Sentiment Analysis of Painkiller Reviews: A Data-Driven Approach to Consumer Insights

Raheela Charania, Anmol Anchala, Emmanuel Wediko
BDA 620: Marketing Analytics 622
Dr. Wei Xiong
Mercer University
December 11, 2024

1. **Introduction:** Our dataset focuses on reviews of various painkillers and was obtained from Kaggle. Understanding consumer feedback is essential for raising customer satisfaction especially in terms of medication and health. This will in turn boost product quality, and guide marketing tactics in today's cutthroat market. Sentiment analysis gives companies useful insights by enabling them to categorize client feedback as either positive or negative. The objective of this project is to use CART (Decision Tree) and logistic regression models to analyze and categorize consumer sentiment. The results will assist companies in prioritizing customer-driven improvements, addressing unfavorable comments, and better understanding the demands of their customers.

2. **Dataset Description:** The dataset consists of customer reviews with binary sentiment labels:

- Positive sentiment: Encoded as 1
- Negative sentiment: Encoded as 0

Dataset Overview:

- Number of Records: 2,500
- Features: 19 variables after preprocessing

The dataset was preprocessed to clean the text and ensure data quality before modeling.

3. **Methodology**

3.1. **Text Preprocessing:** We underwent the following preprocessing steps to prepare them for analysis:

- Lowercasing: Conversion of all text to lowercase.
- Stopword and Punctuation Removal: Elimination of unnecessary words and symbols.
- Stemming: Reduction of words to their root forms.
- Dimensionality Reduction: Sparse terms in the Document-Term Matrix (DTM) were removed.

3.2. **Data Partitioning**: The dataset was split into:

- Training Set: 70% of the data for model building.
- Test Set: 30% of the data for model evaluation.

The training dataset was balanced using ROSE (Random Over-Sampling Examples) to address class imbalance, while the test set remained unbalanced for realistic performance evaluation.

4. **Models Implemented**

4.1. **Logistic Regression**: Logistic Regression is a widely used statistical model for binary classification. The model was trained on the balanced training dataset and evaluated on the test dataset.

Results:

- Accuracy: 55.3%
- Sensitivity: 66.7%
- Specificity: 61.4%
- AUC: 0.603

Logistic Regression achieved balanced performance with a slightly better sensitivity, demonstrating its ability to identify positive cases effectively.

4.2. **CART (Decision Tree):** The CART model uses decision trees to classify reviews by iteratively splitting the data on key features. The model was trained on the balanced dataset and evaluated on the test set.

Results:

- Accuracy: 62.7%
- Sensitivity: 49.1%
- Specificity: 66.3%
- AUC: 0.5725

The CART model performed slightly better in overall accuracy and specificity but demonstrated lower sensitivity, indicating it was better at identifying negative reviews.

5. **Model Comparison**: The performance metrics of the two models are summarized below:

Model	Accuracy	Sensitivity	Specificity	AUC
Logistic Regression	55.3%	66.7%	61.4%	0.603
CART (Decision Tree)	62.7%	49.1%	66.3%	0.573

6. **Results and Recommendations**: Findings:

1. Logistic Regression:
    - Achieved balanced performance across metrics, with an AUC of 0.603.
    - Demonstrated better sensitivity, meaning it effectively identified positive reviews.
2. CART:
    - Achieved slightly higher accuracy and specificity but lower sensitivity.
    - Its AUC of 0.573 indicates moderate performance, but the trade-offs suggest CART is better at identifying negative reviews.

Business Recommendations

All these recommendations are based on our project only:

1. Businesses should deploy Logistic Regression for sentiment classification due to its balanced performance and superior ability to identify positive cases.
2. CART can be used alongside Logistic Regression to further analyze negative reviews, as it shows higher specificity.
3. Sentiment analysis findings should be used to:
    - Address recurring negative feedback to improve customer satisfaction.
    - Highlight positive customer experiences in marketing campaigns.
    - Guide product improvements based on actionable insights from the data.

7. **Conclusion:** Our study used CART and logistic regression models to successfully analyze client sentiment. Although both models performed moderately, Logistic Regression was the suggested model for deployment since it offered superior balance and sensitivity. Businesses may increase consumer satisfaction, product quality, and marketing strategy by utilizing sentiment analysis to obtain important information.

Appendix

Experiment Process:

Text Processing:

```
<<DocumentTermMatrix (documents: 2500, terms: 5940)>>
Non-/sparse entries: 84408/14765592
Sparsity           : 99%
Maximal term length: 51
Weighting          : term frequency (tf)
<<DocumentTermMatrix (documents: 11, terms: 16)>>
Non-/sparse entries: 17/159
Sparsity           : 90%
Maximal term length: 8
Weighting          : term frequency (tf)
Sample             :
    Terms
Docs form great habit headach howev like need sever time toler
  20    0     0     0       0     0    0    1     0    1     1
  21    0     0     0       0     0    0    0     0    0     0
  22    0     0     0       0     1    1    0     1    0     0
  23    1     0     0       0     0    0    0     0    0     0
  24    0     0     0       0     0    0    1     1    0     0
  25    0     0     0       4     0    0    0     2    1     0
  26    0     0     0       0     0    0    0     0    0     0
  27    0     1     0       0     0    0    0     0    1     0
  28    0     0     0       0     1    1    0     0    0     0
  29    0     0     0       1     0    0    0     0    0     0
   [1] "addict"       "care"         "chronic"      "due"          "effect"       "great"        "headach"
   [8] "howev"        "like"         "long"         "need"         "prescrib"     "problem"      "sever"
  [15] "time"         "toler"        "use"          "vicodin"      "will"         "work"         "year"
  [22] "abl"          "everi"        "feel"         "function"     "just"         "medic"        "normal"
  [29] "one"          "reliev"       "tri"          "also"         "doctor"       "find"         "first"
  [36] "help"         "medicin"      "now"          "take"         "thing"        "ago"          "left"
  [43] "morn"         "pain"         "recommend"    "seem"         "actual"       "day"          "dont"
  [50] "far"          "felt"         "get"          "ibuprofen"    "ive"          "littl"        "minut"
  [57] "realli"       "sinc"         "sleep"        "still"        "suffer"       "taken"        "took"
  [64] "well"         "within"       "almost"       "alway"        "drug"         "enough"       "fioricet"
  [71] "given"        "mani"         "narcot"       "prescript"    "relief"       "side"         "back"
  [78] "can"          "disc"         "high"         "leg"          "life"         "live"         "never"

 [85] "night"        "noth"         "percocet"     "put"          "say"          "tramadol"     "come"
 [92] "hurt"         "month"        "old"          "told"         "usual"        "better"       "made"
 [99] "much"         "bad"          "best"         "doesnt"       "last"         "make"         "migrain"
[106] "caus"         "didnt"        "extrem"       "gave"         "hour"         "pill"         "tablet"
[113] "thank"        "withdraw"     "wors"         "lot"          "surgeri"      "tylenol"      "went"
[120] "wonder"       "around"       "away"         "chang"        "control"      "couldnt"      "way"
[127] "anyth"        "hope"         "lortab"       "muscl"        "ultram"       "week"         "med"
[134] "stop"         "less"         "lower"        "nerv"         "notic"        "start"        "10325"
[141] "bed"          "final"        "found"        "got"          "know"         "manag"        "anxieti"
[148] "complet"      "gone"         "head"         "hydrocodon"   "two"          "arthriti"     "cant"
[155] "even"         "fibromyalgia" "keep"         "norco"        "think"        "sometim"      "50mg"
[162] "without"      "advil"        "dose"         "ever"         "injuri"       "later"        "peopl"
[169] "see"          "though"       "differ"       "give"         "daili"        "depress"      "knee"
[176] "shoulder"     "experienc"    "nausea"       "right"        "symptom"      "neck"         "stomach"
[183] "someth"       "good"         "want"         "experi"       "recent"       "anoth"        "alev"
[190] "period"       "naproxen"
<<DocumentTermMatrix (documents: 2500, terms: 19)>>
Non-/sparse entries: 14055/33445
Sparsity           : 70%
Maximal term length: 8
Weighting          : term frequency (tf)
```

Data Partition (2 fold):

```r
#Partition dataset
set.seed(123)
train.index = sample(c(1:dim(final_dataset)[1]), dim(final_dataset)[1]*0.7)
reviews.train = final_dataset[train.index, ]
reviews.test = final_dataset[-train.index, ]

nrow(reviews.train)
nrow(reviews.test)
```

```
[1] 1750
[1] 750
```

Data Balance:

```r
##Balance Dataset##
#Here we can see that the dataset is not balanced so I will balance it before moving forward.
print("Dataset prior to balancing")
table(reviews.train$sentiment)/nrow(reviews.train)

#Balance only train dataset
library(ROSE)
data.train.balanced.over = ovun.sample(sentiment ~ ., data=reviews.train, p=0.5, method="over")

print("Dataset after balancing")
data.train.balanced.over = data.train.balanced.over$data
table(data.train.balanced.over$sentiment)/nrow(data.train.balanced.over)

data.train.balanced.over$sentiment_binary <- ifelse(data.train.balanced.over$sentiment == "Positive", 1, 0)
reviews.test$sentiment_binary <- ifelse(reviews.test$sentiment == "Positive", 1, 0)
```

```
[1] "Dataset prior to balancing"

 Negative  Positive
0.2051429 0.7948571
[1] "Dataset after balancing"

 Negative  Positive
0.4941818 0.5058182
```

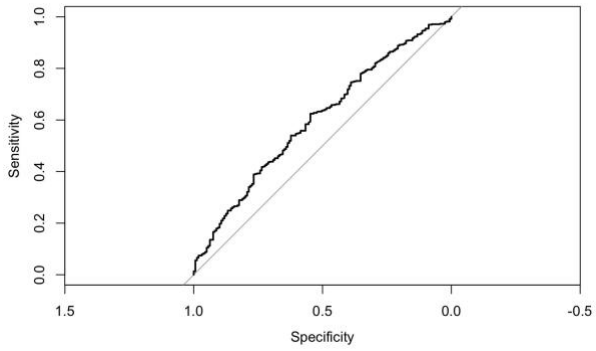
Experiment Results: Model Construction & Evaluation:

Logistic Regression:

```
Call:
glm(formula = sentiment_binary ~ prescrib + time + work + year +
    tri + medicin + pain + get + side + back + can + tramadol,
    family = "binomial", data = data.train.balanced.over)

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -0.47549    0.07852  -6.055 1.40e-09 ***
prescrib    -0.36115    0.07632  -4.732 2.22e-06 ***
time        -0.09660    0.06674  -1.447 0.147759
work         0.21012    0.04721   4.451 8.56e-06 ***
year         0.71249    0.07566   9.417  < 2e-16 ***
tri          0.25898    0.08288   3.125 0.001779 **
medicin      0.31474    0.07875   3.997 6.42e-05 ***
pain        -0.12036    0.02957  -4.071 4.68e-05 ***
get          0.11468    0.07218   1.589 0.112103
side         0.32039    0.08626   3.714 0.000204 ***
back         0.42051    0.07193   5.846 5.03e-09 ***
can          0.73845    0.09883   7.472 7.91e-14 ***
tramadol    -0.08336    0.05495  -1.517 0.129261
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 3842.8  on 2771  degrees of freedom
Residual deviance: 3509.9  on 2759  degrees of freedom
AIC: 3535.9

Number of Fisher Scoring iterations: 4
```

```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0  96 272
         1  63 319

               Accuracy : 0.5533
                 95% CI : (0.5169, 0.5893)
    No Information Rate : 0.788
    P-Value [Acc > NIR] : 1

                  Kappa : 0.097

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.6038
            Specificity : 0.5398
         Pos Pred Value : 0.2609
         Neg Pred Value : 0.8351
             Prevalence : 0.2120
         Detection Rate : 0.1280
   Detection Prevalence : 0.4907
      Balanced Accuracy : 0.5718

       'Positive' Class : 0
```

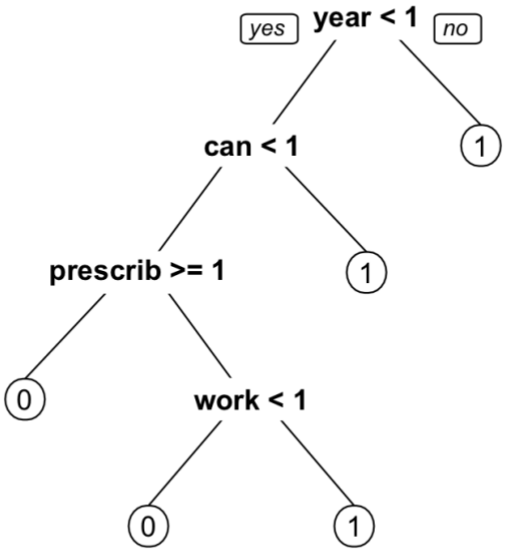
train(x, ...)

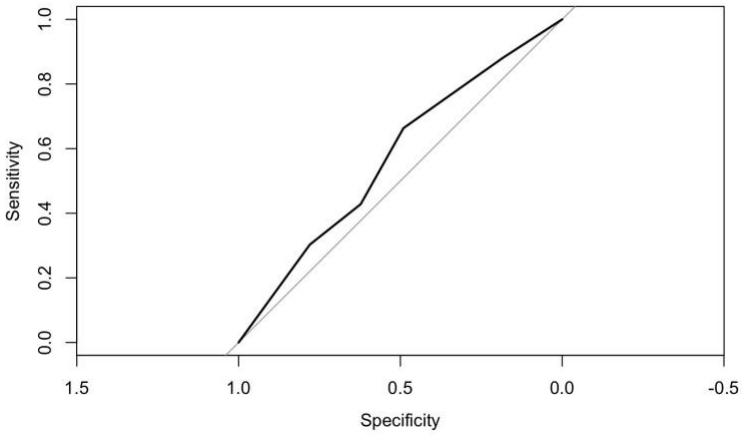


CART Regression:

CART 5-Fold Cross Validation (output is cp value)

```
CART

2772 samples
  19 predictor

No pre-processing
Resampling: Cross-Validated (5 fold)
Summary of sample sizes: 2217, 2218, 2218, 2218, 2217
Resampling results across tuning parameters:

  cp          RMSE       Rsquared    MAE
  0.01625047  0.4828007  0.06837564  0.4649782
  0.02109693  0.4908435  0.03744931  0.4801119
  0.03829069  0.4975349  0.02823849  0.4936239

RMSE was used to select the optimal model using the smallest value.
The final value used for the model was cp = 0.01625047.
```






```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0  78 199
         1  81 392

               Accuracy : 0.6267
                 95% CI : (0.5909, 0.6614)
    No Information Rate : 0.788
    P-Value [Acc > NIR] : 1

                  Kappa : 0.121

 Mcnemar's Test P-Value : 2.708e-12

            Sensitivity : 0.4906
            Specificity : 0.6633
         Pos Pred Value : 0.2816
         Neg Pred Value : 0.8288
             Prevalence : 0.2120
         Detection Rate : 0.1040
   Detection Prevalence : 0.3693
      Balanced Accuracy : 0.5769

       'Positive' Class : 0
```