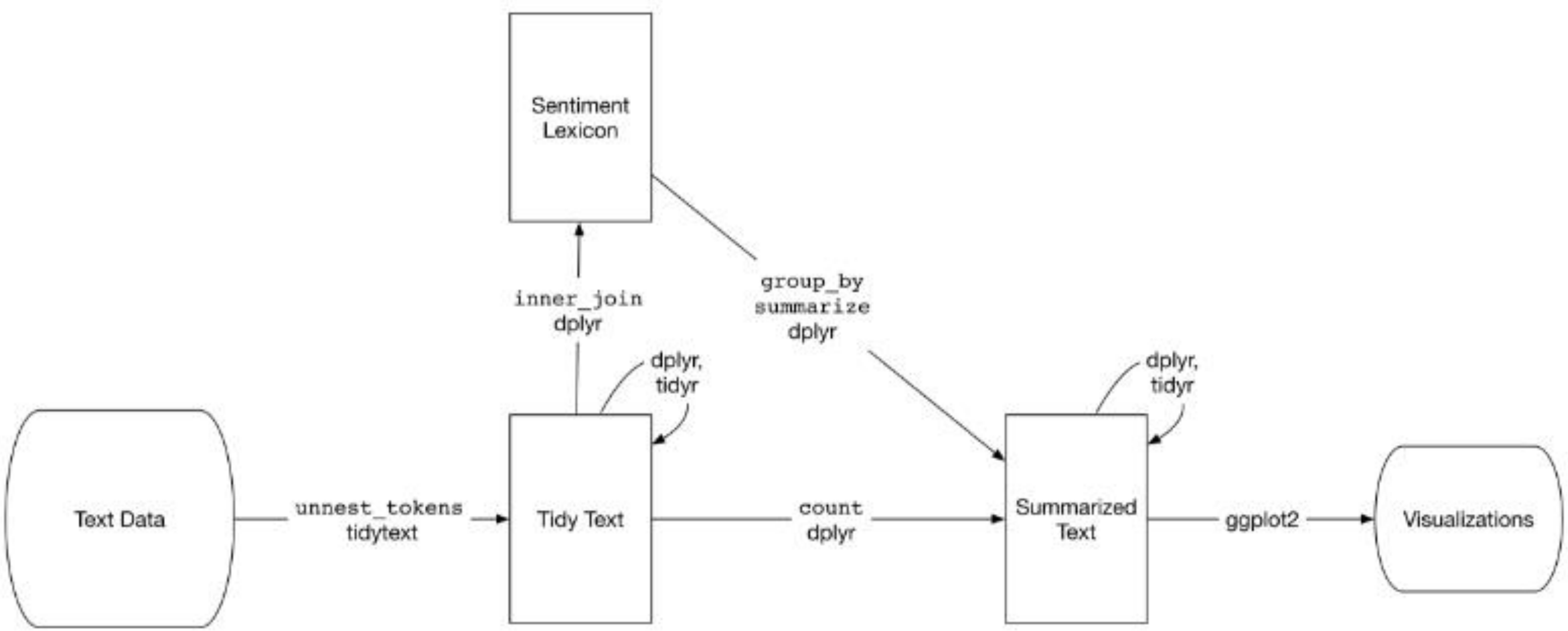
Sentiment Analysis

Siti Mariyah

Sentiment Analysis



Sentiment Analysis

- Install dan load library yang digunakan

```
install.packages(“RTextTools”)
install.packages(“e1071”)
library(RTextTools)
library(e1071)
```

- Masukkan kalimat-kalimat positive

```
pos_tweets =  rbind(
    c('I love this car', 'positive'),
    c('This view is amazing', 'positive'),
    c('I feel great this morning', 'positive'),
    c('I am so excited about the concert', 'positive'),
    c('He is my best friend', 'positive'))
```

- Masukkan data-data negative

Sentiment Analysis

- Masukkan kalimat-kalimat negative

```
neg_tweets = rbind(
  c('I do not like this car', 'negative'),
  c('This view is horrible', 'negative'),
  c('I feel tired this morning', 'negative'),
  c('I am not looking forward to the concert', 'negative'),
  c('He is my enemy', 'negative'))
```

- Masukkan kalimat-kalimat untuk testing

```
test_tweets = rbind(
  c('feel happy this morning', 'positive'),
  c('larry friend', 'positive'),
  c('not like that man', 'negative'),
  c('house not great', 'negative'),
  c('your song annoying', 'negative'))
```

Sentiment Analysis

- Menggabungkan tweet positif, negative dan tweet testing

```
tweets = rbind(pos_tweets, neg_tweets, test_tweets)
```

- Membangun *Document Term Matrix*

```
     matrix = create_matrix(tweets[,1], language=“english”, 
removeStopwords = FALSE, removeNumbers = TRUE, stemWords = 
FALSE)
```

- Lalu kita latih model Naïve Bayes dengan data training. Kita gunakan library e1017 yang mensyaratkan response variable nya numerik atau factor.

```
mat = as.matrix(matrix)
classifier = naiveBayes(mat[1:10], 
as.factor(tweets[1:10,2]))
```

Sentiment Analysis

- Setelah ditest kita bisa mengetes model tadi dan mendapatkan accuracy nya

```
predicted <- predict(classifier, mat[11:15,])
predicted
table(tweets[11:15,2], predicted)
recall_accuracy(tweets[11:15,2], predicted)
```

- Memprediksi kalimat baru

```
predict(classifier, “I am hungry”)
```

Sentiment Analysis

- Bagaimana jika menggunakan machine learning yang lain?? Gunakan library RTextTools

```
install.packages(“RTextTools”)
library(RTextTools)
```

- Bangun data container yang menjelaskan response variabel baik untuk data training dan data testing

```
container = create_container(matrix,
as.numeric(as.factor(tweets[,2])), trainSize = 1:10,
testSize = 11:15, virgin = FALSE)
```

Sentiment Analysis

- Tentukan machine learning algorithm yang akan dipakai

```
      model = train_models(container, algorithms=c(“MAXENT”, “SVM”,
“BAGGING”, “TREE”))
```

- Setelah bangun model, kita bisa klasifikasikan testing set menggunakan model yang sudah ditraining

```
      results = classify_models(container, models)
```

- Melihat accuracy

```
      table(as.numeric(as.factor(tweets[11:15,2])),
results[,”FOREST LABEL”]
      table(as.numeric(as.factor(tweets[11:15,2])),
results[,”FOREST LABEL”]
```

Sentiment Analysis

- Melihat recall accuracy

```
recall_accuracy(as.numeric(as.factor(tweets[11:15, 2])),
results[,"FORESTS_LABEL"])

recall_accuracy(as.numeric(as.factor(tweets[11:15, 2])),
results[,"MAXENTROPY_LABEL"])

recall_accuracy(as.numeric(as.factor(tweets[11:15, 2])),
results[,"TREE_LABEL"])

recall_accuracy(as.numeric(as.factor(tweets[11:15, 2])),
results[,"BAGGING_LABEL"])

recall_accuracy(as.numeric(as.factor(tweets[11:15, 2])),
results[,"SVM_LABEL"])
```

Sentiment Analysis

- Menyimpulkan hasil *(model summary)*

```
analytics = create_analytics(container, results)
summary(analytics)
head(analytics@document_summary)
```

- Melakukan cross validation

```
N=4
set.seed(2014)
cross_validate(container,N,"MAXENT")
cross_validate(container,N,"TREE")
cross_validate(container,N,"SVM")
cross_validate(container,N,"RF")
```