Detecting Clickbait Titles in YouTube Indonesia Using Naïve Bayes Classifier

Jonathan Gunawan
Faculty of Life Science, Human Computer Interaction
Surya University
Tangerang, Indonesia
jonathan.gunawan14@student.surya.ac.id

Zahra Putri Agusta
Faculty of Life Science, Human Computer Interaction
Surya University
Tangerang, Indonesia
Zahra.putri@surya.ac.id

***Abstract* – Many content creator on YouTube rely heavily from the revenues from their videos, due to the high amount of content creators on YouTube makes the competition even tighter. To attract the viewers to watch their videos, they need to make some catchy titles for the videos. Which are also known as 'Clickbait'. Currently the use of clickbait in YouTube Indonesia has become more common. Viewers are often disappointed with videos that do not meet their expectations, eventually they will only waste their time and resources. Machine learning is an appropriate method for solving this problem. Therefore, in this work Naïve Bayes classifier is used to detect clickbait title on YouTube Indonesia. Naïve Bayes classifier will predict the outcome of each words from the title whether those words are clickbait or not. Besides the machine learning method, preprocessing also needed in this research to process the raw data before it will be trained with the machine learning. With those method applied on this study, we can achieve recall to classify clickbait title up to 91%. This study was conducted with a hope to reduce the amount of clickbait titles in YouTube in Indonesia.**

Keywords – Naïve Bayes, machine learning, NLP, detect, YouTube, clickbait

I. INTRODUCTION

Prior to the abundance of internet use, informations were delivered with television, newspaper, radio, etc. At this time, the advancement of internet technology have a very important role in the distribution of information. Alongside with that, everybody can easily share information via internet.

YouTube has become a huge online media platform, many people share information in this platform. According to article from Kompas.com, YouTube ranks first on the most active social media platforms, with 43% of social media user in Indonesia[1].

With the high numbers of YouTube users in Indonesia, it also includes the content creators who uploads their video to YouTube. But, the amount of content creator on YouTube makes the competition on this platform tougher. Content creators finds using clickbait is effective method to attract viewers to watch their videos [2]. By using an ambiguous and persuasive words to their videos, this will increase the viewer's curiosity. Thus, the video will recieve more clicks from the viewers. Clickbait is an internet term, it is (On the Internet) a content whose main purpose is to attract attention and encourage visitors to click on a link to a particular web page[1].

However, the high amount of clickbait or misleading title can cause some problems to the content creator itself and to the audience such as:

a. To content creator:
- Damage the content creator credibility
- Negative response from the audience
- Worthless content

b. To audience:
- Not finding the information that you looking for
- Wasting time and resource
- Hoax news if you not careful

Therefore, a system is needed to detect the clickbait automatically. Machine learning is a suitable method to solve this problem. There are some previous study that explore machine learning to solve a similar problem.

In a previous research that has been done entitled "We used Neural Networks to Detect Clickbaits: You won't believe what happened Next!" (Ankesh Anand, Tanmoy Chakraorty, Noseong Park) that use recurrent neural network to detect the clickbait. [3]

Another research was "Detecting Clickbait in Online Social Media: You Won't Believe How We Did It?" (Aviad Elyashar, Jorge Bendahan, Rami Puzis) That using several machine learning those are ROC & AUC, random forest, and decision tree to detect clickbait title in online social media. [4]

The other research that using Naïve Bayes, neural network, and SVM (Support Vector Machine), to detect fake news "Detecting Fake News with Machine Learning Method" (Supanya Aphiwongsophon, Prabhas Chongstitvatana). [5]

Other research that propose new model for clickbait detection using machine learning "Clickbait Detection" (Martin Potthast, Sebastian Köpsel, Benno Stein, Mattias Hagen) [6]

Research entitled "Stop Clickbait: Detecting Clickbait and Preventing Clickbaits in Online News Media" (Abhijnan Chakraborty, Bhargavi Paranjape, Sourya Kakarla, Niloy

[1] https://en.oxforddictionaries.com/definition/clickbait

ISBN 978-602-53524-0-9

Ganguly) is using SVM, decision tree, and random forest to detect clickbait. [7]

Based on previous research described above, Naïve Bayes classifier is used to detect clickbait. This research explores new implementation from previous study into Indonesian language. Text data is taken from the title of the Indonesian YouTube video. So the biggest challenge is to preprocess the data that will be used for the learning in the Naïve Bayes method. Accordingly, in this paper we using 4 preprocessing approaches, namely (Case folding, Stop words, Word stemming, tokenizing) to the data.

In the next section we will explain the materials and proposed methods in this study, the implementation, the results and the analysis. At the end of this paper, we draw conclusion and analyze based on the results. Then we also provide recommendations for the future research.

II. METHOD

This study was conducted using several methods to achieve the goal. Such as, Naïve Bayes classifier to train the data, some preprocessing method, and Confusion matrix to evaluate the system.

A. Naïve Bayes Classifier

Naïve Bayes classifier is Bayes theorem applied with strong independent assumptions and do not have any correlation between them [8].

Each token is counted according to each quality label. That number will be used to calculate each word probability of being clickbait or not.

To train the data Naïve Bayes classifier is used to calculate the probability of the title. The calculation will use the Naïve Bayes posterior formula (1). In this section we will recall the formula from [9].

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \quad (1)$$

$P(c|x)$ = Posterior probability of class given predictor
$P(c)$ = Prior probability of class
$P(x|c)$ = Posterior probability of predictor given class
$P(x)$ = Prior probability of predictor

This work uses a customized Naïve Bayes classifier formula according to the input. The title input is represented as a list of string $[S_1, S_2, \ldots, S_n]$. To detect the quality of the input so the following formula will be used (2).

$$P(Q|S_i) = \frac{P(Q)\prod_{i=1}^{n} P(S_i|Q)}{Evidence} \quad (2)$$

$P(Q)$ = Prior probability of quality (bait/normal)
$P(S_i|Q)$ = Posterior probability of tokens i given quality
$Evidence = P(Q_B)\prod_{i=1}^{n} P(S_i|Q_B) + P(Q_N)\prod_{i=1}^{n} P(S_i|Q_N)$
$Q_{B/N}$ = Quality (Bait/Normal)

B. Preprocessing Data

The first step needed to be done is case folding. In this phase all inputs are changed to lowercase, this step is necessary because all inputs are needed to be in similar conditions as the dictionary that will be used on next step.

After that, the next preprocessing method is Stop word. In this phase the words are filtered from to be and conjunction. In this case it will use Indonesian language, the example words that are discarded are as follows: “yang”, “dan”, “itu”, “ini”, etc[10]. All of these words will be included into the dictionary file.

Next step is removing affixes and suffixes from the words such as “me-“, “di-“, “-an”, “-kan”, etc [11]. This method is called word Stemming [12]. Those method are required to reduce and minimalize the noise from our data and remove the unnecessary information. Which makes the computer recognize the words more easily and makes the training results more accurate.

After cleaning the data from noises, the titles are separated into corpora(tokens). The tokens are put into a list according into the quality of the title and labeled (bait/normal).

C. Evaluation Measurement

There are evaluations to check the F-measure, precision, recall and accuracy. Which involve TP (True Positive), TN (True Negative), FP (False Positive), FN (False Negative) [13].

TP is the number of tokens that are classify correctly as bait, and TN is the number of tokens that are correctly classify as normal (not clickbait).

FP is the number of tokens that are classify incorrectly as bait, and FN is the number of tokens that are incorrectly classify as normal.

The evaluations are calculated by formula as equation (3) to (6). This section will recall the formula from [14] and [15].

$$F - Measure = \frac{2*Precision*Recall}{Precision + Recall} \quad (3)$$

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (6)$$

F-Measure (F-Score) is the test’s accuracy score. Precision is the rate of correct positive (true positive) classification from cases that are predicted as positive. Recall is the rate of correct positive classification from cases that actually positive.

ISBN 978-602-53524-0-9

Accuracy is the rate of correct classification from overall number of cases.

III. IMPLEMENTATION

To implement all methods mentioned above, there are several steps taken which will be discussed in this section. In this work there are 2 programs, the first one is to train the data and the second one to predict the clickbait title. Overall steps can be seen in Figure 1 and will be explained thoroughly.

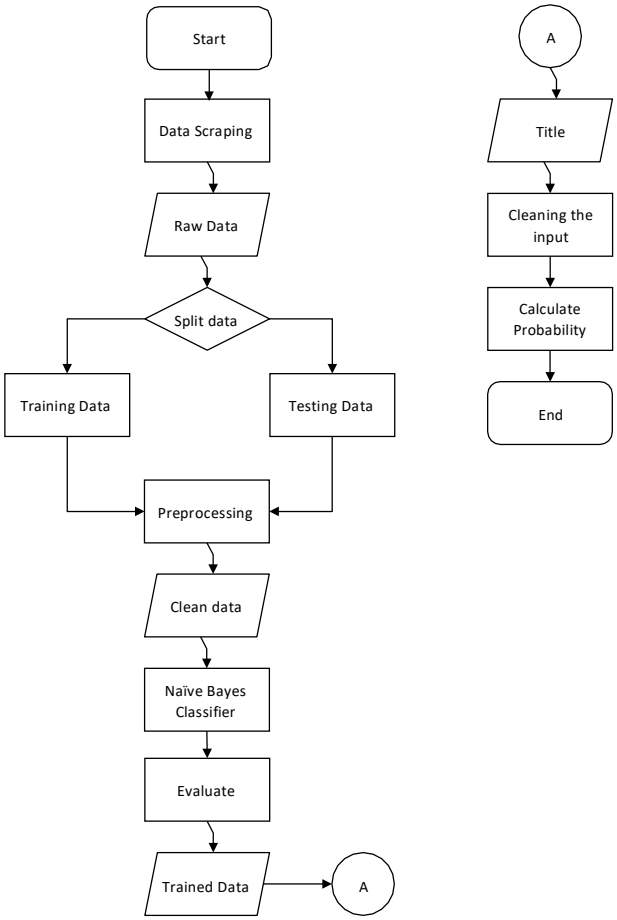

Figure 1. Flowchart of Research Processes

This study was done using 580 data samples that contains 280 clickbait titles and 300 normal titles. The data taken manually from YouTube website (Data scraping) with some criteria that has been fulfilled. According into previous study there are some feature from clickbait that distinguish it from ordinary titles [7].

The data then separated into two groups, the data with clickbait titles and the data with normal titles. Which also separated into two groups, training data and testing data. Training data and testing data are divided with 5 ratio, those are 50%:50%, 60%:40%, 70%:30%, 80%:20%, 90%:10%. The splitting was done with 5 ratio to find the ratio with the most efficient and the highest results. The raw data then preprocessed to clean the data.

After that, the data then trained with Naïve Bayes classifier. The training results 2 file containing vocabulary and the training results value.

To know the measurement of the training performance rate, the program also evaluate the training with confusion matrix. These steps are done on the first program.

After the data are trained, the trained data are going to be used on the second program. This program is going to predict the input title whether it is clickbait or not. The input title also needs to be cleaned with preprocessing, this is because the input needed to be at the same conditions as the trained data

The program will check each token from the input whether the token is in vocabulary file or not. After that it will calculate the probability of the title whether it is clickbait title or a decent title.

IV. RESULT AND ANALYSIS

The testing is done for each ratio. The testing begins on splitting the data training into 5 category, 50%:50%; 60%:40%; 70%:30%; 80%:20%; 90%:10%. Each category is trained and then evaluated. The scores of the evaluation can be seen in table 1 to table 5

Table 1. Evaluation Measure 50%:50% Ratio

Quality	F-Score	Precision	Recall	Accuracy
Bait	0.782419	0.682680	0.916289	0.697651
Normal	0.504672	0.756177	0.378712	

In table 1 it can be seen the result of classification evaluation. Every bait tokens are correctly identified with 91% recall, and the tokens classification are likely to be correct with precision about 68%. The normal tokens are incorrectly classified with 37% recall, and the classification are likely to be correct with precision about 75%. All of the classification are correctly classified with accuracy 69%.

Table 2.Evaluation Measure 60%:40% Ratio

Quality	F-Score	Precision	Recall	Accuracy
Bait	0.788338	0.694860	0.910877	0.710548
Normal	0.542350	0.764814	0.420142	

In table 2 the overall accuracy is increased into 71%. The same as the accuracy, the precision of each quality are also increased into 69% for clickbait and 76% for normal title classification. However the clickbait recall is slightly decreased, but the recall for the normal title increase to 42%. Overall the results on 60%:40% is slightly better than the 50%:50% ratio.

Table 3. Evaluation Measure 70%:30% Ratio

Quality	F-Score	Precision	Recall	Accuracy
Bait	0.795742	0.706395	0.910965	0.725522
Normal	0.581426	0.785067	0.462050	

The same as table 2, in table 3 the evaluation measurement results are keep increasing than from the previous ratio. With the overall accuracy reach 72%, precision 70% for the clickbait

title and 78% for the normal title classification, and the recall for both quality.

Table 4. Evaluation Measure 80%:20% Ratio

Quality	F-Score	Precision	Recall	Accuracy
Bait	0.799754	0.716171	0.905424	0.736245
Normal	0.613744	0.792048	0.500967	

The results in table 4 are even better, with the highest overall accuracy it can reach 73%, precision 71% for the clickbait title and 79% for normal title classification. The recall for clickbait title are slightly decreased into 90%, however the recall for normal title are increased into 50%.

Table 5. Evaluation Measure 90%:10% Ratio

Quality	F-Score	Precision	Recall	Accuracy
Bait	0.799001	0.720720	0.896358	0.732113
Normal	0.598503	0.764331	0.491980	

Lastly in table 5, the results are slightly decreased. With overall accuracy, recall, and normal precision decreased compared to 80%:20% ratio.

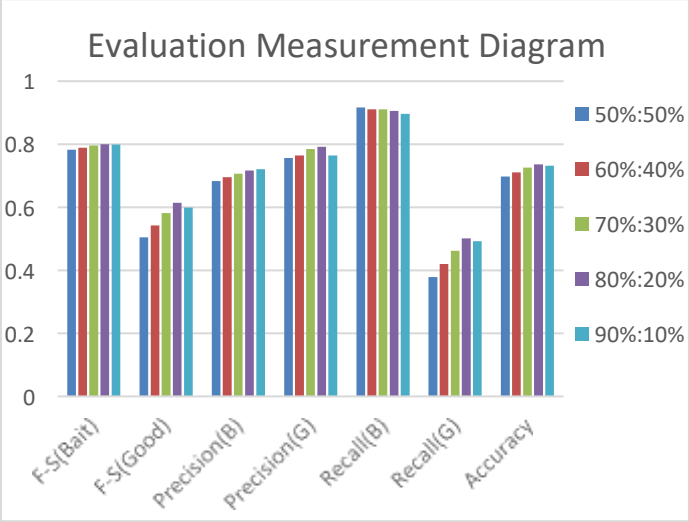

Figure 2. Evaluation Measurement Results Diagram

This research calculates the probability of a title whether it is clickbait or not by using Naïve Bayes classifier. The lowest training accuracy is 0.697651 and the highest is 0.736245. The results can be influenced with the amount of data and the preprocessing phase.

According to the chart above, 50%:50% ratio gain the lowest accuracy and the accuracy is increasing until the 80%:20% ratio. However, there was no significant difference from the 70%:30% ratio to the 80%:20% ratio, and on the 90%:10% ratio the accuracy is decreasing.

Even though the results from the 90%:10% ratio are not significantly decreased compare to 70%:30% and 80%:20%, the low testing data can lead the training results not reliable. This is because the 10% testing data are not enough to represent the training data.

The evaluation measure also depends on the ratio of the training data and the testing data. More training data will produce higher evaluation measure results. However, less testing data will make the predicting program with low credibility. Thus, 7:3 to 8:2 ratio are the suitable ratio for training data and the testing data.

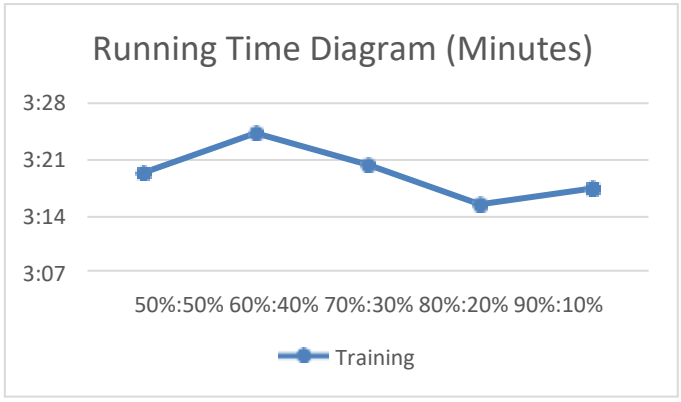

Figure 3. Running Time Each Scenario Diagram

The training duration from each ratio are not significantly different. In this scenario we can assume the ratio of training data and the testing data do not affect the duration of training process.

From the results of training process, we use the data to detect clickbait title. In this process we insert the title to be verified. The output from this process are the probability of that title is a clickbait title or a normal title.

```
Input title to verify: JANGAN BERSIUL DI DANAU KALIMANTAN!!! Ini Yang Akan Terjadi Jika Mencoba
Cleaned and tokenized title: ['jangan', 'siul', 'danau', 'kalimantan', 'akan', 'jadi', 'coba']

Result:
Bait: 0.9988561263976152
Good: 0.0011438736023842557
```

Figure 4. Predicting the System with Clickbait Title Example

```
Input title to verify: Advokat Indonesia Laporkan Hanum Rais ke PDGI Terkait Kasus Hoaks Ratna Sarumpaet
Cleaned and tokenized title: ['advokat', 'indonesia', 'lapor', 'hanum', 'rais', 'ke', 'pdgi', 'kait', 'kasus', 'hoaks', 'ratna', 'sarumpaet']

Result:
Bait: 1.0115241320569132e-07
Good: 0.9999998988475874
```

Figure 5. Predicting the System with Normal Title Example

CONCLUSION

Naïve Bayes classifier is an efficient and effective classifier despite its simplicity. However, naïve Bayes classifier is rely heavily on the amount of the data. If the amount of data are pretty low, then the evaluation measure will remain low. On the other hand, more data are going to make the training process longer and using more resources.

High amount of testing data with the fairly low training data makes the learning process harder for the system. Resulting low accuracy, recall, and precision. However the low amount of testing data makes the trained data less reliable because the

testing data does not adequately represent the data. Future researches and more data collecting can be done to increase the evaluation measures and the program credibility.

REFERENCES

[1] K. P. Wahyunanda, "Riset ungkap pola pemakaian medsos orang Indonesia", https://tekno.kompas.com/read/2018/03/01/10340027/riset-ungkap-pola-pemakaian-medsos-orang-indonesia. Accessed : 4 October 2018.

[2] H. Nathan, "To clickbait or not to clickbait? An examination of clickbait effects on source credibility", University of Missouri-Columbia, 2016.

[3] A. Ankesh, C. Tanmoy, and P. Noseong, "We used neural networks to detect clickbaits: you won't believe what happened next!," European Conference on Information Retrieval, 2016.

[4] B. Jorge, E. Aviad, and P. Rami, "Detecting clickbait in online social media: you won't believe how we did it?", 2017.

[5] A. Supanya, and C. Prabhas, "Detecting fake news with machine learning method," 2018.

[6] H. Matthias, K. Sebastian, P. Martin, and S. Benno, "Clickbait Detection," Bauhaus-Universität Weimar, 2016.

[7] C. Abhijnan, G. Niloy, K. Sourya, and P. Bhargavi, "Stop clickbait: detecting and preventing clickbaits in online news media," IEEE/ACM international conference on ASONAM, 2016.

[8] N. Sairam, and S. Karthika, "A Naïve Bayes for educational qualification," Indian Journal of Science and Technology, 2015.

[9] M. Musa, and T. Sona, "Learning the naïve Bayes classifier with optimization models," International Journal AMCS, 2013.

[10] N. Dinda, "Analisis penggunaan konjungsi pada buku teks catatan peristiwa sejarah indonesia SMA/MA kelas X karangan Matroji, terbutan Bumi Aksara, tahun 2014," Jurnal ilmiah pendidikan bahasa dan sastra Indonesia , 2017.

[11] Defina, "Materi imbuhan bahasa Indonesia dalam BPPB pada tingkat keterampilan berbahasa mahasiswa asing," Jurnal pendidikan bahasa dan sastra, 2017.

[12] K. Nilay, and S. R. Brajendra, "A survey of stemming algorithms for information retrieval," IOSR Journal of Computer Engineering, 2015.

[13] J. R. Maria Navin, and R. Pankaja, "Performance analysis of text classification algorithms using confusion matrix," International Journal of Engineering and Technical Research, 2016.

[14] R. D. Prakash, and S. K. Rajani, "Efficient classification technique for outlier detection," International Journal of Coumputer Science and Mobile Computing, 2015.

[15] M. Hossin, and M. N. Sulaiman, "A review on evaluation metrics for data classification evaluation," International Journal of Data Mining & Knowledge Management Process, 2015.