Spam Classification

Jasmine Dogu, Christos Chen, Brian Wimmer

Meet the Team

Christos Chen

Jasmine Dogu

Brian Wimmer

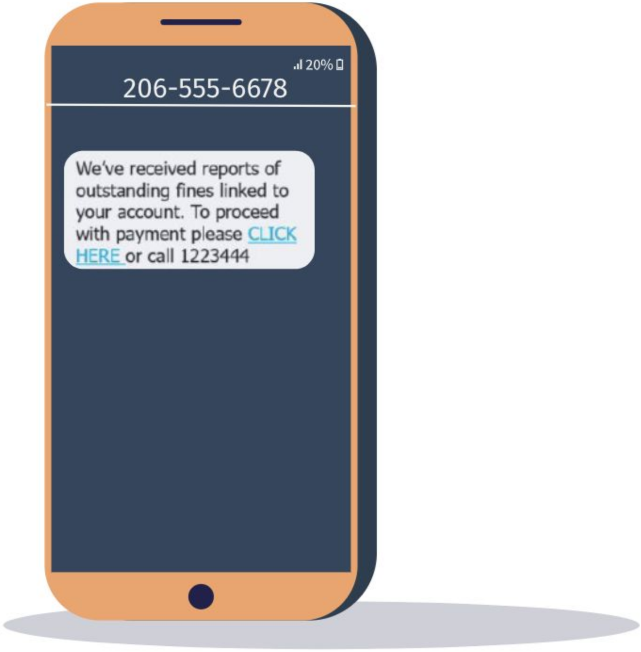
Table of Contents

Background

General Questions

1 Is there a Distinct Separation Between the Topics Found in **Spam** and **Ham** Text Messages?

2 Can we Predict whether a Text will be Considered **Spam** or **Ham**?

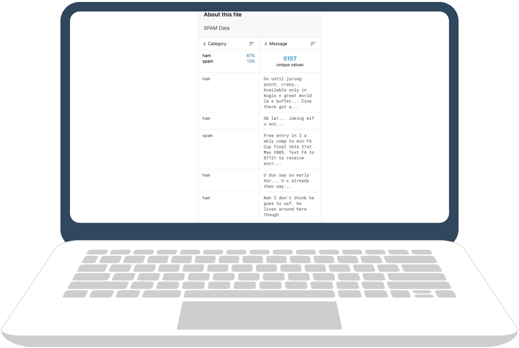


Data Set Information

Kaggle
"Spam Text Message Classification"

Label
Spam vs. Ham (Non-Spam)

Observations
5157 unique messages



Base Rate
13% spam
87% non-spam

Location
Singapore and UK

Columns
2 columns: category and message

Background Information

Prevalence

30 billion scam-messages per year.
Government regulations to prevent spam messages (CAN-SPAM)

Wireless Providers

Machine learning and artificial intelligence are utilized to filter out spam messages

Economic Loss

Est. 43 million Americans lost $10.5 billion to cellular-related scams (2018)

Hypotheses

Hypotheses

PRIMARY

- ***Null Hyp*:** The variation between spam and non-spam messages within the LDA Topic Model Gamma will not be statistically significant (alpha of 0.05)
- ***Alt Hyp*:** The variation between spam and non-spam messages within the LDA Topic Model Gamma will be statistically significant (alpha of 0.05)

**Will be utilizing a two sample t-test

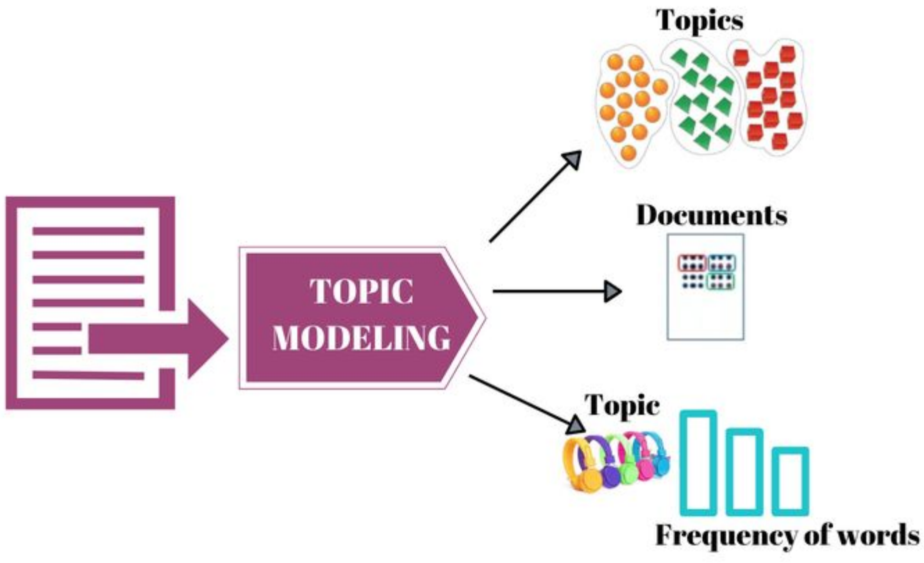
SECONDARY

- ***Null Hyp:*** A SVM Kernel Model will classify spam messages with at a 0.9 recall rate or less.
- ***Alt Hyp:*** A SVM Kernel Model will classify spam messages with a recall rate greater than 0.9.

Modeling

a) Topic Modeling

Topic Modeling



Latent Dirichlet Allocation

- Unsupervised machine learning algorithm, similar to clustering

Gamma

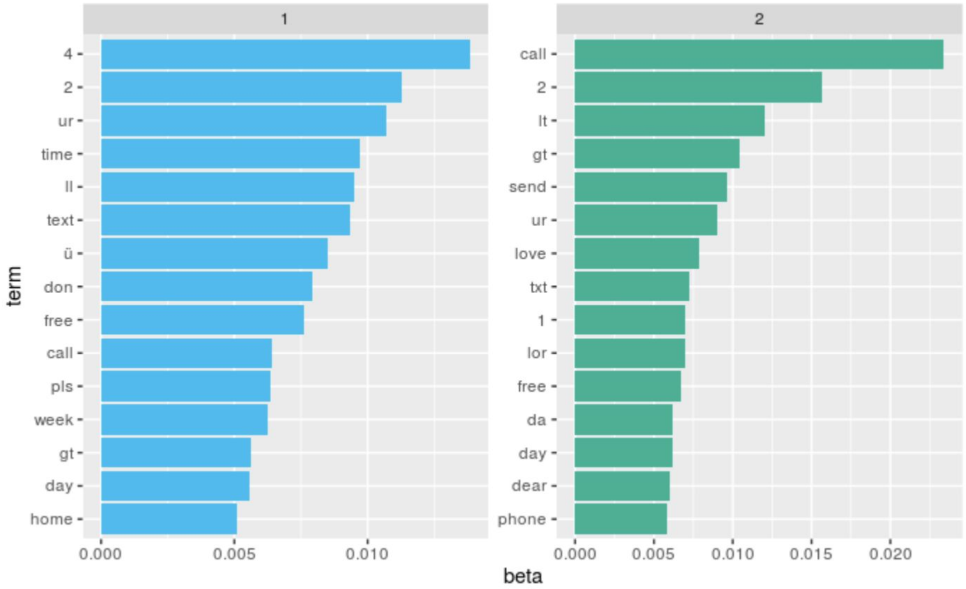
- Estimated proportion of words from a topic

Beta

- Density of words within a topic

Topic Modeling

Topic 1 Topic 2



Key Insight

Top Beta-valued words between the topics did not reveal an obvious classification of models

Topic 1

- More Structured
- Time, places, locations

Topic 2

- More Conversational
- Slightly more slang

Two Sample T-Test

Assumptions

- Independence
- Randomly Sampled from Population
- Data is Continuous
- Normal Distribution & Equal Variance

document	topic	gamma
1501	1	0.4789798
1596	1	0.4922286

document	topic	gamma
1501	2	0.5210202
1596	2	0.5077714
1928	2	0.4770852
3409	2	0.4978154
4205	2	0.4682499
5057	2	0.4736144

```
        Welch Two Sample t-test

data:  gammaValsStatsTopic1$gamma and gammaValsStatsTopic2$gamma
t = -0.072668, df = 60, p-value = 0.9423
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.005216226  0.004850516
sample estimates:
mean of x mean of y
0.4999086 0.5000914
```

P-Value = 0.9423

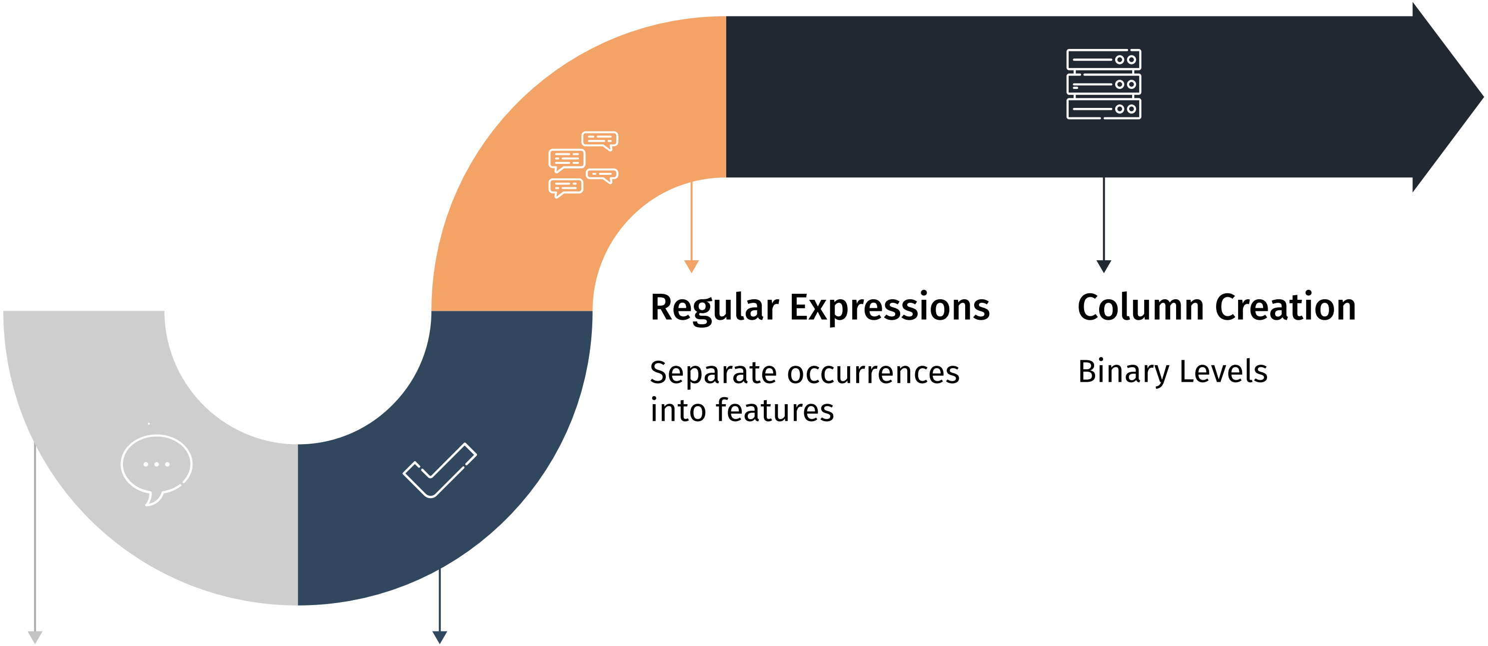
We Fail to Reject our **Primary** Null Hypothesis

The results are insignificant at $p > 0.05$

Modeling

b) Classification

Feature Engineering Steps



Regular Expressions

Separate occurrences into features

Column Creation

Binary Levels

Isolation

“Spam” messages

Commonalities

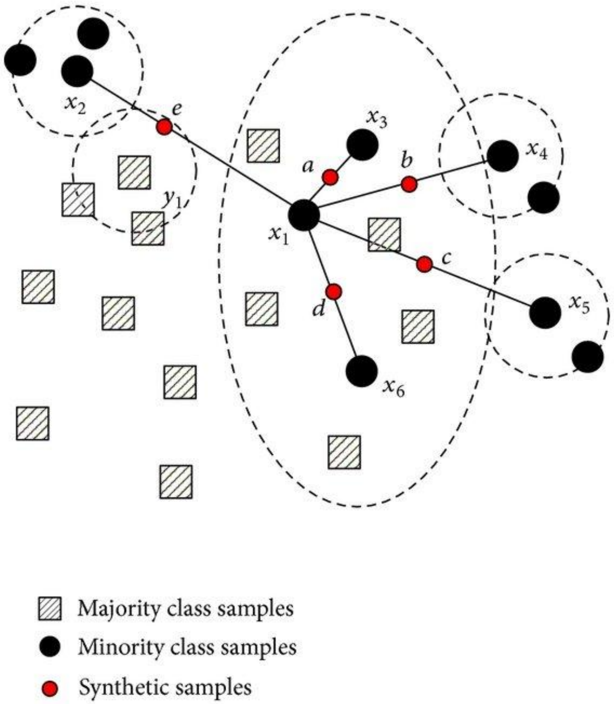
13 frequently used symbols/words/numbers

Feature Engineering Results

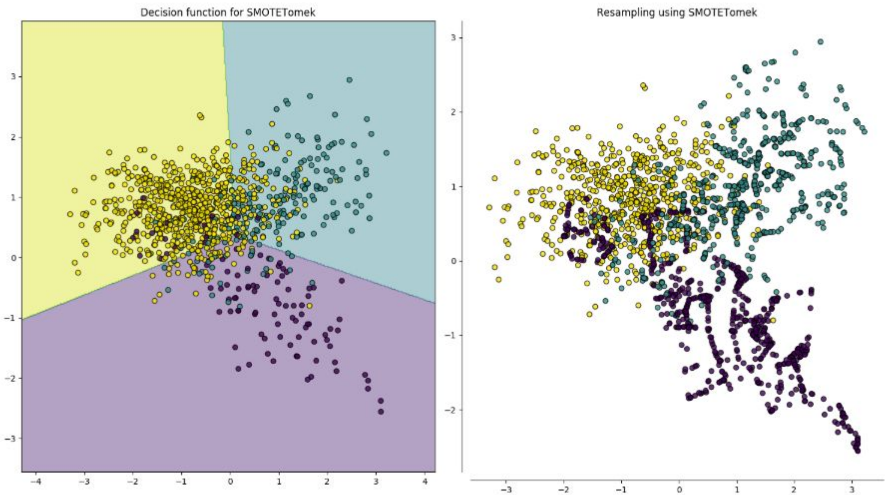
- Symbol - *1921*
- Reply_Yes - *1815*
- Call - 634
- Digits - 634
- Link - 567
- Free - 330
- Won_Win - 320
- Mobile_Phone - 291
- Currency_Symbol - 273
- Please - 137
- Eighteen - 132
- XXX - 63
- FreeMsg - 14

SMOTE - Synthetic Minority Oversampling Technique

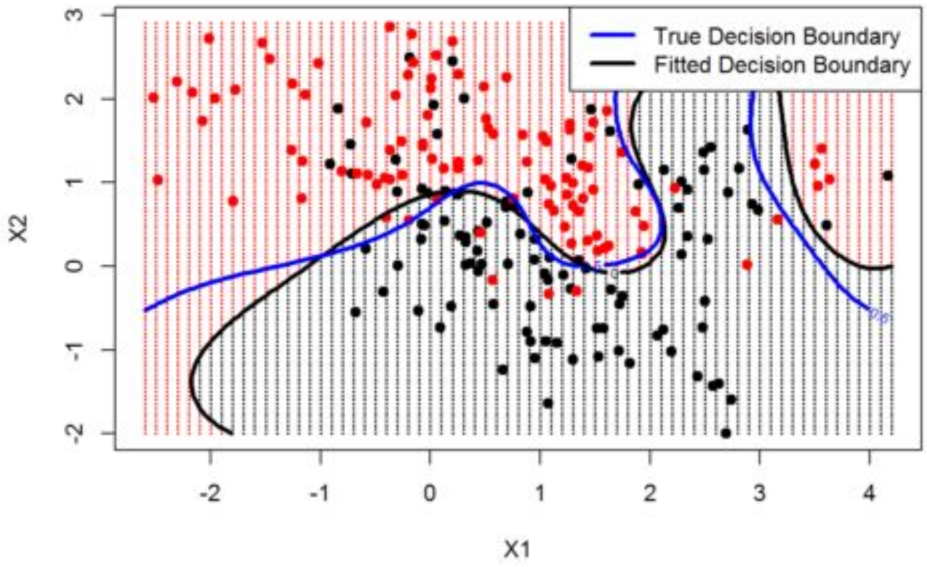
KNN



SVM



Radial Support Vector Machines



M.Rubin Julis, S.Alagesan. INTERNATIONAL JOURNAL OF SCIENTIFIC & TECHNOLOGY RESEARCH VOLUME 9, ISSUE 02, FEBRUARY 2020, Spam Detection In Sms Using Machine Learning Through Text Mining: p. 498-503

Support Vector Machine

	Model - SVM	Recall/Sensitivity	Accuracy	Kappa
ORIGINAL	**Untuned Model**	.9947	.9743	.8824
	Tuned Model *Gamma: 0.5 , Cost: 4*	.9950	.9815	.9172
SMOTE-KNN	**Untuned Model**	.9768	.9619	.9169
	Tuned Model *Gamma: 1 , Cost: 4*	.9840	.9712	.9371
SMOTE-SVM	**Untuned Model**	.9746	.9102	.8205
	Tuned Model *Gamma: 1, Cost: 4*	.9832	.9140	.8281

**Tuned models based on lowest error rate

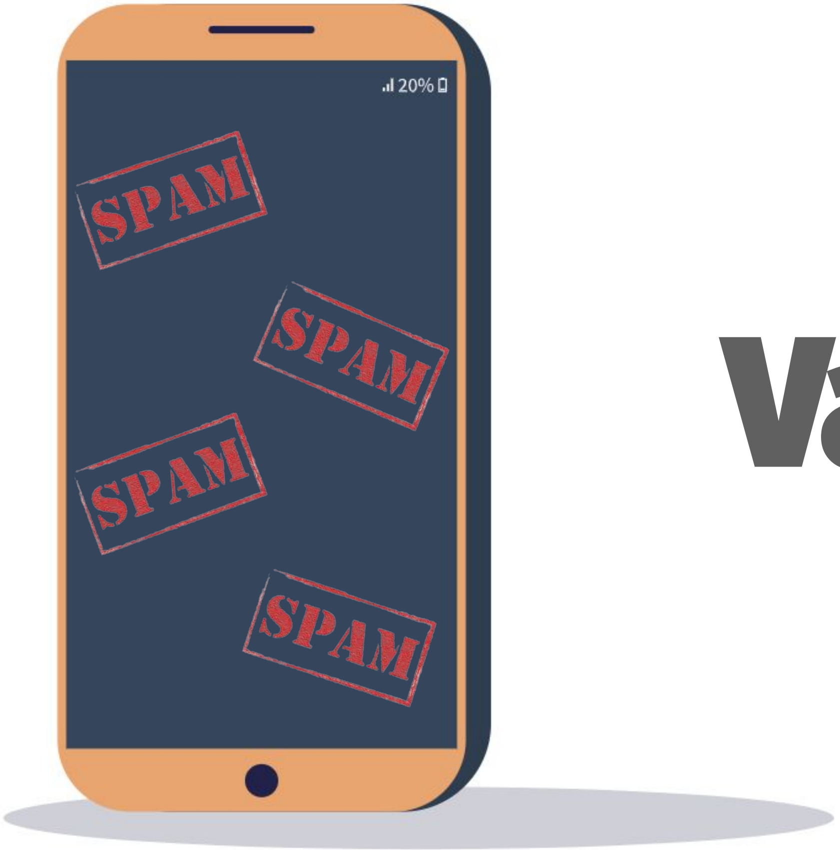
Optimal Classifier - Tuned SMOTE KNN

- ***Null Hyp:*** A SVM Kernel Model will classify spam messages with a recall of 0.9 or less.
- ***Alt Hyp:*** A SVM Kernel Model will classify spam messages with a recall rate greater than 0.9.

TUNED, SMOTE-KNN Support Vector Machine

Recall value of 98.40%

We reject the secondary null hypothesis.



Value

$624 million

In Cost Savings to Americans/year if a large cellular service provider began implementation

Cost Breakdown

Assumptions

- U.S. loss of *$10.5 billion* in 2018 from cellular-related spam messages
  - Average American lost $32.01/ year
- A message will be blocked by the cellular provider if it is believed to be spam
- Outreach 133 M American customers/year given that top 3 cellular service providers have an average of 133 M coverage/year
- Data is representative of current population

Cost Matrix

		Actual	
		Yes Scam	*Ham*
Predictions	*Yes Scam*	$ -	$ -
	Ham	$ 32.00	$ -

Population Matrix

Assuming population reach of **133,000,000** /year

		Actual	
		Yes Scam	*Ham*
Predictions	*Yes Scam*	19526753	1794630
	Ham	1064128	110614489
	People	**20,590,881**	**112,409,119**
	SMOTE Base Rate	0.15	0.85

*Full analysis on Appendix A

Limitations

Geographic Boundaries

Data from UK and Singapore. Text messages have variation around the world in syntax, morphology, slang, etc.

Generalizability

SMOTE Methods increases the likelihood of overfitting as it replicates the minority class events.

Time Sensitive

Spam messages, like all other scams, vary in common approaches over time.

SMOTE Documentation

Unclear documentation of how the SMOTE algorithm utilizing SVM works in R

Future Analysis

Other Types of Messages	Run similar modeling on spam/ham emails
Different Locations	Look at data solely from the United States
Utilize Other Features	Evaluate the messages differently and include different features in the model
Sampling Methods	Use different oversampling methods to see change in performance
Cost Consideration	Incorporate a cost matrix to inject bias into the model for max organizational benefit

THANK YOU!

Questions?

Citations

http://www.ijstr.org/final-print/feb2020/Spam-Detection-In-Sms-Using-Machine-Learning-Through-Text-Mining.pdf

https://rstudio.github.io/reticulate/articles/calling_python.html

https://escholarship.org/content/qt99x0w9w0/qt99x0w9w0_noSplash_6386a738c0e8b3d02aa47b6a4cda0b3f.pdf

https://hmjianggatech.github.io/files/BHAMProject/SentimentAnalysis.pdf

https://medium.com/analytics-vidhya/re-sampling-imbalanced-training-corpus-for-sentiment-analysis-c9dc97f9eae1

https://medium.com/analytics-vidhya/re-sampling-imbalanced-training-corpus-for-sentiment-analysis-c9dc97f9eae1

https://www.researchgate.net/publication/224600045_MASS_A_Malay_language_LVCSR_corpus_resource

https://towardsdatascience.com/how-to-handle-smote-data-in-imbalanced-classification-problems-cf4b86e8c6a1

https://www.securitymagazine.com/articles/90146-phone-scams-cause-americans-to-lose-105-billion-in-2018

https://www.ctia.org/news/protecting-consumers-by-stopping-text-messaging-spam

**Other Sources Used and Listed in R-Markdown

Appendix A

Cost Matrix

		Actual	
		Yes Scam	***Ham***
Predictions	***Yes Scam***	$ -	$ -
	Ham	$ 32.00	$ -

Population Matrix

Assuming population reach of **133,000,000** /year

		Actual	
		Yes Scam	***Ham***
Predictions	***Yes Scam***	19526753	1794630
	Ham	1064128	110614489

People	**20,590,881**	**112,409,119**
SMOTE Base Rate	0.15	0.85
SUM	20590881	112409119

Positive Pred Rates

		Actual	
		Yes Scam	***Ham***
Predictions	***Yes Scam***	0.948320413	0.01596517
	Ham	0.051679587	0.98403483
		1	1

Confusion Matrix

	Yes Scam	***Ham***
Yes Scam	367	11
Ham	20	678
SUM	387	689

COST SAVINGS

$ 624,856,084 / year

Cost Matrix relative to Status Quo = No model*

		Actual	
		Yes Scam	***Ham***
Predictions	***Yes Scam***	$ (32.00)	$ -
	Ham	$ -	$ -

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 367  11
         1  20 678

               Accuracy : 0.9712
                 95% CI : (0.9594, 0.9803)
    No Information Rate : 0.6403
    P-Value [Acc > NIR] : <2e-16

                  Kappa : 0.9371

 Mcnemar's Test P-Value : 0.1508

            Sensitivity : 0.9840
            Specificity : 0.9483
         Pos Pred Value : 0.9713
         Neg Pred Value : 0.9709
             Prevalence : 0.6403
         Detection Rate : 0.6301
   Detection Prevalence : 0.6487
      Balanced Accuracy : 0.9662

       'Positive' Class : 1
```