Text Mining: Detecting Insults in Social Commentary?

—2012 kaggle competition

Jumao Yuan

EXST 7152 Final Project
April 28th, 2015

Agenda

- Background
- Data Description
- Data Preprocessing
- Evaluation
- Models
- Results
- Summary

Agenda

- Background
- Data Description
- Data Preprocessing
- Evaluation
- Models
- Results
- Summary

Comments	Insult
“Either you are fake or extremely stupid...maybe both...”	yes
“@tonnyb Or they just don’t pay attention”	no
“You with the ‘racist’ screen name\\n\nYou are a PieceOfShit.....”	yes
“your such a dickhead...”	yes
http://www.youtube.com/watch?v=tLYLLPHKRU4	no
“You are a liar.”	no

Agenda

Background

Data Description

Data Preprocessing

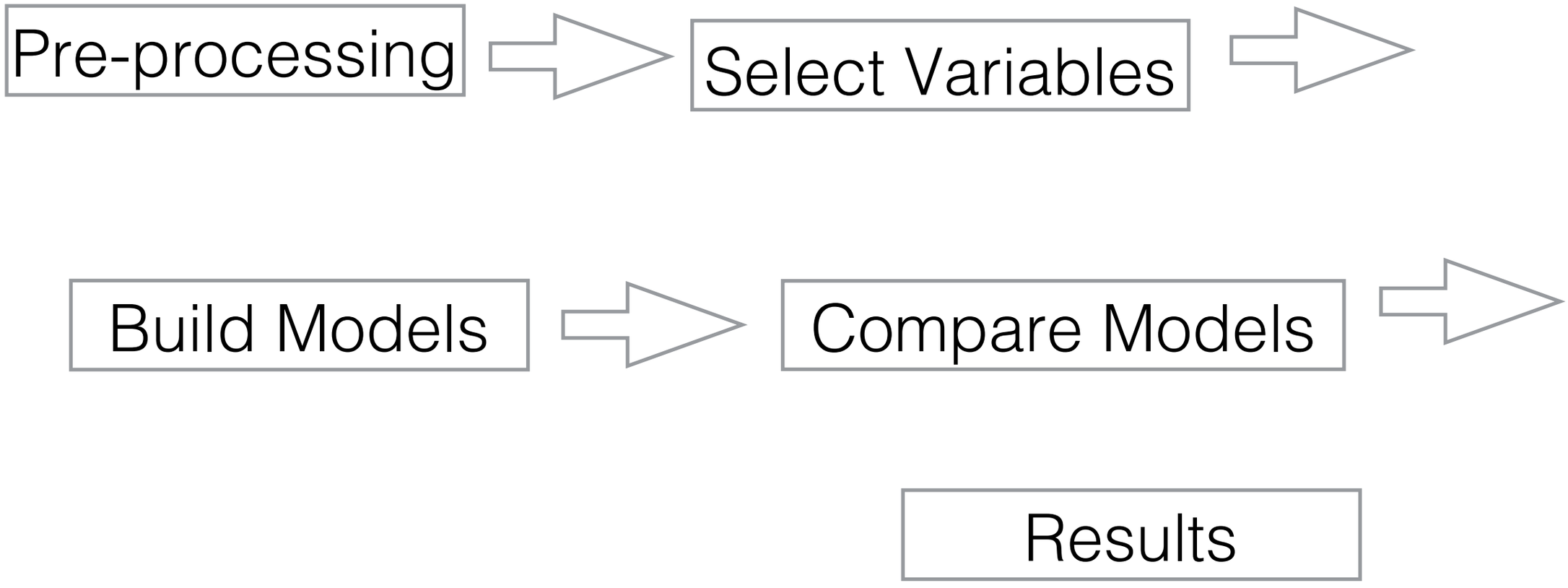
Evaluation

Models

Results

Summary

Workflow


Pre-processing
Select Variables
Build Models
Compare Models
Results

Single-classifier problem

Insult	Comment
1	"You fuck your dad."
0	"i really don't understand your point.\xa
0	"A\\xc2\\xa0majority of Canadians can
0	"listen if you dont wanna get married tc
0	"C\xe1c b\u1ea1n xu\u1ed1ng \u0111\
0	"@SDL OK, but I would hope they'd sign
0	"Yeah and where are you now?"
1	"shut the fuck up. you and the rest of yc
1	"Either you are fake or extremely stupid
1	"That you are an idiot who understands
0	"@jdstorm dont wish him injury but it h
0	"Be careful,Jimbo.OG has a fork with yo

Agenda

Background

Data Description

Data Preprocessing

Evaluation

Models

Results

Summary

2 variables and 3947 observations

Response variable is binary: “Is Insult” or “Is Not Insult”.

No missing values, but need do token and regular expression analysis.

TF-IDF (term frequency – inverse document frequency): reflect how important a word is to a document in a collection

Library(tm) in R

Text Parsing

“You with the ’racist’ screen name\\n\n\xc2You are a PieceOfShit 012……”

TermDocumentMatrix

D1 = “I like databases”

D2 = “I hate databases”,

	I	like	hate	databases
D1	1	1	0	1
D2	1	0	1	1

Stemming

“use”, “uses”“used”, “useful”, “using” have the same stem “use”

Feature Extraction

The sparse matrix is huge with high dimensional variables.

Our main task is to downsize the high dimensional sparse matrix since the stack pointer of memory in R is limited (500000).

Data split

70% training data set and 30% testing data set

R code

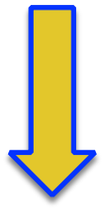
```
dd <- Corpus(VectorSource(docs))
dd <- tm_map(dd, stripWhitespace) % Eliminating Extra White Spaces
dd <- tm_map(dd, tolower) % Convert to Lower Case
dd <- tm_map(dd, removePunctuation) % Remove Punctuations
dd <- tm_map(dd, removeWords, stopwords("english")) % Remove stopwords %dd <- tm_map(dd, stemDocument)
dd <- tm_map(dd, removeNumbers) % Remove numbers
dd <- tm_map(dd, stemDocument,language = 'english') % Do Stemming
```

16297 features

“You with the ’racist’ screen name\\n\n\xc2You are a PieceOfShit 012.....fuckkkkkkkk”

You with the racist screen name You are a PieceOfShit fuckkkkkkkk

racist screen name PieceOfShit fuckkkkkkkk



racist screen name PieceOfShit fuck

Agenda

Background

Data Description

Data Preprocessing

Evaluation

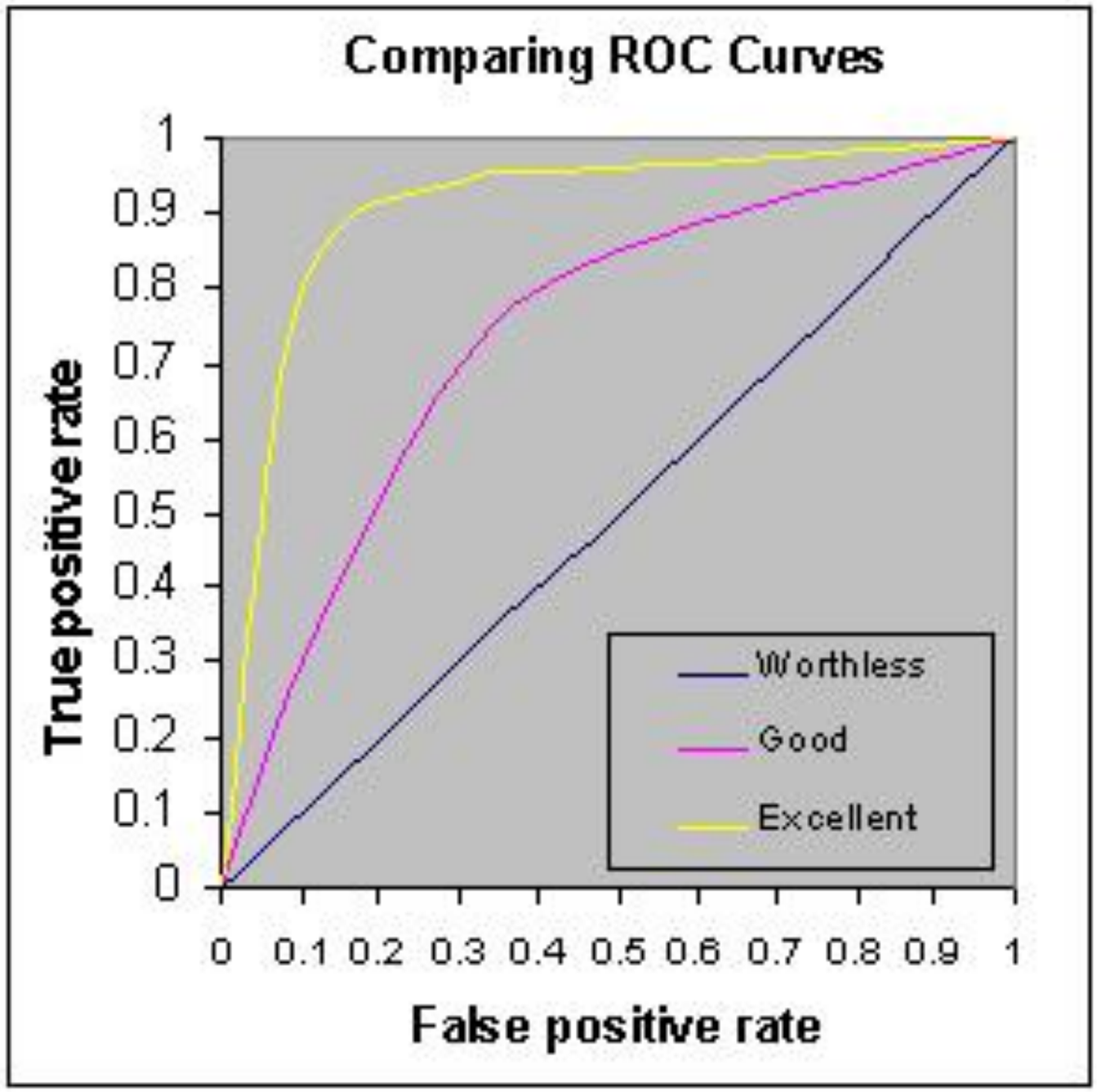
Models

Results

Summary

In R (using the verification package):
auc = roc.area(true_labels, predictions)

ROC Curve
AUC Score



https://www.kaggle.com/c/stumbleupon/details/evaluation
http://gim.unmc.edu/dxtests/roc3.htm

Agenda

Background

Data Description

Data Preprocessing

Evaluation

Models

Results

Summary

Models

Classification Tree

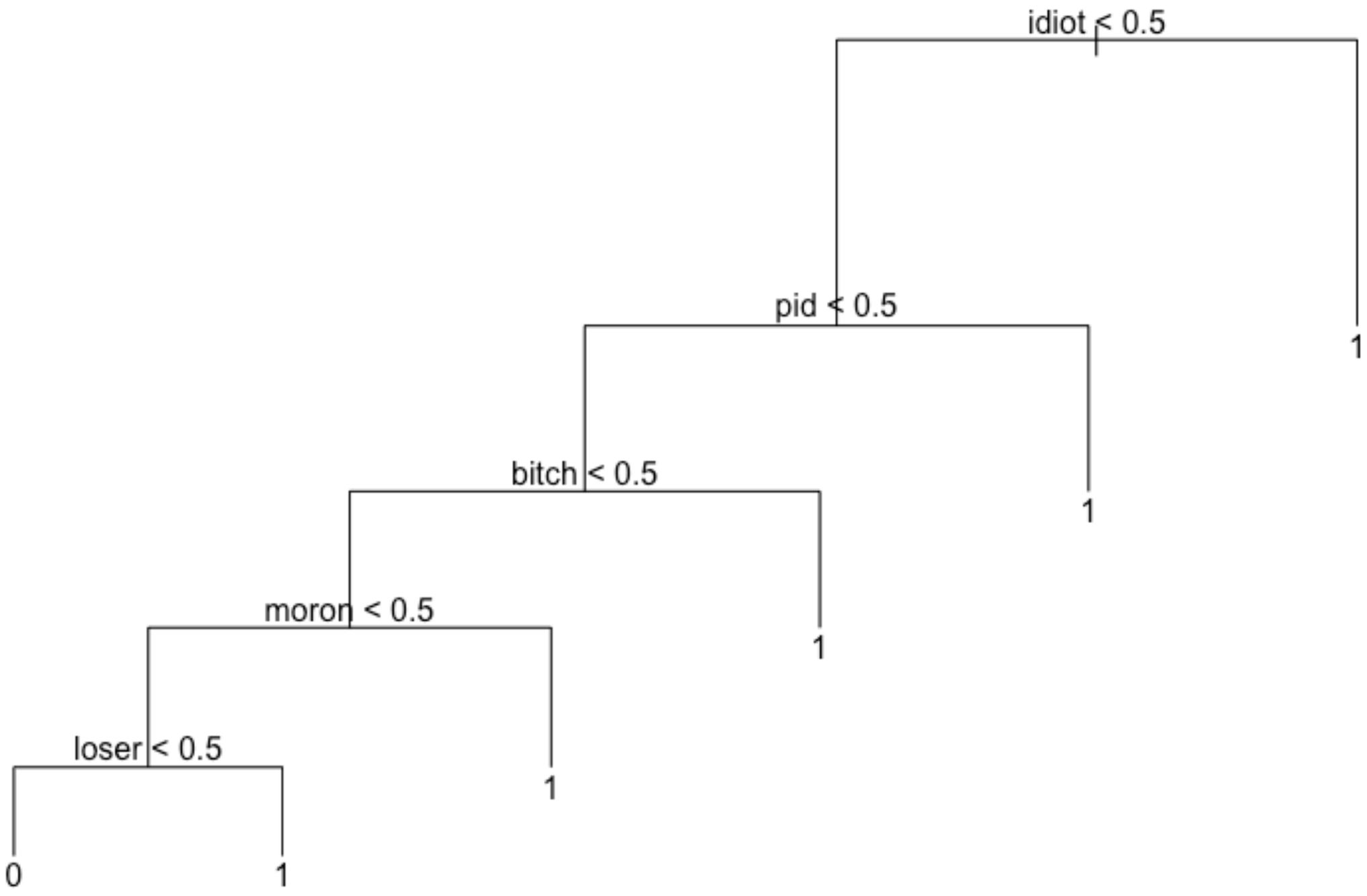
Random Forest

Models

Classification Tree

Random Forest

Library(tree)

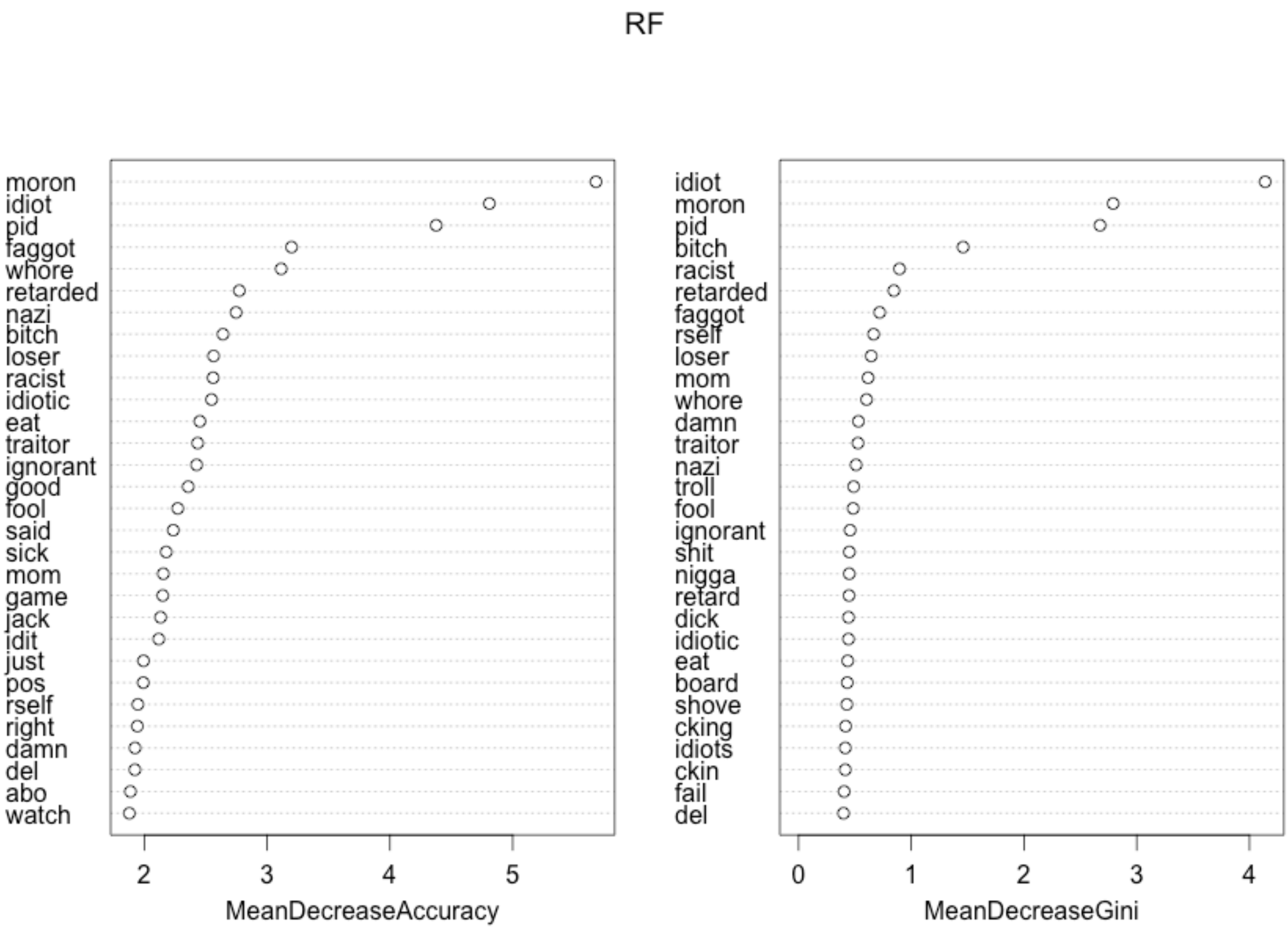


Models

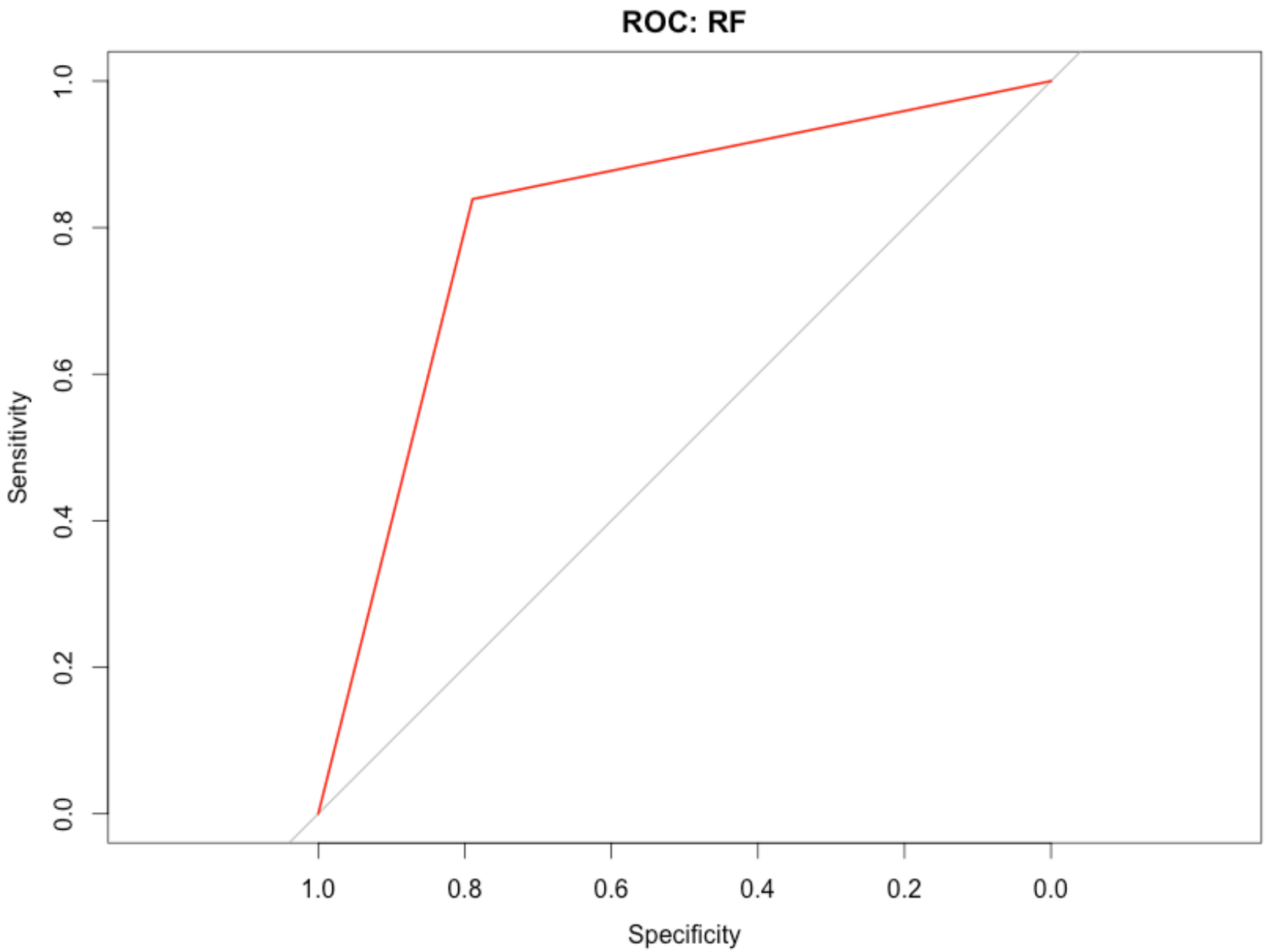
Classification Tree

Random Forest

Library(randomForest)



AUC score = 0.814



Agenda

Background

Data Description

Data Preprocessing

Evaluation

Models

Results

Summary

Method	AUC Score	Computation Time (s)
Random Forest	0.814	323.362
Decistion Tree	0.789	46.725
Logistic Regression	0.785	–
Naive Bayes	0.798	–
KNN	0.730	–
SVM	0.786	–

Agenda

Background

Data Description

Data Preprocessing

Evaluation

Models

Results

Summary

✓1. Sparse matrix optimization (16297 variables)

PieceOfShit = Shit; You're (need be deleted)

✓2. Random Forest, SVM and Naïve Bayes work well

✓3. It is still difficult to detect some false negative results such as "this book is fXXXing good" or new words (wordplays) such as "yuck fou"

Thanks ☺