# Sentiment Analysis of Drug Reviews

**Prathamesh Pradip Datar**
School of Information Studies,
Syracuse University, NY, 13210
pdatar@syr.edu

**Avin Deshmukh**
School of Information Studies,
Syracuse University, NY, 13210
avdeshmu@syr.edu

**Abstract:**

The main objective of the project was to predict customer sentiment based on drug reviews and to identify ambiguous reviews to better serve drug manufacturers and new customers. The dataset for the project was collected from Kaggle 2018 University Club Hackathon and consisted of customer provided ratings for a drug and its review. To strike the right balance between the vocabulary size and the model accuracy, we used a custom stop words list along with various parameters in the vectorization process. We conducted a comparative study of various supervised machine learning classifiers such as SVM and Naive Bayes models to better predict the sentiment of a customer. Based on the evaluation parameters such as Precision, Recall, F-score, Accuracy, and extreme misclassification errors, we concluded that LinearSVC classifiers performed better than Naive Bayes models for predicting sentiment on the given dataset. We hypothesized that the number of conjunctions used in a review is directly proportional to the ambiguity of a review. Therefore, to identify the ambiguous reviews, we used a combination of misclassification errors of LinearSVC with a high number of conjunctions.

**Introduction:**

Sentiment classification is a special case of text categorization problem, where the classification is done based on the attitude expressed by the authors. Sentiment analysis requires a deep understanding of the document under analysis because the concern here is how the sentiment is being communicated. Usually, sentiments are classified as positive, neutral, and negative based on the rating scale.

Drug surveillance is a huge challenge when the drug hits the market after the controlled trials. Trials are often performed on limited test samples and do not truly represent the entire population. It is also important to understand how the general population uses a particular drug, perceives its safety, and side effects (The Conversation, 2017). In the digital age, we now have access to large numbers of drug reviews. Patients feel connected by sharing their experiences and are often looking for stories from other patients that cannot be shared comfortably with family or friends. Thus, it becomes necessary to perform sentiment analysis to better assist drug manufacturers as well as fellow users in getting valuable feedback.

Humans are inclined towards using ratings as an indicator of the quality or effectiveness of a drug (Gopalakrishnan, & Ramaswamy, 2017). There are several complex cases where it is not clear to the user about how to rate the side effects or when to write a review in their medication journey leading to ambiguous reviews and discrepancies between the review and its rating. Therefore, along with sentiment analysis, it is important to consider the ambiguity of a review to provide additional information to the drug manufacturers or the users about the side effects or patient experiences.

Conjunctions combine multiple phrases to form long sentences. The more conjunctions we use in a review, the more complex it becomes to classify sentiment. Therefore, the ambiguity of a review can be visualized from the number of conjunctions used in the review. By effectively utilizing the conjunction count in the review, we can make good conclusions about its ambiguity.

**Literature Survey:**

Sentiment analysis is a hot topic in the text mining industry. Even though there had been some efforts to classify sentiments using unsupervised/semi-supervised learning techniques, most of the attempts focus on using supervised learning techniques (Moraes, Valiati, & Gavião, 2013). The most popular sentiment analysis algorithms are SVM and Naive Bayes, and many authors have published good results with better accuracy using SVM models. One of the important characteristics of the sentiment literature is that many experiments have been conducted on balanced datasets but there has been little discussion on the effects of unbalanced data for sentiment classification. Although, it is typical of the review datasets to have substantially more positive reviews than negative reviews. There are certain ways to deal with unbalanced data such as adopting a random under-sampling method, but it can discard potentially useful data.

In the sentiment literature, conjunction analysis has been done for two purposes (Farooq, Nongaillard, Ouzrout, & Qadir, 2013). First, the role of conjunctions is analyzed to improve sentiment analysis. Second, it is used for predicting the sentiment orientation of words to build a lexicon dictionary of opinionated words with their associated polarity.

**Project Pipeline:**

The following is the summary of our methodology for developing and implementing the prediction models along with proposing a unique technique for ambiguity detection.

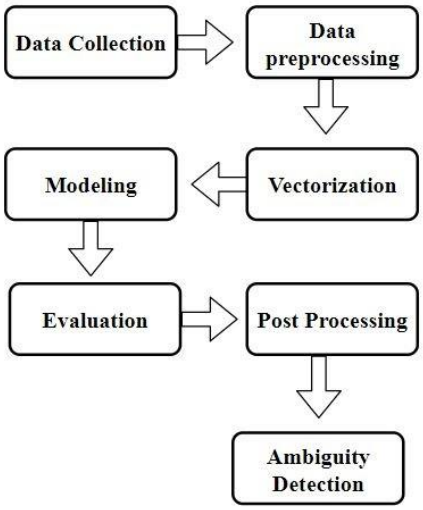


*Fig 1. Project Pipeline*

1) Data collection using the Kaggle website for 2018 Winter Kaggle University Club Hackathon
2) Data preprocessing by removing HTML characters, delimiters, and special characters
3) Defining custom stop words list, using unigrams and bigrams for better vectorization
4) Apply the following classification models using the respective training data set
    i. LinearSVC
    ii. Multinomial Naive Bayes

iii. Bernoulli Naive Bayes

5) Model Evaluation using evaluation parameters such as Precision, Recall, F-scores, Accuracy, and extreme misclassification errors
6) Data post-processing to create a conjunction count for all reviews
7) Ambiguity detection by finding reviews misclassified by supervised learning models and having a higher conjunction occurrence

**Data Collection:**

The UCI ML Drug Review dataset was obtained by crawling online pharmaceutical review sites. This data was originally posted on the UCI Machine Learning repository (Gräßer, Kallumadi, Malberg, & Zaunseder, 2018) and was used for the Winter 2018 Kaggle University Club Hackathon. The dataset provided patient reviews on specific drugs along with related conditions and a 10-point patient rating system reflecting overall patient satisfaction. The dataset contained two separate train and test CSV files with a 75-25 % split accounting for 215,065 examples.

.

| | uniqueID | drugName | condition | review | rating | date | usefulCount |
|---|---|---|---|---|---|---|---|
| 0 | 206461 | Valsartan | Left Ventricular Dysfunction | "It has no side effect, I take it in combinati... | 9 | 20-May-12 | 27 |
| 1 | 95260 | Guanfacine | ADHD | "My son is halfway through his fourth week of ... | 8 | 27-Apr-10 | 192 |

*Fig 2. Dataset example*

| **Dataset Variables** | **Explanation** |
|---|---|
| drugName | Name of the drug |
| condition | Condition faced by the patients |
| review | Drug reviews posted by the patients |
| rating | 10-star patient rating system reflecting overall patient satisfaction |
| date | Date of posted review |
| usefulCount | Number of users finding the review useful |

*Table 1. Explanation of variables in the dataset*

**Data preprocessing:**

To maintain a good quality of the textual reviews, we performed pre-processing to remove undesirable components from the dataset. In the beginning, we checked for duplicates in the data and removed all NA's from the dataset. There were some discrepancies in the "condition" variable that it contained several examples having uninterpretable conditions such as "2 users found this comment helpful.", "3 users found this comment helpful." etc. Since we wanted to ensure good data quality, we removed those reviews. The dataset was extracted from web crawling and thus contained HTML characters and delimiters. Keeping that in mind, we eliminated HTML characters and delimiters from the dataset. Special characters and punctuation marks are a way of

communication and do not contain any information value. Thus, we removed special characters and punctuation marks from the dataset to improve text quality for the vectorization process.

**Vectorization:**

Text vectorization is the process of converting text into numerical representations. This module in the project pipeline answers what we count and how to count. The following table summarizes various combinations of parameters used for vectorization and corresponding vocabulary size along with model accuracies of supervised machine learning algorithms.

| Parameters | Vocabulary Size | Model Accuracy (On LinearSVC) | Model Accuracy (On Multinomial NB) | Model Accuracy (On Bernoulli NB) |
|---|---|---|---|---|
| Lowercase = False | 64659 | 76.225% | 66.582% | 65.118% |
| Lowercase = True | 49836 | 74.463% | 65.912% | 63.939% |
| Min_df = 3, Max_df = 0.7 | 24770 | 71.866% | 65.106% | 62.874% |
| ngram_range = (1,2) | 382802 | 85.508% | 74.46% | 73.22% |
| English stopwords | 360705 | 84.63% | 73.518% | 72.053% |
| Custom stopwords | 366559 | 85.481% | 74.161% | 73.012% |

*Table 2. Parameters and model accuracies*

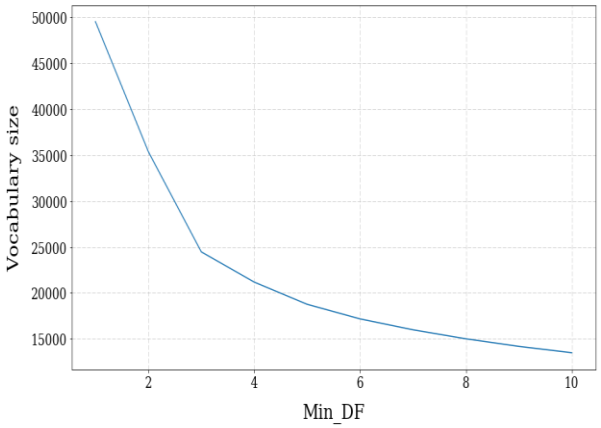


*Fig 3. Vocabulary size vs min_df*

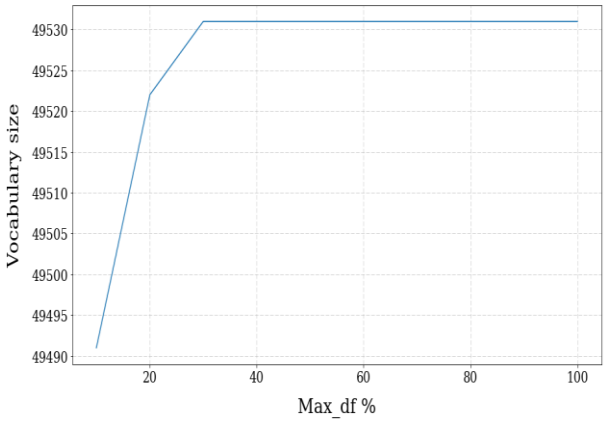


*Fig 4. Vocabulary size vs max_df*

The sparsity of the vectorized matrix is directly proportional to the vocabulary size. If we have a highly sparse matrix, it is difficult for the machine learning algorithm to optimize and find an accurate solution. To better assess the effect of parameters on vectorization, we created a baseline representation to compare vocabulary size and model accuracy. Baseline representation consisted of using a unigram count vectorizer with lowercase = False and other default parameters. Then we started to add more parameters to see what the changes in the vocabulary size and model accuracy across different models were. By changing all words to lowercase, we were able to strike the right balance between vocabulary size and model accuracy.

For a good combination of min_df and max_df, we plotted various values of the same and tracked the change in vocabulary size. Looking at Fig 3, we could see that min_df = 3 would be good enough to remove typos or unwanted words in the dataset thus retaining the uniqueness of words. Highly frequent words lose their ability to uniquely represent a review. Therefore, setting max_df = 0.7 would ensure removing most frequent words.

Unigrams, unfortunately, do not comprehend multiple word patterns in the data. Therefore, by incorporating unigrams as well as bigrams we would account for double words patterns for sentiment analysis. Even though vocabulary size increased a lot, we were also able to increase the model accuracies across different models.

The words that appear in more than 70% of the reviews lose their information value to uniquely represent the review and therefore need to be removed. We found common words from our vocabulary with the stopword list provided by the sklearn package and removed those words which occurred more than 70% in our reviews. This custom list allowed us to avoid removing some of the informative words such as “can”, “cannot”, “hasn't”, “without”.

Final CountVectorizer combination consisted of all lowercase words with min_df = 3, max_df = 0.7, custom stopwords list and unigrams and bigrams that provided better model accuracies with a reduction in vocabulary size.

**Modeling:**

Before applying various supervised models, it is necessary to define our target labels and target categories. From Fig 5 we can deduce that the ratings provided by the patients are polar with most of the ratings being positive and highly rated. Therefore, instead of applying ML models to predict 10 classes, we can better divide the sentiment analysis into three categories such as positive, negative, and neutral. We treated rating 1 and 2 as negative sentiments, rating 9 and 10 as positive sentiments, and the rest of the ratings as neutral sentiments. From table 3, we can see that the category division is carried out to reduce the skewness of the data while preserving true sentiments.

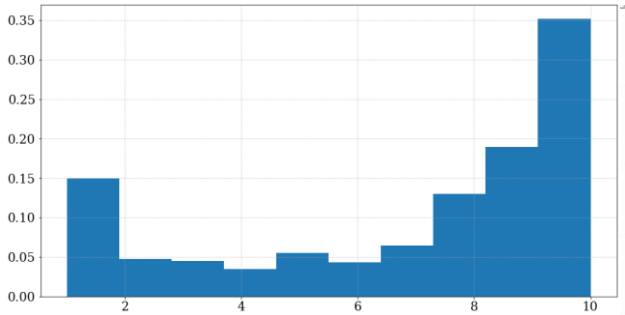


*Fig 5. Histogram of rating distribution*

| Category | Percentage Distribution |
|---|---|
| Negative | 17.753 |
| Positive | 33.509 |
| Neutral | 48.737 |

*Table 3. Category distribution*

SVMs and Naive Bayes models are famous in text mining to produce good results on text classification problems. Naive Bayes models depend on the calculation of conditional and posterior probabilities to predict a category whereas SVM depends on finding a hyperplane to distinguish between two categories.

Two common Naive Bayes models for text classification are as follows:

1) Multinomial model (MNB)

MNB models the number of counts of a feature. Therefore, a CountVectorizer is required to convert words into its numerical representation. Since MNB uses term frequency it can be used to classify large texts.

2) Bernoulli model (BNB)

BNB models the presence/absence of a feature. Therefore, a BooleanVectorizer is required to convert words into its numerical representation. Since BNB only accounts for the presence or absence of a feature, it is better for classifying shorter texts.

There are various SVM implementations such as non-linear kernels like RBF, polynomial, and linear kernels. Since most of the textual features are linearly separable, we used the LinearSVC model. The reason behind not using SVC programming implementation is the higher compute power and one Vs one multiclass strategy as opposed to one vs all strategy for LinearSVC.

**Evaluation:**

We compared three supervised learning algorithms such as LinearSVC, Bernoulli NB, and Multinomial NB mentioned in the modeling section to evaluate the best model for the sentiment analysis of patient reviews. Below in Table 4, we have presented a comparison between various evaluation parameters for the supervised algorithms.

| Supervised Learning Models | Model Accuracy | F-score | Precision | Recall | Extreme Misclassification Error |
|---|---|---|---|---|---|
| LinearSVC | 0.85481 | 0.85 | 0.85 | 0.86 | 1.598% |
| Multinomial Naive Bayes | 0.74160 | 0.74 | 0.74 | 0.74 | 2.973% |
| Bernoulli Naive Bayes | 0.73011 | 0.73 | 0.73 | 0.73 | 3.539% |

*Table 4. Model Comparison and evaluation*

Based on all the evaluation parameters, LinearSVC proved to be a better model for sentiment analysis. Naive Bayes models did not perform well on the dataset because the reviews are complex and contain various phrases portraying different sentiments. Therefore, a classifier modeling just the presence or the count of a feature is not enough to categorize the sentiments effectively.

Model prediction on the test dataset allowed us to identify the misclassified errors in predicting the sentiment analysis. A qualitative analysis of the errors revealed some general patterns in the way patients composed their reviews. These errors can be effectively classified into four types as follows:

1. Patients are more sensitive towards the side effects than the effectiveness of the drug thus producing lower ratings

   Had open heart surgery and double mastectomy in a span of 2 years Very painful nerve pain most of the time Bio Freeze Rol l On Definitely works for the pain Unfortunately it has literally burned my chest Don t want anyone to go through this Tha nk you Be well everyone

   In this case, the drug is working perfectly fine, but the reviewer is not happy with the side effects and decided to provide a lower rating for the drug.

2. Patients are less sensitive towards the side effects than the effectiveness of the drug thus producing higher ratings

   I took victoza for well over a year and it improved my blood sugar readings tremendously My A1C had been as high as 9 and it got down to 6 9 with victoza However I was sick SO MUCH Constant diarrhea burping the nasty boiled egg kind and extreme nausea and vomiting I finally started tracking my sick days and in one month I was sick 4 days And by sick I mean throwing up every few minutes all day long I had several tests done and the only thing discovered was that I digest food slower tha n normal which is what victoza is designed to do I decided it was probably the victoza making me so sick I stopped taking it completely and have not been sick for over month Blood sugars are not good now so waiting to see my Dr

   In this case, despite facing some side effects, the reviewer was happy and decided to give a higher rating

3. Patients writing the reviews too soon for the drug to create any impact

```
I m 37 and my doctor advised me to take 1 4 of a 20 mg pill and the results were nothing the first time I took it on an e
mpty stomach as advised I tried 1 2 of a pill a week later two hours after I ate and the results were nothing once again I
m going to take the whole 20mg next week
```

   In the example, the reviewer expressed his views before he took the full course of the drug and thus gave a lower rating.

4. Patient experiences based on bad practices

```
Periodically on Flagyl bc of crohns and rarely drink however I was on it during holidays a couple years ago and toasted t
he celebration with a half glass of champagne 3 hours later I was burning up felt like my chest was abt to explode 1hour a
fter that the vomiting began I puked my guts up for 15 hours had to go to ER for dehydration and was miserable for another
day I WOULD NOT RECOMMEND ALCOHOL EVEN IN TINY AMOUNTS WHILE ON THIS DRUG IT LL RUIN YOUR NEXT DAY OR TWO
```

   In the example, the reviewer is just sharing his bad experience when he drank alcohol with his medicine. Therefore, even though facing some bad outcomes, that person gave a good rating.

**Post Processing:**

A sentence could be made of different phrases each representing a unique sentiment. Therefore, for a sentiment classifier, it becomes difficult to classify a sentence containing phrases of different sentiments. If we track examples that are misclassified by the LinearSVC model, we can identify ambiguous reviews that would be useful for the drug manufacturers and the patients. Thus, the ambiguous reviews can be effectively identified by analyzing the complexity of a review to be classified.

Phrases of different or similar sentiments are joined using conjunctions. For the task of ambiguity detection in the reviews, we formed a hypothesis that the number of conjunctions is directly proportional to the ambiguity of reviews.

**Ambiguity Detection:**

To prove our hypothesis that ambiguous reviews contain a higher number of conjunctions, we calculated the number of conjunctions in each misclassified review and found that each ambiguous review contains many conjunctions. The evaluation of the results proved our hypothesis to be true. Below is an example of an ambiguous review with high numbers of conjunctions.

In this example, the customer has many side effects but still rated 10 to the drug. It is an ambiguous review. The number of conjunctions is high in this review as compared to unambiguous reviews.

```
This was and is still a nightmare contraceptive Bought this and used with my boyfriend Used 3 in one night We both had side
effects We are exclusive if not I would have thought I caught something Felt good after using but 2 hours later the excess i
s flowing out leaves me feeling slightly irritated I wash again but still feel a persistent irritation Day 2 he had a weird
sensation while peeing Both of us have itchy genitals and it only gets worse from there I almost feel like a yeast infection
My anal region itches my vaginal area feels dry but it isn t It s just highly irritated I ve used raw yogurt acidophilus pul
ls and so feel irritated Never using this again It s day 4 and these symptoms are still annoying
Number of Conjunction: 8
```

**Conclusion:**

This project used the dataset of a pharmaceutical review website to identify the sentiment of customers towards the drug based on reviews and ratings. After comparing different supervised machine learning models, unigram-bigram LinearSVC proved to be the best model with the best accuracy of 85% to predict the sentiments of the customer. Using Error Analysis, misclassified reviews were also detected.

Along with the sentiment detection, we were able to detect the ambiguous reviews in misclassified reviews by proving the hypothesis that the ambiguity of review is directly proportional to the number of conjunctions present in the reviews. Thus, we concluded that more numbers of conjunctions are used to join phrases which makes the statement ambiguous.

**References:**

Farooq, U., Nongaillard, A., Ouzrout, Y., & Qadir, M. A. (2013, December). Product reputation evaluation: The impact of conjunction on sentiment analysis. In *Proceedings of the 7th international conference on software, knowledge, information management and applications (SKIMA'2013), Chiang-Mai, China* (pp. 590-602)

Gopalakrishnan, V., & Ramaswamy, C. (2017). Patient opinion mining to analyze drugs satisfaction using supervised learning. *Journal of Applied Research and Technology, 15*(4), 311-319. doi:10.1016/j.jart.2017.02.005

Gräßer, F., Kallumadi, S., Malberg, H., & Zaunseder, S. (2018). Aspect-based sentiment analysis of drug reviews applying cross-domain and cross-data learning. Paper presented at the 121-125. doi:10.1145/3194658.3194677

Moraes, R., Valiati, J. F., & Gavião Neto, W. P. (2013). Document-level sentiment classification: An empirical comparison between SVM and ANN. *Expert Systems with Applications, 40*(2), 621-633. doi:10.1016/j.eswa.2012.07.059

The Conversation. (2017, April 06). The problem with online reviews and ratings. Retrieved August 10, 2020, from https://www.smartcompany.com.au/marketing/problem-online-reviews-ratings/