DIAMONDS

Service Like No Other

Showing insightful trends,

Emitting the beauty of data

How to use data to enhance the efficiency of customer service?

Key Features

Auto-generated Answers

SMART Priority Filtering

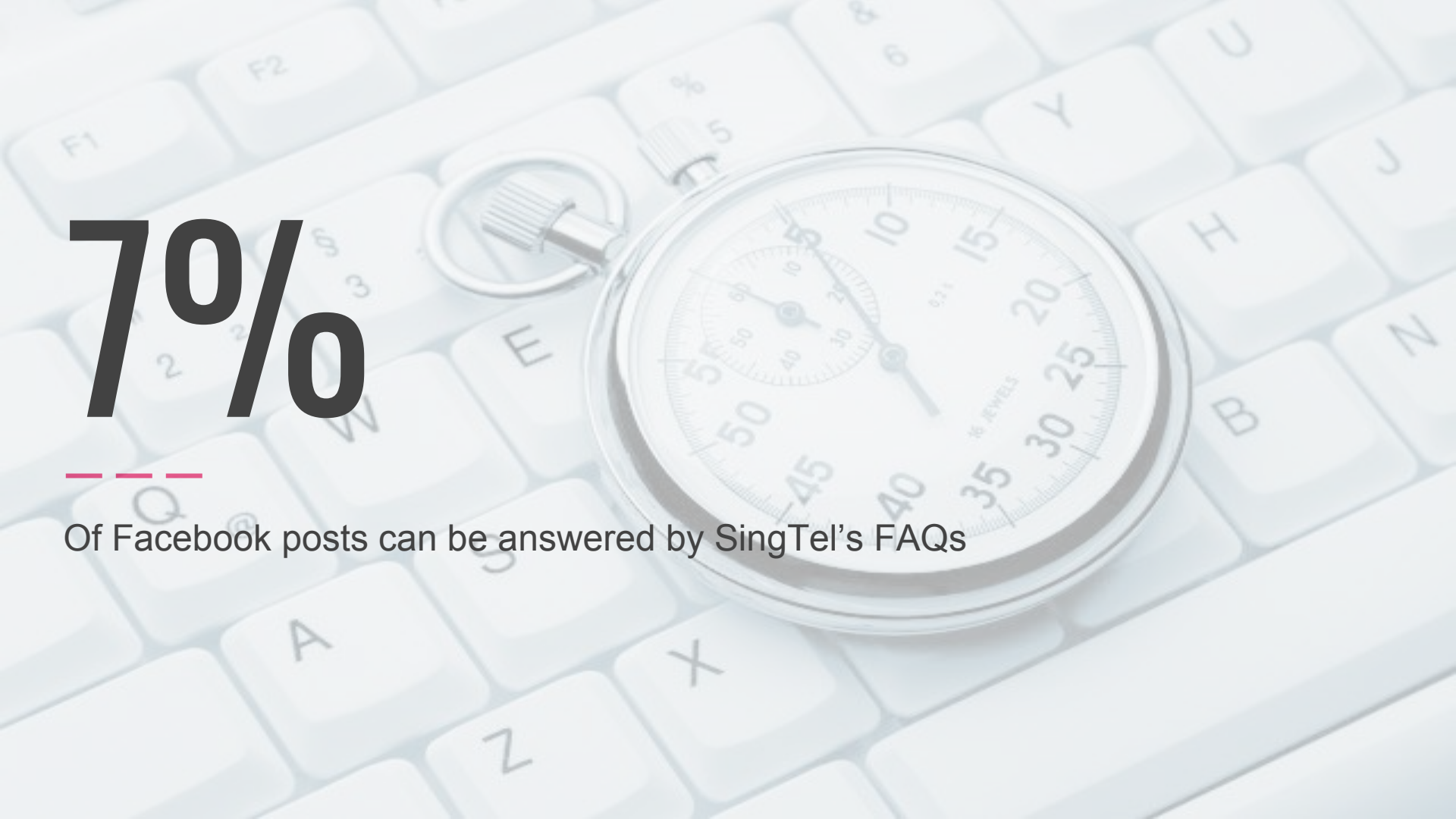
Interactive Dashboard

Key Features

Auto-generated Answers

SMART Priority Filtering

Interactive Dashboard



7%

Of Facebook posts can be answered by SingTel’s FAQs

Avg ~1000

Facebook posts per month

Avg ~70

Facebook posts per month which can be covered by FAQs

Avg ~840

Facebook posts per year which can be covered by FAQs

Key Features

Auto-generated Answers

SMART Priority Filtering

Interactive Dashboard

Priority Filters

- Category (e.g. tech support, request, complain, enquiry, compliment, others)
- Junk/Spam
- Priority Score

Priority Algorithm

Following the guidelines from ITIL (Information Technology Infrastructure Library), our tickets are prioritized based on **urgency** and **business impact**.

	Tech Support	Request	Complaint	Enquiry	Compliment	Suggestion/ Others
Negative	Highest	Highest	High	High	Medium	Low
Neutral	High	High	Medium	Medium	Low	Low
Positive	Medium	Medium	Medium	Low	Lowest	Lowest

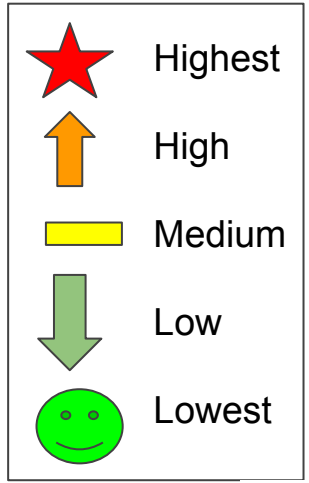


Table: Priority Matrix

Priority Algorithm

Score Range

Category		Label		Sentiment	
Tech Support	4	Broadband	2	Negative	1.5
Request	3	Lifestyle Services	1.3	Neutral	1.2
Complaint	3	Mobile	2	Positive	1
Enquiry	2	Others	1		
Compliment	1	Telephone Services	1.5		
Suggestion/Others	1				

Priority Algorithm

$$Score = Category + Sentiment * Label + Num\_of\_Days\_Since\_Posted*0.1$$

Key Features

Auto-generated Answers

SMART Priority Filtering

Interactive Dashboard

Key Parties

Entry Level

Management Level

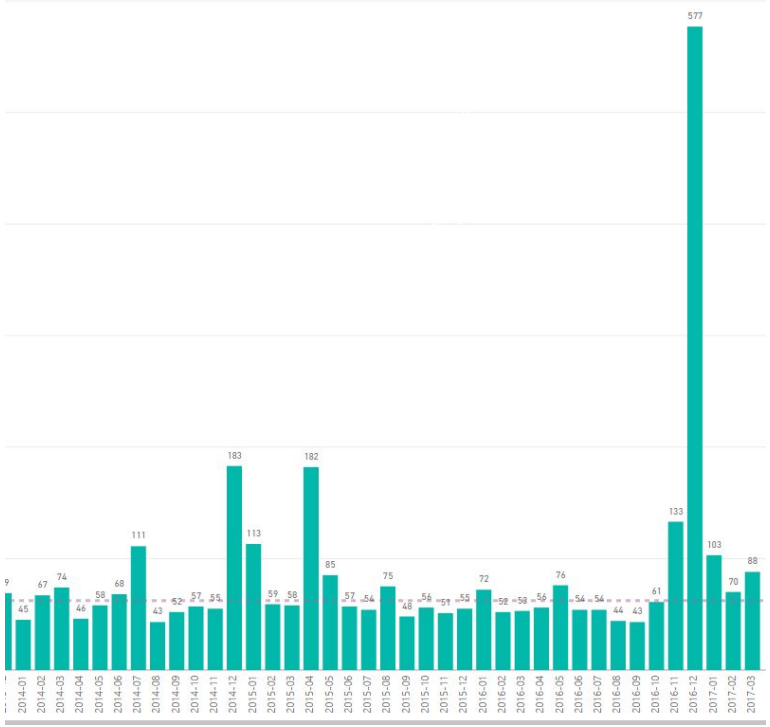
The Solution

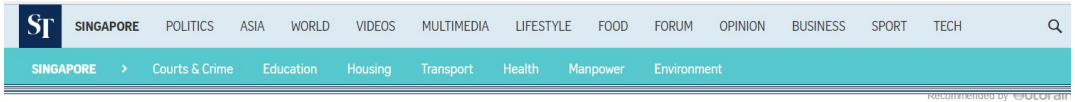
Entry Level Dashboard DEMO

Management Level Dashboard DEMO

Insights from Data Analysis

1) 2 main spikes in #posts - Apr 2015 and Dec 2016
2) More than 500% increase in number of posts compared to average
    a) Significantly higher number of *broadband* posts

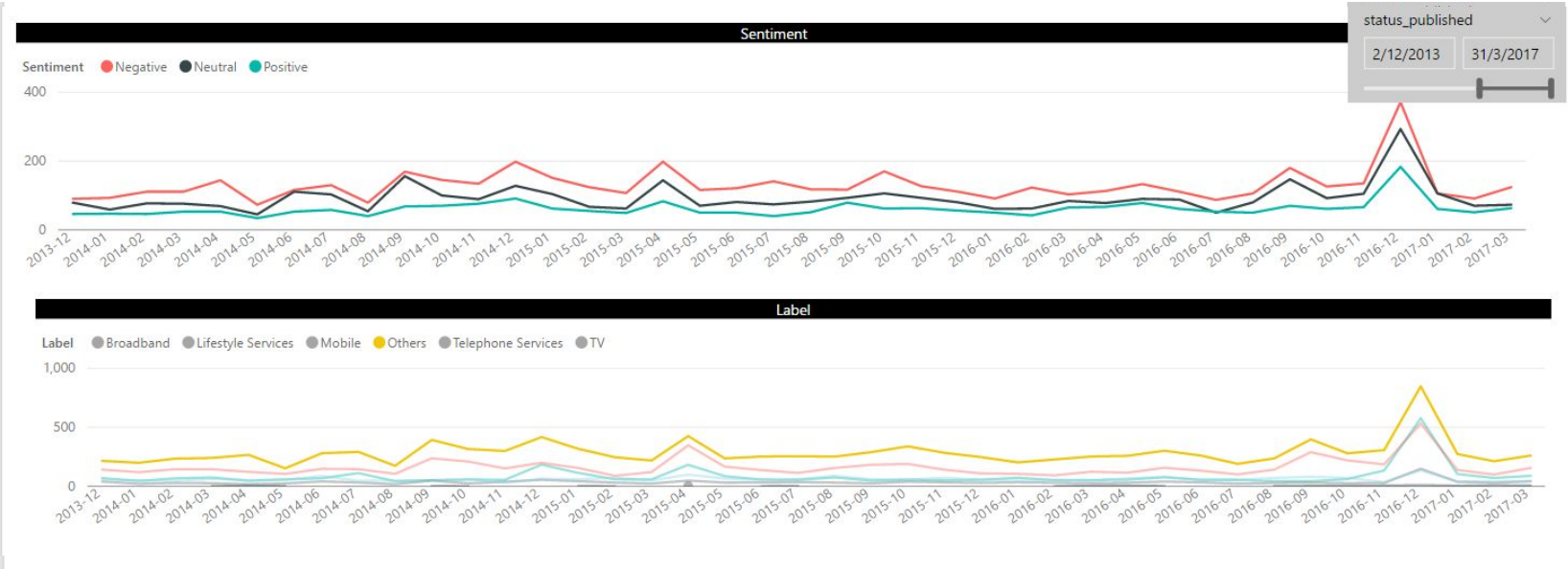




Singtel disruption: Fibre broadband services fully restored, 10% discount on broadband subscription for December

Insights from Data Analysis

3) Posts labelled under *Others* are mostly *negative* sentiments & are under *complaints*

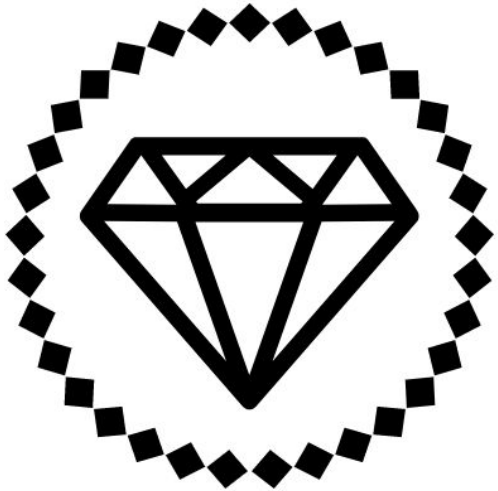


Key Learning Points

Our main challenges and takeaways

- Working with unclassified data
- Accuracy of model's categorization

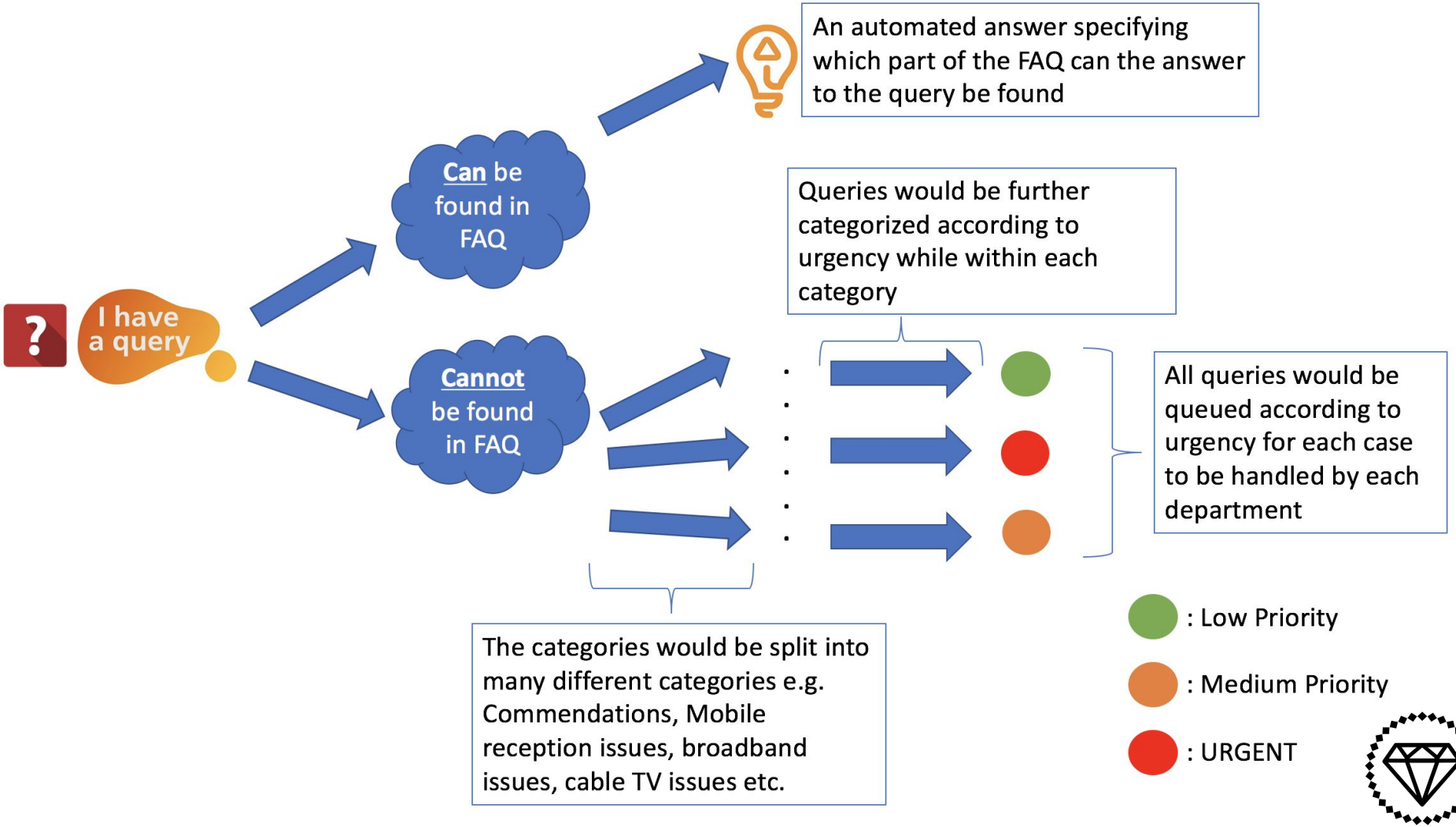
DIAMONDS



Service Like No Other

Thank You :)

Any Questions?


I have a query
Can be found in FAQ
Cannot be found in FAQ
An automated answer specifying which part of the FAQ can the answer to the query be found
Queries would be further categorized according to urgency while within each category
All queries would be queued according to urgency for each case to be handled by each department
The categories would be split into many different categories e.g. Commendations, Mobile reception issues, broadband issues, cable TV issues etc.
: Low Priority
: Medium Priority
: URGENT

40% Training 60% Test

RF: (Mean 0.515)

```
> RF <- train_models(container, algorithms= c("RF"))
> RF_CLASSIFY = classify_models(container, RF)
>
> SVM_CV <- cross_validate(container, 10, "RF")
Fold 1 Out of Sample Accuracy = 0.5213549
Fold 2 Out of Sample Accuracy = 0.5109923
Fold 3 Out of Sample Accuracy = 0.4945881
Fold 4 Out of Sample Accuracy = 0.5150421
Fold 5 Out of Sample Accuracy = 0.5183541
Fold 6 Out of Sample Accuracy = 0.5322581
Fold 7 Out of Sample Accuracy = 0.5231748
Fold 8 Out of Sample Accuracy = 0.5088562
Fold 9 Out of Sample Accuracy = 0.5110783
Fold 10 Out of Sample Accuracy = 0.5188962
>
```

BAGGING (Mean 0.497):

```
> SVM_CV <- cross_validate(container, 10, "BAGGING")
Fold 1 Out of Sample Accuracy = 0.4945881
Fold 2 Out of Sample Accuracy = 0.4883926
Fold 3 Out of Sample Accuracy = 0.5034216
Fold 4 Out of Sample Accuracy = 0.488081
Fold 5 Out of Sample Accuracy = 0.4927105
Fold 6 Out of Sample Accuracy = 0.5001516
Fold 7 Out of Sample Accuracy = 0.5111715
Fold 8 Out of Sample Accuracy = 0.5043452
Fold 9 Out of Sample Accuracy = 0.4988138
Fold 10 Out of Sample Accuracy = 0.4894377
>
```

50% Training 50% Test

RF: (Mean 0.514)

```
> RF_CLASSIFY = classify_models(container, RF)
>
> SVM_CV <- cross_validate(container, 10, "RF")
Fold 1 Out of Sample Accuracy = 0.5245025
Fold 2 Out of Sample Accuracy = 0.5051668
Fold 3 Out of Sample Accuracy = 0.5224903
Fold 4 Out of Sample Accuracy = 0.5373633
Fold 5 Out of Sample Accuracy = 0.5046062
Fold 6 Out of Sample Accuracy = 0.515433
Fold 7 Out of Sample Accuracy = 0.5105804
Fold 8 Out of Sample Accuracy = 0.5202723
Fold 9 Out of Sample Accuracy = 0.5082596
Fold 10 Out of Sample Accuracy = 0.5127042
>
> analytics = create_analytics(container, RF_CLASSIFY)
```

BAGGING: (Mean 0.498)

```
> RF_CLASSIFY = classify_models(container, RF)
>
> SVM_CV <- cross_validate(container, 10, "BAGGING")
Fold 1 Out of Sample Accuracy = 0.5114345
Fold 2 Out of Sample Accuracy = 0.490995
Fold 3 Out of Sample Accuracy = 0.4968722
Fold 4 Out of Sample Accuracy = 0.5224787
Fold 5 Out of Sample Accuracy = 0.4882615
Fold 6 Out of Sample Accuracy = 0.4986515
Fold 7 Out of Sample Accuracy = 0.4933495
Fold 8 Out of Sample Accuracy = 0.4980764
Fold 9 Out of Sample Accuracy = 0.4926254
Fold 10 Out of Sample Accuracy = 0.4972777
>
> analytics = create_analytics(container, RF_CLASSIFY)
```

60% Training 40% Test

RF: (Mean 0.515)

```
>
> RF <- train_models(container, algorithms= c("RF"))
> SVM_CV <- cross_validate(container, 10, "RF")
Fold 1 Out of Sample Accuracy = 0.5038497
Fold 2 Out of Sample Accuracy = 0.5195698
Fold 3 Out of Sample Accuracy = 0.5070796
Fold 4 Out of Sample Accuracy = 0.5142258
Fold 5 Out of Sample Accuracy = 0.515343
Fold 6 Out of Sample Accuracy = 0.5237812
Fold 7 Out of Sample Accuracy = 0.5323952
Fold 8 Out of Sample Accuracy = 0.4980888
Fold 9 Out of Sample Accuracy = 0.5262201
Fold 10 Out of Sample Accuracy = 0.5175282
>
```

BAGGING: (Mean 0.496)

```
> SVM_CV <- cross_validate(container, 10, "BAGGING")
Fold 1 Out of Sample Accuracy = 0.50886
Fold 2 Out of Sample Accuracy = 0.4938053
Fold 3 Out of Sample Accuracy = 0.4862275
Fold 4 Out of Sample Accuracy = 0.4872972
Fold 5 Out of Sample Accuracy = 0.5160163
Fold 6 Out of Sample Accuracy = 0.497022
Fold 7 Out of Sample Accuracy = 0.4854634
Fold 8 Out of Sample Accuracy = 0.4871717
Fold 9 Out of Sample Accuracy = 0.4992544
Fold 10 Out of Sample Accuracy = 0.5064275
>
```

Google Cloud Platform

IBM Watson

---

https://app.powerbi.com/view?r=eyJrIjoiMzQ5OGQ5YTAtZTQ5Yi00MDRhLTgwMmEtMThkNzdmMDJiYzhhIiwidCI6ImI3YzgzYWYyLWYxM2QtNDU0Yi04NjAwLTlhZDhmZWMxMDE3YyIsImMiOjEwfQ%3D%3D

https://badiamondsdash.shinyapps.io/dashboard/