FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY

BITI 2513

INTRODUCTION TO DATA SCIENCE

CAPSTONE PROJECT

PROJECT TITLE : FAKE AND REAL NEWS CLASSIFIER

TASK 3 : MODELLING AND EVALUATION

GROUP : DS GENG

Name:	No Matric :
Muhammad Naim Syahmi bin Roslan	B031810312
Muhamed Hussain Bin Hithayatullah	B031810392
Muhammad Fikrun Amin	B031810404
Ramanan Gobalakrishnan	B031810334

Lecturer Name : AP DR SHARIFAH SAKINAH SYED AHMAD

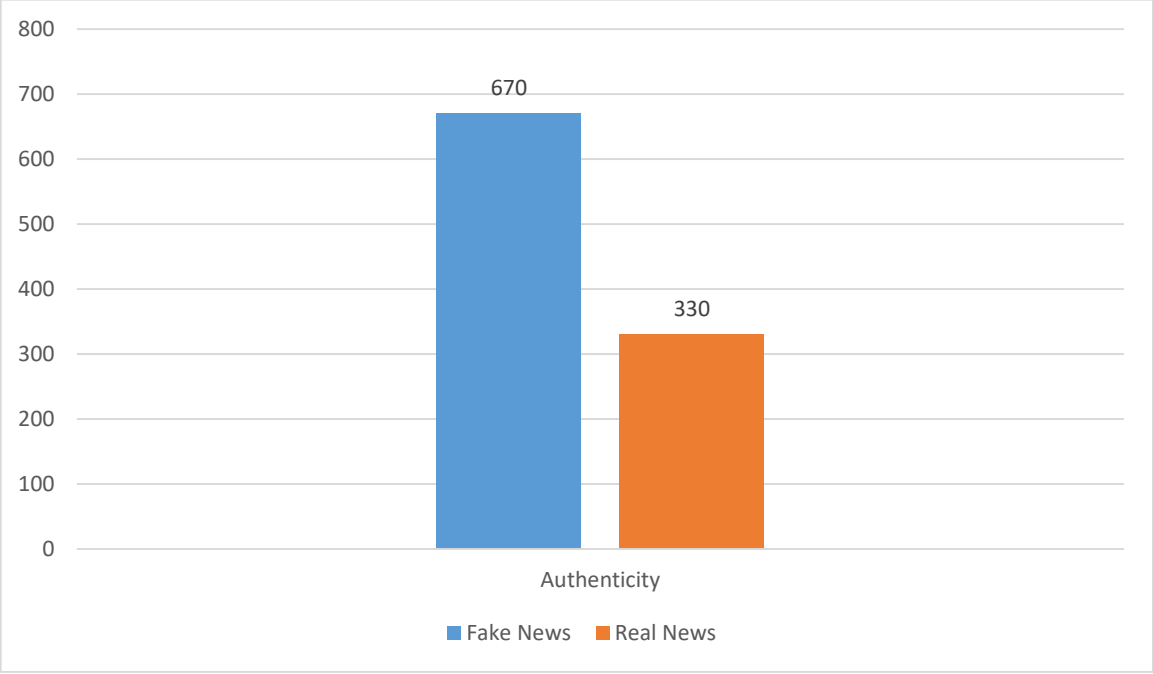
TABLE OF CONTENTS

1. Modeling Preparation

In this task 3, before we do modeling our data, we labeled again our data, but this time we did it manually. Previously, in task 2, we did data labeling automatically using a python library that is TextBlob to get the value of polarity of a tweet. And then, the tweet that had polarity value greater than 0 was labeled as a real news and otherwise it was labeled as a fake news. After that, we found that our data had a lot of label errors, for example something fake news labeled real automatically and vice versa.

Our goal of doing data labeling manually is to make sure the model we will get is reliable and shows the real results. We did labeling manually 1 by 1 according to the criteria that we have agreed together in the team. The following criteria are criteria that consider a tweet as a fake news:

1. The authenticity of the hoax from the news contained in the tweet;
2. The news source in the tweet is unclear or damaged;
3. Non-fundamental statements;
4. Opinion (satire, provocation, question, feeling, and not relevant to Covid-19);
5. Unclear language (not written in English).



As a result of our manual labeling of our data, we found out that our data are not balanced where the real news is only 330, while fake news is 670. The histogram above is a visualization of comparison between the two labels, fake news and real news.

2. Algorithms To Be Used

A. Decision Tree

- Decision tree can used for both regression and classification. It also a type of supervised learning algorithm. We use decision tree because to help us make better decision. For our project we use decision tree model as one of the model to predict whether the tweet about covid 19 is fake or real. The pros of decision trees model is easy to understand and visualise even for people from non-analytic background. Decision tree also can perform both numerical and categorical variable. The cons of Decision tree is, it would be overfitting when the attribute is sparse and it will perform poorly on testing data.

B. Logistic Regression

- Another algorithm that we used for this project is Logistic Regression. It is a supervised learning algorithm used for classification problem. We use Logistic Regression because our target label only have 2 possible classes. In other words, Our target label is dichotomous and the value is either 1(stand for real tweet) or 0(stand for fake tweet). The advantage using Logistic Regression for our project is the algorithm can tell us the influence of the attributes of the data on the target label. However the disadvantage of Logistic Regression is it doesn't work well if the predicted variable is not binary.

C. Support vector machine (SVM)

- Support vector machine is one of the algorithms which we have used in this project. It is a supervised learning algorithm which is used mainly for classification as it uses data points to help with the classification. SVM is used to separate the two classes of data points in this case real tweets and fake tweets, there are many possible hyperplanes that could be chosen. Our objective is to find a plane that has the maximum margin between data points of both classes which allows us to classify whether the tweet is either fake or real. We used

SVM for our project as works relatively well when there is clear margin of separation between classes. Also, SVM also has stability as a small change to the data does not greatly affect the hyperplane. Furthermore, SVM has L2 Regularization feature. So, it has good generalization capabilities which prevent it from over-fitting. However, the disadvantages of using SVM are it requires extensive memory requirement as the algorithmic complexity and memory requirements of SVM are very high and it also requires long training time.

D. Random Forest

- Random forest a supervised learning algorithm which is used for classification or regression by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. In simpler words, random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction. The advantages of random forest is that it reduces overfitting problem in decision trees and also reduces the variance and therefore improves the accuracy. Furthermore, random forest can automatically handle missing values and no feature scaling required in case of Random Forest as it uses rule based approach instead of distance calculation. Other than that random forest is also very stable and is less impacted by noise. The disadvantages of Random Forest are it is very complex as it creates a lot of trees and combines their outputs so it requires much more computational power and resources. Moreover, Random Forest require much more time to train as compared to decision trees.

3. Performance Metrics

An important aspect of any Machine Learning or Data Science task that include training a model is to evaluate the model. The model can be evaluated by performance metrics. Data Scientist and Machine Learning enthusiasts commonly use not one, but a series of performance metrics to determine the performance of a trained model. However, not all performance metrics are suitable for each model. The performance metrics used must be carefully selected based on the problem at hand. This to ensure the results are fair and the performance of the model can be read accurately. For example, the accuracy performance metric is not commonly used to evaluate a model trained for a regression problem. Similarly, the root mean squared error (RMSE) is not used for a model trained for a classification task.

Regarding the project, the performance metrics used to evaluate the 4 models are performance metrics most often used for binary classification. They are mostly based on the values of the confusion matrix. The performance measures are accuracy, F-measure, specificity, recall, and precision. The Performance Measures Table illustrates the focus of each performance measure and their respective formula [1].

Performance Measures Table

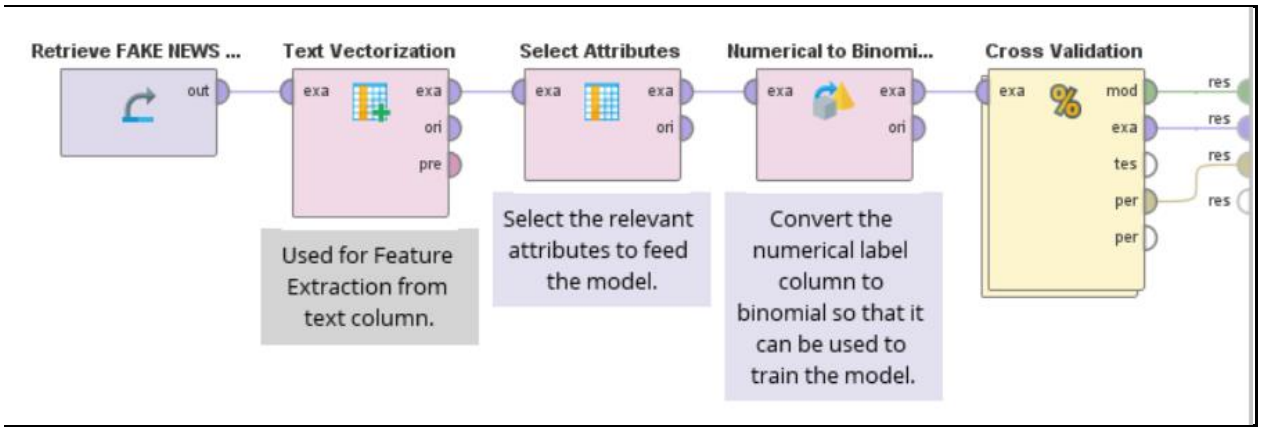
Performance Measures	Focus	Formula
Accuracy	Overall effectiveness of the classifier	$\frac{tp + tn}{tp + fn + fp + tn}$
F-Measure	Relations between positive labels and those given by the classifier	$\frac{(\beta^2 + 1)tp}{(\beta^2 + 1)tp + \beta^2 fn + fp}$
Specificity	How effectively the classifier identifies negative labels.	$\frac{tn}{fp + tn}$
Recall	Effectiveness of a classifier to identify positive labels	$\frac{tp}{tp + fn}$
Precision	Class agreement of the data labels with positive labels given by the classifier	$\frac{tp}{tp + fp}$

Note. Adapted from "A Systematic Analysis of Performance Measures for Classification Tasks", by Sokolova, M., Lapalme, G., 2009, *Information Processing and Management, 45,* p. 427.

4. Model Implementation and Results

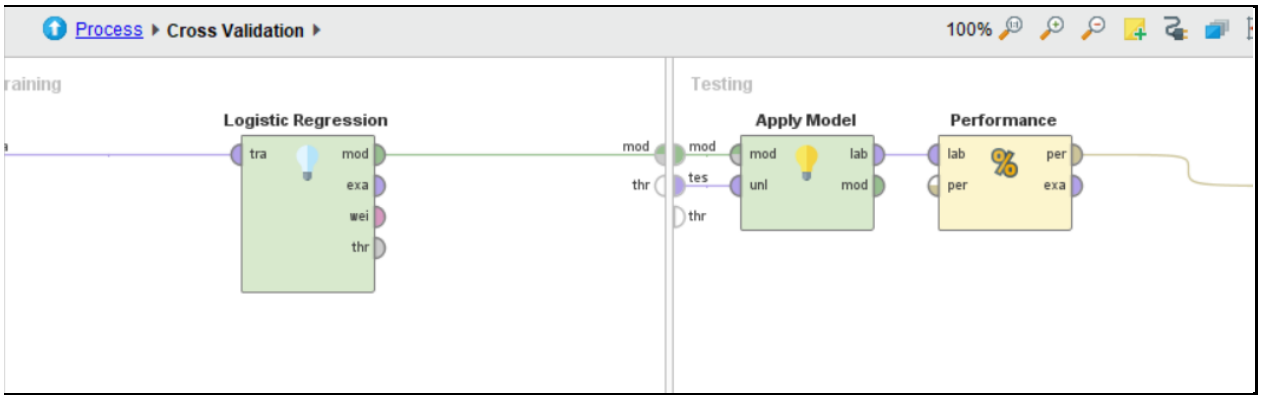
A. Logistic Regression

The first model is Logistic Regression, for this model we use RapidMiner's Logistic Regression operator to train the model.



Operators involved in the Logistic Regression Process.

In RapidMiner, we first import the labelled data and set the 'authenticity' column as the target label. Next, we apply the Text Vectorization operator with the number of max features set to 277 so that the model can use our text column for training. Next, we only pass the relevant attributes to the model using the Select Attributes operator and convert the target label column from numerical data type to binomial data type using the Numerical to Binomial operator.



Inside the Cross Validation operator for Logistic Regression

For training and validation purposes, we use the Cross Validation operator with the number of folds set to 10. Then, we set the Logistic Regression operator in the training section of the operator and the Apply Model and Performance operator for the testing section of the operator.

accuracy: 66.20% +/- 3.61% (micro average: 66.20%)

	true false	true true	class precision
pred. false	504	172	74.56%
pred. true	166	158	48.77%
class recall	75.22%	47.88%	

Accuracy of Logistic Regression

After running the process and obtaining the results, we observe the various performance measures that we have chosen. The first performance measure is accuracy, in the logistic regression, the accuracy obtained is 66.20%

precision: 49.04% +/- 5.32% (micro average: 48.77%) (positive class: true)

	true false	true true	class precision
pred. false	504	172	74.56%
pred. true	166	158	48.77%
class recall	75.22%	47.88%	

Precision of Logistic Regression

Based on the image shown, the precision that we obtained for the logistic regression model is 49.04%

recall: 47.88% +/- 7.11% (micro average: 47.88%) (positive class: true)

	true false	true true	class precision
pred. false	504	172	74.56%
pred. true	166	158	48.77%
class recall	75.22%	47.88%	

Recall of Logistic Regression

For this performance measure, our logistic regression model has obtained a recall of 47.88%, which is the highest recall obtained among all the four models trained.

f_measure: 48.23% +/- 5.06% (micro average: 48.32%) (positive class: true)

	true false	true true	class precision
pred. false	504	172	74.56%
pred. true	166	158	48.77%
class recall	75.22%	47.88%	

F-Measure of Logistic Regression

Based on the image shown, the F-Measure obtained for logistic regression is 48.23%. This value is the highest F-Measure value obtained among all 4 models.

specificity: 75.22% +/- 5.27% (micro average: 75.22%) (positive class: true)

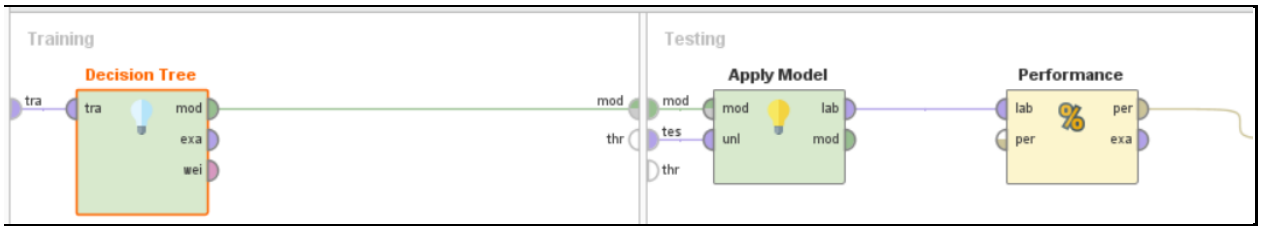
	true false	true true	class precision
pred. false	504	172	74.56%
pred. true	166	158	48.77%
class recall	75.22%	47.88%	

Specificity of Logistic Regression

For the specificity performance measure, the value we obtained of logistic regression is 75.22%

B. Decision Tree

The second model is Decision tree. We use RapidMiner software to train the model. For this model, we choose decision tree operator.



Inside the Cross Validation operator for Decision Tree

For training and validation, we also use cross validation and we put Decision Tree operator with the default parameter settings as the main model to use for training.

accuracy: 68.30% +/- 2.31% (micro average: 68.30%)

	true false	true true	class precision
pred. false	638	285	69.12%
pred. true	32	45	58.44%
class recall	95.22%	13.64%	

Accuracy of Decision Tree

After performing the decision tree process, The image above showed us the result of accuracy for the decision tree process. We received the accuracy for the decision tree is 68.30%.

precision: 61.38% +/- 19.72% (micro average: 58.44%) (positive class: true)

	true false	true true	class precision
pred. false	638	285	69.12%
pred. true	32	45	58.44%
class recall	95.22%	13.64%	

Precision of Decision Tree

Based on our data, The image above showed us the result of precision for the decision tree process. We get the precision is 61.38%.

recall: 13.64% +/- 4.10% (micro average: 13.64%) (positive class: true)

	true false	true true	class precision
pred. false	638	285	69.12%
pred. true	32	45	58.44%
class recall	95.22%	13.64%	

Recall of Decision Tree

The image above, showed us based on our data, the train model received 13.64% for recall performance measure.

f_measure: 21.96% +/- 5.99% (micro average: 22.11%) (positive class: true)

	true false	true true	class precision
pred. false	638	285	69.12%
pred. true	32	45	58.44%
class recall	95.22%	13.64%	

F-Measure of Decision Tree

The image above showed us the result of accuracy for the decision tree process. We received the F-Measure for decision tree is 68.30%.

specificity: 95.22% +/- 2.97% (micro average: 95.22%) (positive class: true)

	true false	true true	class precision
pred. false	638	285	69.12%
pred. true	32	45	58.44%
class recall	95.22%	13.64%	

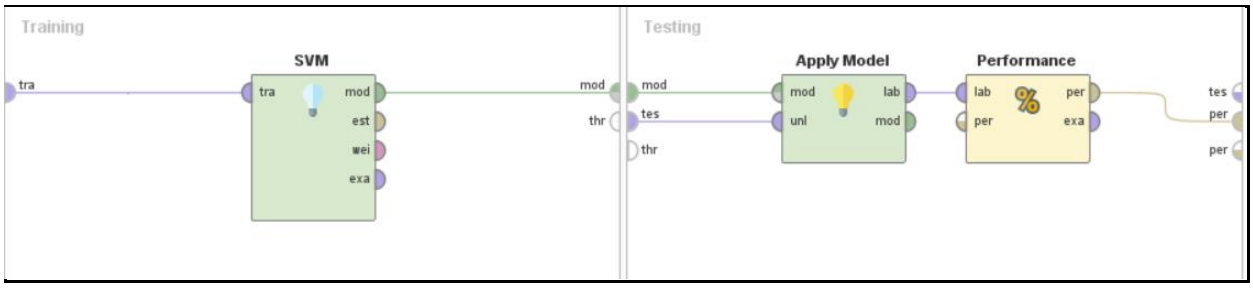
,

Specificity of Decision Tree

Based on our data, The image above showed us the result of specificity for the decision tree process. We get the specificity is 61.38%.

C. Support Vector Machine (SVM)

The third model that we have is Support Vector Machine (SVM). We use RapidMiner software to train the model. For this model, we choose the SVM operator.



Inside the Cross-Validation operator for SVM

For training and validation, we also use cross validation and we put SVM operator with the default parameter settings as the main model to use for training. After running the process, we obtain the following results:

accuracy: 69.10% +/- 3.00% (micro average: 69.10%)

	true false	true true	class precision
pred. false	624	263	70.35%
pred. true	46	67	59.29%
class recall	93.13%	20.30%	

Accuracy of SVM

The first performance measure is accuracy. The value obtained from the model is 69.10%.

precision: 60.43% +/- 14.46% (micro average: 59.29%) (positive class: true)

	true false	true true	class precision
pred. false	624	263	70.35%
pred. true	46	67	59.29%
class recall	93.13%	20.30%	

Precision of SVM

The second performance measure is precision and the value we obtained from the model is 60.43%.

recall: 20.30% +/- 5.35% (micro average: 20.30%) (positive class: true)

	true false	true true	class precision
pred. false	624	263	70.35%
pred. true	46	67	59.29%
class recall	93.13%	20.30%	

Recall of SVM

The third performance measure is recall and the value obtained from the model is 20.30%. This is because there is more fake news than real news in our data which is an imbalance in data.

f_measure: 30.06% +/- 7.35% (micro average: 30.25%) (positive class: true)

	true false	true true	class precision
pred. false	624	263	70.35%
pred. true	46	67	59.29%
class recall	93.13%	20.30%	

F-measure of SVM

The third performance measure is F-measure and the value we obtained for SVM is 30.06%. Which is low because it considers the percentage of both Recall and Precision.

specificity: 93.13% +/- 3.74% (micro average: 93.13%) (positive class: true)

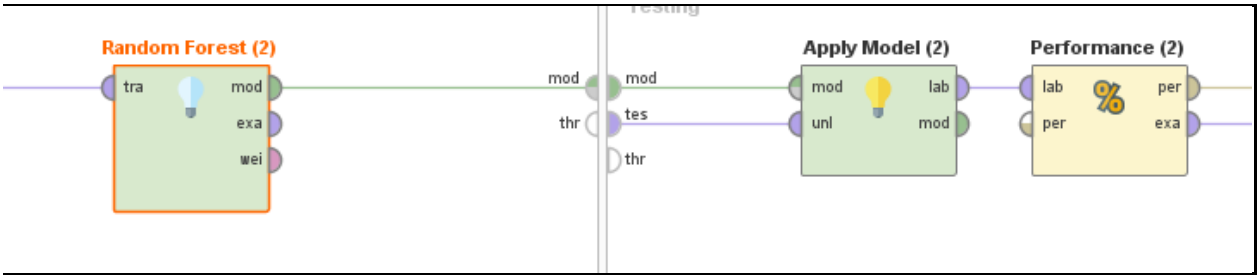
	true false	true true	class precision
pred. false	624	263	70.35%
pred. true	46	67	59.29%
class recall	93.13%	20.30%	

Specificity of SVM

The fourth performance measure is Specificity and the value obtained from it is 93.13%. This means this model has a good ability to predict truly fake news.

D. Random Forest

The fourth model that we have is Random Forest. We use RapidMiner software to train the model. For this model, we choose the Random Forest operator.



Inside the Cross-Validation operator for Random Forest

For training and validation, we also use cross-validation and we put the Random Forest operator with the default parameter settings as the main model to use for training. After running the process, we obtain the following results:

accuracy: 69.50% +/- 1.08% (micro average: 69.50%)

	true 0	true 1	class precision
pred. 0	668	303	68.80%
pred. 1	2	27	93.10%
class recall	99.70%	8.18%	

Accuracy of Random Forest

We obtain the first performance measure that is accuracy. The image above shows that the model using the Random Forest algorithm successfully got 69.50% for the accuracy. This accuracy is the highest accuracy among the 3 other models.

precision: 95.50% +/- 9.56% (micro average: 93.10%) (positive class: 1)

	true 0	true 1	class precision
pred. 0	668	303	68.80%
pred. 1	2	27	93.10%
class recall	99.70%	8.18%	

Precision of Random Forest

For the second performance, based on the image above we gain the Precision percentage of the model using Random Forest that is 95.5%. And again, this precision is the highest precision compared to the 3 other models.

recall: 8.18% +/- 3.51% (micro average: 8.18%) (positive class: 1)

	true 0	true 1	class precision
pred. 0	668	303	68.80%
pred. 1	2	27	93.10%
class recall	99.70%	8.18%	

Recall of Random Forest

The third performance is Recall. The percentage of recall is only 8.18%. The percentage of Recall is very low and extremely bad. This is caused by an imbalance in the data where the fake news data is more than the real news. This results in the weakness of this model to consider and predict real news.

f_measure: 14.85% +/- 5.99% (micro average: 15.04%) (positive class: 1)

	true 0	true 1	class precision
pred. 0	668	303	68.80%
pred. 1	2	27	93.10%
class recall	99.70%	8.18%	

F-measure of Random Forest

The fourth, the F-measure of this model is really low. The percentage of F-measure is only 14.85% based on the image above. The reason for this is a very bad percentage obtained in Recall. Because this F-measure considers the percentage of Precision and also Recall.

specificity: 99.70% +/- 0.63% (micro average: 99.70%) (positive class: 1)

	true 0	true 1	class precision
pred. 0	668	303	68.80%
pred. 1	2	27	93.10%
class recall	99.70%	8.18%	

Specificity of Random Forest

The fifth performance is Specificity. The specificity for this model is almost perfect that is 99.70%. This specificity is the highest specificity compared to the 3 other models. The good news behind the high percentage of specificity is that this model has an almost perfect ability to predict truly fake news. Specificity only considers fake news.

E. Best Algorithm

Model/ Performance Measure	Accuracy	Precision	Recall	F-Measure	Specificity
Logistic Regression	66.20%	49.04%	47.88%	48.23%	75.22%
Decision Tree	68.30%	61.38%	13.64%	21.96%	95.22%
SVM	69.10%	60.43%	20.3%	30.06%	93.13%
Random Forest	69.50%	95.5%	8.18%	14.85%	99.70%

A table of all results, the green cells indicate the best result obtained for that performance measure.

After we have implemented all the 4 models and run their processes in RapidMiner, we recorded the results obtained from each performance measure of each model. From the results, we found that the logistic regression model has the best f-measure and recall. However, the random forest algorithm has the most performance measures that are considered best. Random forest has the highest accuracy (69.50%), the best specificity (99.70%) and the most precision (95.5%). Therefore, we have concluded that the best algorithm for our project is the random forest algorithm.

5. **References**

1. B. Bhutani, N. Rastogi, P. Sehgal, and A. Purwar, "Fake News Detection Using Sentiment Analysis," 2019 Twelfth International Conference on Contemporary Computing (IC3), Noida, India, 2019, pp. 1-5, doi: 10.1109/IC3.2019.8844880.
2. Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. Information Processing & Management, 45(4), 427–437. https://doi.org/10.1016/j.ipm.2009.03.002.
3. Gupte, Amit, et al. "Comparative study of classification algorithms used in sentiment analysis." *International Journal of Computer Science and Information Technologies* 5.5 (2014): 6261-6264.
4. Poddar, Karishnu, and K. S. Umadevi. "Comparison of Various Machine Learning Models for Accurate Detection of Fake News." *2019 Innovations in Power and Advanced Computing Technologies (i-PACT)*. Vol. 1. IEEE, 2019.