos which is gotten by using YouTube API.

A video that was clicked will be saved into a text file named “seen.txt”. The data that exists in the “seen.txt” mixed with some of the video data that has never been watched, then saved in a text file named “training.txt”. While the video data that has never been watched will be saved into a text file named “testing.txt”.

| ID | Kategori | Durasi | Youtuber Account | Date upload | Keterangan |
|---|---|---|---|---|---|
| 1 | Horror | 15 menit - 1 jam | Movie Official | 2016-12-01 | ditonton |
| 2 | Musik | <10 menit | Tulus | 2016-12-01 | ditonton |
| 3 | Horror | 15 menit - 1 jam | Top Movie | 2016-12-02 | ditonton |
| 6 | Musik | 10 - 15menit | Agnez Mo | 2016-12-01 | ditonton |
| 9 | Musik | 10 - 15menit | Project Pop | 2016-12-04 | ditonton |
| 10 | Komedi | 10 - 15menit | Movie Official | 2016-12-01 | ditonton |

*Figure 1. Example of data that will save in “Seen.txt”*

| ID | Kategori | Durasi | Youtuber Account | Date upload | Keterangan |
|---|---|---|---|---|---|
| 1 | Horror | 15 menit - 1 jam | Movie Official | 2016-12-01 | ditonton |
| 2 | Musik | <10 menit | Tulus | 2016-12-01 | ditonton |
| 3 | Horror | 15 menit - 1 jam | Top Movie | 2016-12-02 | ditonton |
| 6 | Musik | 10 - 15menit | Agnez Mo | 2016-12-01 | ditonton |
| 9 | Musik | 10 - 15menit | Project Pop | 2016-12-04 | ditonton |
| 10 | Komedi | 10 - 15menit | Movie Official | 2016-12-01 | ditonton |
| 4 | Horror | 1jam - 2jam | Movie Official | 2016-12-02 | tidak ditonton |
| 5 | Horror | 10 - 15menit | Movie Official | 2016-12-04 | tidak ditonton |
| 7 | Komedi | <10 menit | Candra Liow | 2016-12-02 | tidak ditonton |
| 8 | Musik | <10 menit | Agnez Mo | 2016-12-02 | tidak ditonton |

*Figure 2. Example of data that will save in "Training.txt"*

| ID | Kategori | Durasi | Owner | Date upload | Keterangan |
|---|---|---|---|---|---|
| 4 | Horror | 1jam - 2jam | Movie Official | 2016-12-02 | ???? |
| 5 | Horror | 10 - 15menit | Movie Official | 2016-12-04 | ???? |
| 7 | Komedi | <10 menit | Candra Liow | 2016-12-02 | ???? |
| 8 | Musik | <10 menit | Agnez Mo | 2016-12-02 | ???? |
| 11 | Horror | 15 menit - 1 jam | Movie Official | 2016-12-01 | ???? |
| 12 | Musik | <10 menit | Tulus | 2016-12-01 | ???? |
| 13 | Horror | 15 menit - 1 jam | Top Movie | 2016-12-02 | ???? |
| 14 | Musik | 10 - 15menit | Agnez Mo | 2016-12-01 | ???? |
| 15 | Musik | 10 - 15menit | Project Pop | 2016-12-04 | ???? |
| 16 | Komedi | 10 - 15menit | Movie Official | 2016-12-01 | ???? |

*Figure 3. Example of data that will save in "Testing.txt"*

### 4.1.4. Naive Bayes Algorithm

The Naive Bayes algorithm has formula :

$$\mathbf{P(Y|X) = P(X|Y) * P(Y)}$$

“X” is an “attributes” and “Y” is the “class”. Then, P(Y) can define as a probability of each class in a set of data which usually called as “Prior Probability”. P(X|Y) is a probability which gotten from a number of specific feature which stayed in specific class compared with a number of specific feature in a set of data. P(X|Y) usually called as “Likehood”. P(Y|X) is a result from multiplication between likehoods and prior.

#### 4.1.4.1. Prior Probability Calculation

In this case, prior probability is probability of class (keterangan) frecuency in training data. To calculate it, the frecuency of each class must be known. Then, that number of frecuency divided by the number of total of all datas in training data. In the example, *keterangan = “ditonton”* can be assume as “Recommended class”. While *keterangan = “tidak ditonton”* can be assume as “Not Recommended class”.

P(Keterangan = “ditonton”) = 6 / 10
= 0,6

P(Keterangan = “tidak ditonton”) = 4 / 10
= 0,4

4.1.4.2. Likehood Probability

Likehood is a probability of attribute frecuency which is stayed in specific class. In this step, each of testing data will be compared with training datas.

| ID | Kategori | Durasi | Owner | Date upload | Keterangan |
|---|---|---|---|---|---|
| 4 | Horror | 1jam - 2jam | Movie Official | 2016-12-02 | ???? |
| 5 | Horror | 10 - 15menit | Movie Official | 2016-12-04 | ???? |
| 7 | Komedi | <10 menit | Candra Liow | 2016-12-02 | ???? |
| 8 | Musik | <10 menit | Agnez Mo | 2016-12-02 | ???? |
| 11 | Horror | 15 menit - 1 jam | Movie Official | 2016-12-01 | ???? |
| 12 | Musik | <10 menit | Tulus | 2016-12-01 | ???? |
| 13 | Horror | 15 menit - 1 jam | Top Movie | 2016-12-02 | ???? |
| 14 | Musik | 10 - 15menit | Agnez Mo | 2016-12-01 | ???? |
| 15 | Musik | 10 - 15menit | Project Pop | 2016-12-04 | ???? |
| 16 | Komedi | 10 - 15menit | Movie Official | 2016-12-01 | ???? |

*Figure 4. Example of data which is used to calculated*

P(Kategori = “Horror” | Keterangan = “ditonton”)

= number of “Horror” which is stay in “ditonton” divided by number of “Horror” in training data.

= 2 / 4

= 0,5

P(Kategori = “Horror” | Keterangan = “tidak ditonton”)

= number of “Horror” which is stay in “tidak ditonton” divided by number of “Horror” in training data.

= 2 / 4

= 0,5

P( durasi = “1jam - 2jam” | keterangan = “ditonton” )

= 0 / 1

= 0

P( durasi = “1jam - 2jam” | keterangan = “tidak ditonton” )

= 1 / 1

= 1

P( youtuber account = “Movie Official” | keterangan = “ditonton” )

= 2 / 4

= 0,5

P(youtuber account = “Movie Official” | keterangan = “tidak ditonton” )

= 2 / 4

= 0,5

P( date upload = “2016-12-02” | keterangan = “ditonton” )

= 1 / 4

= 0,25

P( date upload = “2016-12-02” | keterangan = “tidak ditonton” )

= 3 / 4

= 0,75

4.1.4.3. Posterior Probability Calculation

Posterior probability is a probability of specific class when specific attribute appear. Posterior probability can be define as a result of multiplying each probability likehood in its class with its prior probability.

P( Kelas = “ditonton” | Atribut )
= { P( kategori = “Horror” | keterangan = “ditonton” ) X P( durasi = “1jam - 2jam” | keterangan = “ditonton” ) X P( youtuber account = “Movie Official” | keterangan = “ditonton” ) X P( date upload = “2016-12-02” | keterangan = “ditonton” ) } X P(kelas = “ditonton”)
= { 0,5 X 0 X 0,5 X 0,25 } X 0,6
= 0

P( Kelas = “tidak ditonton” | Atribut )
= { P( kategori = “Horror” | keterangan = “tidak ditonton” ) X P( durasi = “1jam - 2jam” | keterangan = “tidak ditonton” ) X P( youtuber account = “Movie Official” | keterangan = “tidak ditonton” ) X P( date upload = “2016-12-02” | keterangan = “tidak ditonton” ) } X P( keterangan = “tidak ditonton”)
= { 0,5 X 1 X 0,5 X 0,75 } X 0,4
= 0,1875 X 0,4
= 0,075

4.1.4.4. Maximum a Posteriori (MAP)

Maximum a posteriori is to find out which is the posterior probability with the maximum value. The video which can classified in “Recommended class” is a video which has a value of P( Kelas = “ditonton” | Atribut ) greater than value of P( Kelas = “tidak ditonton” | Atribut ). Because the result of P( Kelas = “ditonton” | Atribut ) in the example is smaller than P( Kelas = “tidak ditonton” | Atribut ), so that data is classified in “Not Recommended class”.

| ID | Kategori | Durasi | Owner | Date upload | Keterangan | P( Kelas = tidak ditonton \| Atribut ) | P( Kelas = ditonton \| Atribut ) |
|---|---|---|---|---|---|---|---|
| 4 | Horror | 1jam - 2jam | Movie Official | 2016-12-02 | Tidak direkomendasikan | 0.075 | 0 |
| 5 | Horror | 10 - 15menit | Movie Official | 2016-12-04 | direkomendasikan | 0.0125 | 0.05625 |
| 7 | Komedi | <10 menit | Candra Liow | 2016-12-02 | Tidak direkomendasikan | 0.1 | 0 |
| 8 | Musik | <10 menit | Agnez Mo | 2016-12-02 | Tidak direkomendasikan | 0.025 | 0.01875 |
| 11 | Horror | 15 menit - 1 jam | Movie Official | 2016-12-01 | direkomendasikan | 0 | 0.15 |
| 12 | Musik | <10 menit | Tulus | 2016-12-01 | direkomendasikan | 0 | 0.1 |
| 13 | Horror | 15 menit - 1 jam | Top Movie | 2016-12-02 | direkomendasikan | 0 | 0.075 |
| 14 | Musik | 10 - 15menit | Agnez Mo | 2016-12-01 | direkomendasikan | 0 | 0.1125 |
| 15 | Musik | 10 - 15menit | Project Pop | 2016-12-04 | direkomendasikan | 0 | 0.225 |
| 16 | Komedi | 10 - 15menit | Movie Official | 2016-12-01 | direkomendasikan | 0 | 0.1125 |

*Figure 5. The results of calculation with Naive Bayes algorithm*

### 4.1.5. Result

After getting the result, the recommended data will be sorted based on Bayesian result from the highest to the lowest and saved in the file text named recommendation.txt

| ID | Kategori | Durasi | Owner | Date upload | Keterangan | P( Kelas = tidak ditonton \| Atribut ) | P( Kelas = ditonton \| Atribut ) |
|---|---|---|---|---|---|---|---|
| 15 | Musik | 10 - 15menit | Project Pop | 2016-12-04 | direkomendasikan | 0 | 0.225 |
| 11 | Horror | 15 menit - 1 jam | Movie Official | 2016-12-01 | direkomendasikan | 0 | 0.15 |
| 14 | Musik | 10 - 15menit | Agnez Mo | 2016-12-01 | direkomendasikan | 0 | 0.1125 |
| 16 | Komedi | 10 - 15menit | Movie Official | 2016-12-01 | direkomendasikan | 0 | 0.1125 |
| 12 | Musik | <10 menit | Tulus | 2016-12-01 | direkomendasikan | 0 | 0.1 |
| 13 | Horror | 15 menit - 1 jam | Top Movie | 2016-12-02 | direkomendasikan | 0 | 0.075 |
| 5 | Horror | 10 - 15menit | Movie Official | 2016-12-04 | direkomendasikan | 0.0125 | 0.05625 |

*Figure 6. Sorting result*

## 4.2. Design

### 4.2.1. Flowchart

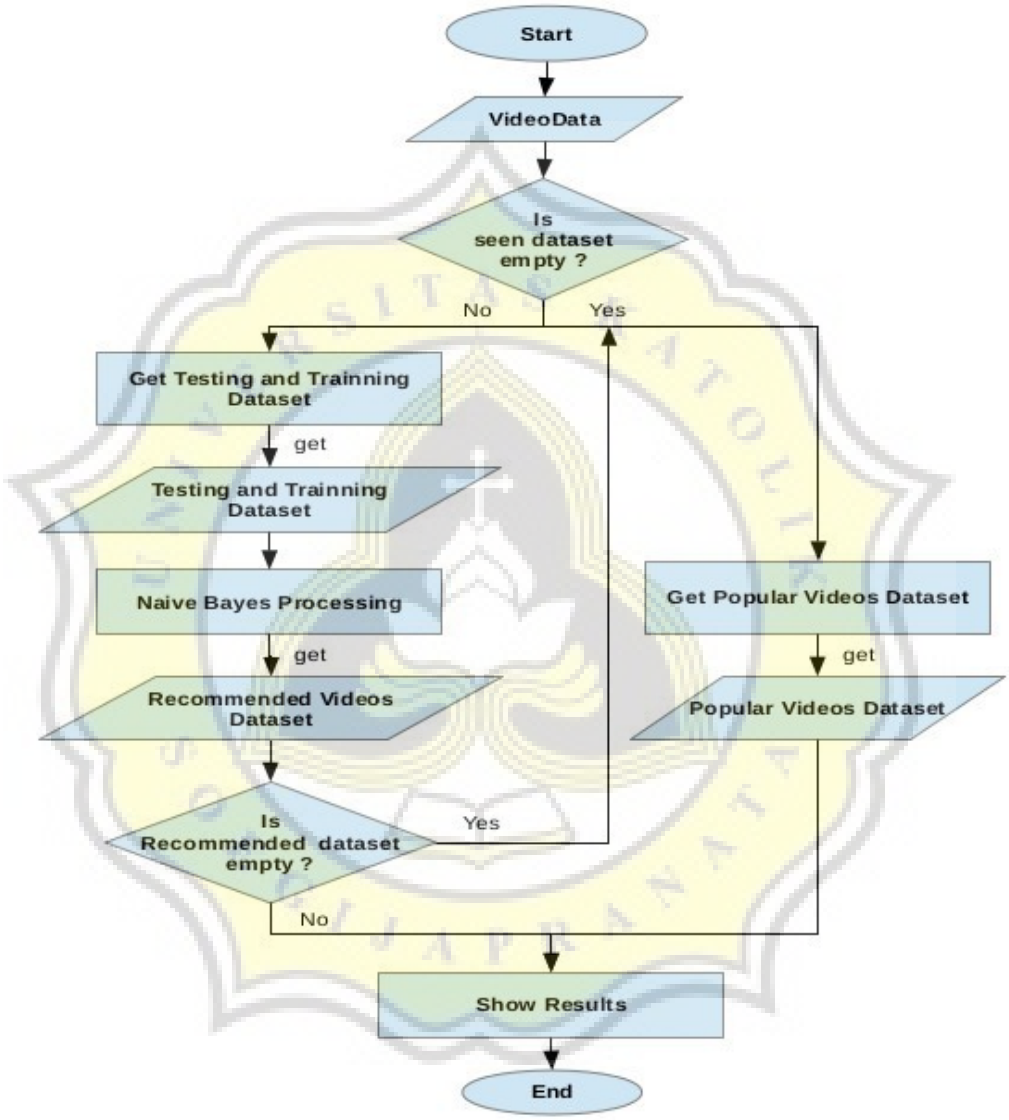


*Figure 7. Flowchart of program*

The figure above is about flowchart of program. Input in this program is a dataset of video which is gotten by using YouTube API. Then, program will check seen dataset. If it is not empty, program will run a function to get testing and training dataset. Then, testing and training

dataset will be processed with Naive Bayes algorithm. If the result is not empty, program will display it as recommended videos. If the seen or the result dataset is empty, program will run a function to get popular videos then display it as a recommendation.

### 4.2.2. Output Program

The output of the program is a list of videos that recommended for user. It will be saved in "Recommendation.txt", then displayed in the interface.

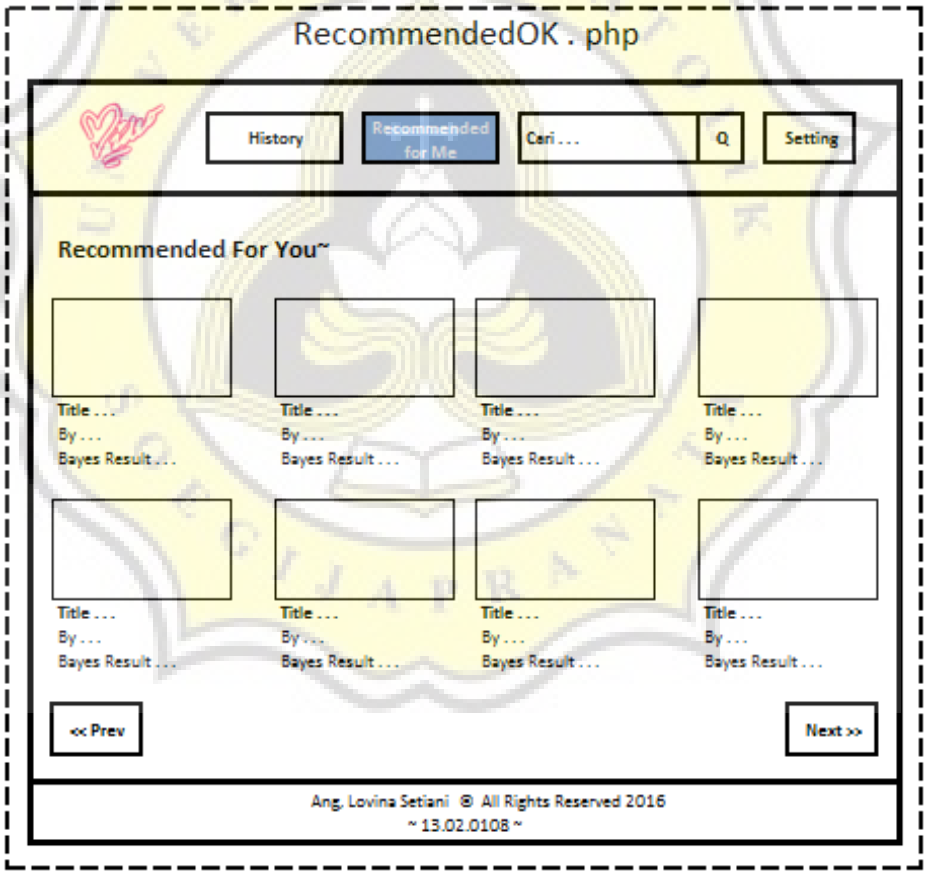


*Figure 8. Mock up of Recommendation page*