Sentiment Analysis and Summarization of Drug Reviews

Atul Lal*, Pavan Gaddam*, Srivatsa Gunturi*
Stanford University
atulblal@stanford.edu, gaddamp@stanford.edu, sgunturi@stanford.edu

Abstract

Online review sites contain a wealth of information regarding user preferences and experiences over pharmaceutical products. This information can be leveraged to obtain valuable insights using machine learning approaches. In this work we perform analysis to predict the sentiments concerning overall satisfaction and keyword summarization. Online user reviews contain information related to multiple aspects such as effectiveness of drugs and side effects, which make automatic analysis very interesting but also challenging. However, analyzing sentiments and summarization can provide valuable insights, help with decision making and improve monitoring public health by revealing collective experience.

1 Task Definition

This project focuses on sentiment analysis of drug reviews with secondary goal to summarize reviews. Analysis will predict the sentiments concerning overall satisfaction with drug.

1.1 Motivation

Analyzing sentiments of drug reviews can provide valuable insights, help with decision making and improve monitoring public health by revealing collective experience. This system will allow health care providers to identify effective / ineffective drugs more quickly. Patients can learn from experiences of other patients, compare and contrast different drugs and also get a summarized view of review.

1.2 Problem Definition

Given a review for a drug for a condition, predict the sentiment of the review and summarize the review in sentences and keywords.

The AI model uses the drug review data set (drugs.com) from UCI machine learning database for training. For Sentiment Analysis, the baseline implementation will use a simple feature vector and train model based on logistic regression. Our task is to compare the baseline with additional implementations such as 1.) logistic regression with different n-grams model, identifying several stop words and removing them from the reviews and 2.) Using NLP techniques such as stop word removal, lemmatization and stemming to pre-process the reviews and then use Recurrent Neural Network based on Long Short Term Memory cells to predict the sentiment. At the end of all implementations we are targeting to have approximately 90 accuracy score.Our Oracle is a human reading the review and guessing if it is a positive or negative.

Reviews in the data set have 86 words on average with maximum length of 1857 words. To help readers get to the main point quickly and then decide if they want to read the full review, there is a need to summarize the reviews. Summarization refers to the technique of shortening long pieces

CS221 Artificial Intelligence: Principles and Techniques (Spring 2019), Stanford, CA.

of text. The intention is to create a coherent summary having only the main points outlined in the original text. While there are many approaches, extraction-based summarization will be implemented in this project. The extractive text summarization technique involves pulling key words from the source text and showing them in order of importance. The extraction is made according to the defined metric without making any changes to the texts.

2 Infrastructure

We are using Drug Review Dataset (Drugs.com) Data Set available at UCI Machine Learning Repository.

Data Set Characteristics:	Multivariate, Text	**Number of Instances:**	215063
Attribute Characteristics:	Integer	**Number of Attributes:**	6
Associated Tasks:	Classification, Regression, Clustering	**Missing Values?**	N/A

Figure 1: Dataset Description.

The data set provides patient reviews on specific drugs along with related conditions and a 10 star patient rating reflecting overall patient satisfaction.The data was obtained by crawling online pharmaceutical review sites. Following Attributes are available:

1. drugName (categorical): name of drug
2. condition (categorical): name of condition
3. review (text): patient review
4. rating (numerical): 10 star patient rating
5. date (date): date of review entry
6. usefulCount (numerical): number of users who found review useful

3 Approach

There are reviews and ratings in the dataset. We plan to extract features from data and try different supervised learning approaches and predict sentiment based on review provided.

There are two main ways of how to summarize text in NLP:

- Extraction-based summarization
- Abstraction-based summarization

We plan to focus on extraction based approach initially

3.1 Model and Algorithm

We have modeled sentiment analysis and summarization.

3.1.1 Sentiment Analysis

The AI Model in our project performs sentiment analysis of Drug Reviews. Given an input Drug review, our AI model predicts if the review, is positive, negative or neutral. We call these as the labels. Figure2 shows brief architectural overview of our Model. We extract features of a review using common NLP techniques. Each feature carries a weight and our goal is to train the model to obtain the weights.

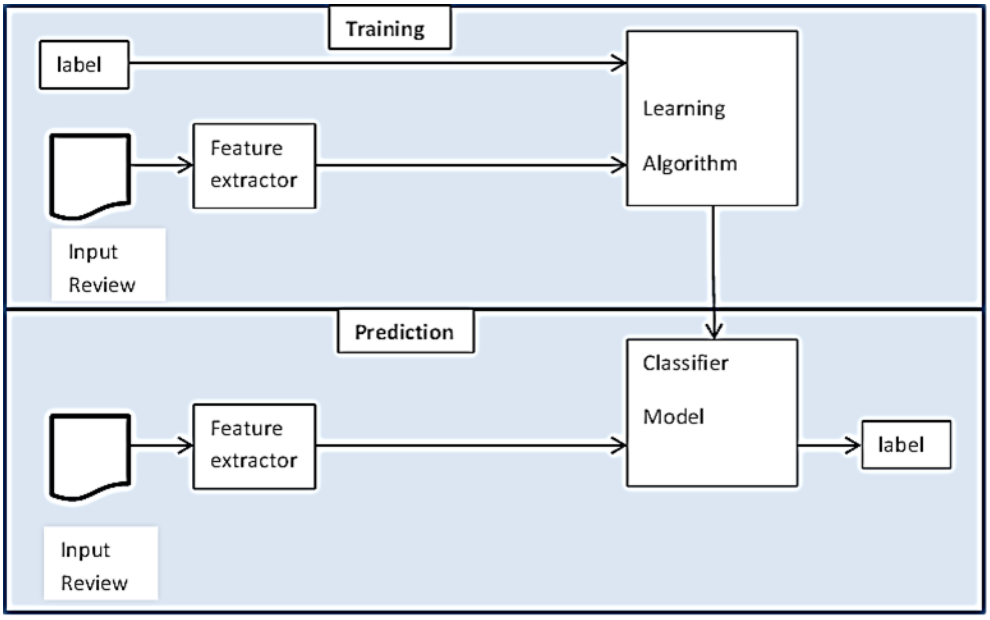


Figure 2: Brief Architectural Overview of our Model, Training and Inference.

Labels In the data set, every drug review has a numerical rating. We used the star rating to define our labels for the training data set. We defined the range of rating for each of the label as below.

Table 1: Labels

Rating	Sentiment
1 - 4	Negative
5 - 6	Neutral
7 - 10	Positive

Input / Output Behavior Output of sentiment analysis model is Positive / Neutral / Negative label for a review

Figure 3: Input / Output.

Features We extract features using a common - Bag of Words (BoW). In addition, we enhance the feature vector by applying some NLP techniques such as stop words and lemmatization. These techniques are implemented as an addition to baseline to achieve further improvement to the baseline. We use NLTK library of python to perform feature enhancement.

Algorithm Baseline - Logistic Regression
Logistic regression is a classification algorithm used to assign observations to a discrete set of classes. There is different type of classes, binary, multi and ordinal. Since our model predicts the review in three different categories, we will be using a multinomial logistic regression model. The cost function of logistic regression model is derived from the non-linear transformation of linear predictors. (given below)

$$J(\theta) = -\frac{1}{m}\sum\left[y^{(i)}\log(h\theta(x(i))) + \left(1-y^{(i)}\right)\log(1-h\theta(x(i)))\right]$$

Figure 4: Cost function of Logistic Regression.

Algorithm such as stochastic gradient descent is generally applied to to obtain weight coefficient of feature vector. In our project, we have used logistic regression as our baseline implementation. To implement logistic regression, we have used python library scikit learn.

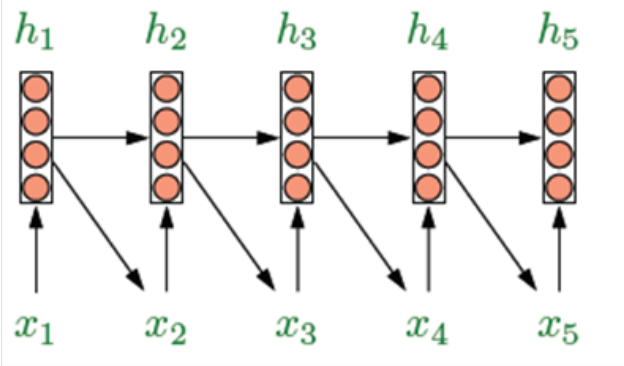
Recurrent Neural Network – Long Short -Term Memory Networks
While logistic regression is a good baseline, it has many shortcomings, it doesn't fully take advantage of the review and understand the words and context. For example, given the following review from the drug review set:

"I am so happy with the samples provided by my Endocrinologist. The only thing I am so sad about is that I cannot afford the prohibitive costs. However, overall this is the best thing I have ever had to make my Irritable Bowel Syndrome-D tolerable. I can go out of the house again without worrying and the terrible stomach pains are a thing of the past."

The star rating for the above review is 9 which means the review should obtain a positive label.But there is a high chance the review might get a negative label due to words like tolerable, prohibitive costs, irritable, pains and sad.

Thus, we need a better alternative to improve the performance of the prediction. After reading further, we found out Recurrent Neural Network (RNN) are a good alternative especially the Long Short-Term Memory (LSTM) Networks.

All recurrent neural networks have the form of a chain of repeating modules of neural network. In standard RNNs, this repeating module will have a very simple structure, such as a single tanh layer.But in LSTM, there are multiple layers interacting in a special way. This helps in remembering long term dependencies good for NLP problems especially when the context is in the beginning of the review.



$$(h_t, c_t) = \mathsf{LSTM}(h_{t-1}, c_{t-1}, x_t)$$

Figure 5: LSTM.

3.1.2 Text Summarization and Keyword Extraction

Automatic text summarization is the task of producing a concise and fluent summary while preserving key information content and overall meaning. In general there are two different approaches, extraction based and abstraction based.

Extraction summarization works by identifying important sections of text generating as it is, thus they only depend on the extraction of sentences/keywords from the original text. The extractive summarization techniques construct an intermediate representation of the input text, calculate the score of the intermediate representation and select a summary based on the importance.

On the other hand, the abstractive techniques aim at producing important material using natural language techniques in order to generate a new shorter text that conveys the most critical information from the original text. We are only considering extractive based summarization technique in this project.

In extractive summarization, we implemented two different approaches. The first approach is based on Term Frequency Inverse Document Frequency (TFDIF). In this method, it gives weights to the words in the documents. This weighting technique assesses the importance of words and identifies very common words in the document and giving them low weights to words appearing in most documents. The weight of each word w in document d is computed as follows:

$$q(w) = f_d(w) * log(\frac{|D|}{f_D(w)})$$

where $f_d(w)$ is term frequecny of word w in the document d, $f_D(w)$ is the number of documents that contain word w and $|D|$ is the number of documents in the collection D.

We used the TFDIF method to extract keywords from a given reviews. Shown below is our implementation of keyword extraction using TFIDF.

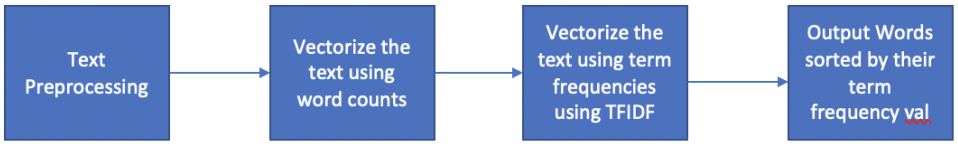


Figure 6: Implementation of Keyword Extraction using TFIDF

In addition to the TFDIF, our second approach is to implement text summarizing using text rank algorithm. Text rank algorithm is based on the page rank algorithm. In page rank algorithm, link analysis of each webpage is done. It assigns numerical weighting to each webpage that has links to other webpages. The numerical weight it assigns is called the page rank. In text rank the link is between sentences. Similarity between any two sentences is used as an equivalent to the web page transition probability. The similarity scores are stored in a square matrix, similar to the similarity matrix used for PageRank. The output of this approach is important sentences from the given review.

We used text rank algorithm to perform text summarization. Shown below is the figure of our implementation of text summarization using text rank

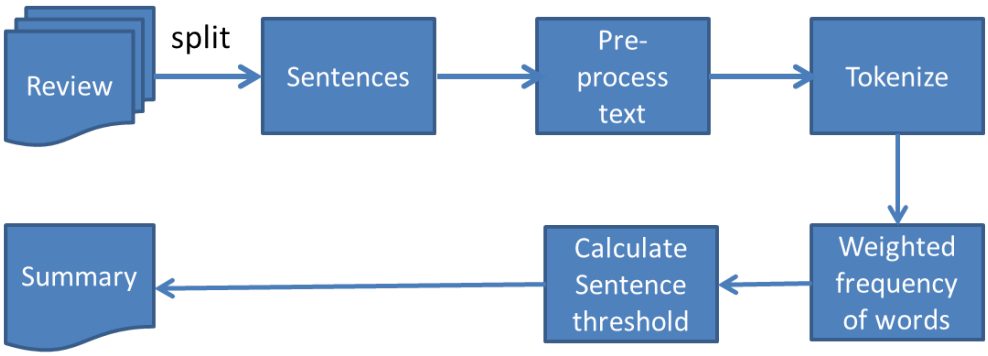


Figure 7: Implementation of Text summarization using Text Rank

Both Text Summary and Keywords will help a potential user in choosing the right drug.

4 Results and Error Analysis

Here are results from experiments.

4.1 Sentiment Analysis - Logistic Regression

The data set consists of 215063 individual reviews.

1. Among these reviews, we considered 75% (161297) as our training data set
2. We also use 25% (40324) of the training data set for training validation and 25% (53766) as our test data set.

We conducted multiple initial experiments using logistic regression using the drug reviews data set.

1. Pre-processed data. For ex. removing different special characters and removing the stop words
2. Tried to identify the step size where we could find the optimum results
3. From the train data identify what percentage of data to use for train and how much to use for validation before running the model on the test data

Initial run results:

The confusion matrix:

Table 2: LR Model Confusion Matrix - uni-gram model

		Predicted Labels		
Confusion Matrix		Negative	Neutral	Positive
True Labels	Negative	9941	282	3274
	Neutral	1385	831	2613
	Positive	1954	386	33100

From the above table, we came to know that:

- We have predicted 43872 (9941 + 831 + 33100) correctly
- We have predicted 3274 negative reviews as positive reviews
- We have predicted 2613 neutral reviews as positive review etc
- We need to work to make negative/neutral reviews correctly to improve our predictability

Re-ran Logistic Regression - this time removed stop words and used tri-gram model Got much improved results

Table 3: LR Model Evaluation

Accuracy Metric	Value
Accuracy Score	0.92
F1 Score	0.86
Precision Score	0.91
Recall Score	0.83

Table 4: LR Model Confusion Matrix - tri-gram model

		Predicted Labels		
Confusion Matrix		Negative	Neutral	Positive
True Labels	Negative	12178	142	1177
	Neutral	687	3046	1096
	Positive	817	158	34465

4.2 Sentiment Analysis - LSTM

Prepared data for LSTM.

Removed outlier reviews, with words over 100

Tokenized review text and converted to sequence of integers

Configured LSTM model as below:

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 100, 160)          480000
_________________________________________________________________
lstm_1 (LSTM)                (None, 50)                42200
_________________________________________________________________
dense_1 (Dense)              (None, 3)                 153
=================================================================
Total params: 522,353
Trainable params: 522,353
Non-trainable params: 0
_________________________________________________________________
```

Figure 8: LSTM Model.

Chose softmax as activation function

Chose binary cross-entropy loss

Trained model.

Training stopped early after 7 epochs. Minimum validation error achieved in third epoch

Figure 9: LSTM Training Loss.

Achieved overall test data loss: 0.346 and Accuracy: 0.888

Table 5: LSTM Model Confusion Matrix

		Predicted Labels		
Confusion Matrix		Negative	Neutral	Positive
True Labels	Negative	10612	1440	2370
	Neutral	811	1686	1187
	Positive	2074	1703	31883

As is evident from table 5 above - achieved much higher accuracy for positive reviews at 89% vs. 74% for negative reviews

4.3 Text Summarization and Keyword Extraction

The following are the steps in implementing Text Summarization:

1. Split the given review paragraph into sentences
2. Convert it to vector representation of each of the sentences
3. Similarities between sentence vectors are then calculated and stored in a matrix
4. The similarity matrix is then converted into a graph, with sentences as vertices and similarity scores as edges, for sentence rank calculation
5. Applying text rank to obtain scores for each sentences
6. A certain number of top-ranked sentences form the final summary

The following is a result for a review when top 2 sentences where choosen to be part of the summary

Review:
"Irsquo;m currently in month 11 of a Pristiq taper I have tried several times over the past 20 years to stop SSRI or SNRI including Effexor and Pristiq but have experienced such horrible withdrawal

symptoms that I have not been able to stop taking them So basically I have been dependant on these drugs for 20 years, not because I really need them, but because my brain reacts violently due to withdrawal This time Irsquo;m reducing the dose extremely slowly and it has been better than previous attempts, although Irsquo;m finding the past few weeks very, very difficult because Irsquo;m down to a quarter of a tablet every two days Irsquo;ve been reducing the dose about 12mg-25mg at a time and staying on that dose for one month This is the most horrible and difficult thing I have ever experienced in my life and my heart goes out to all of you who are suffering and fighting to survive this hell I lsquo;m also astounded, disappointed and angry at all of the doctors and Psychiatrists who are so dismissive of this real condition and who have ignored my fears about the withdrawal syndrome and then labelled me as a lsquo;chronic depressiversquo; I hope one day this comes to public the attention and that the medical profession finally acknowledges the seriousness of this growing problem and that many people need help The same kind of long term FREE medical rehabilitation that drug addicts and alcoholics are given."

Summary:
"reducing the dose extremely slowly and it has been better than previous attempts, although finding the past few weeks very, very difficult because down to a quarter of a tablet every two days"

The following are the steps in implementing Keyword Extraction:

1. Text Pre-Processing to perform Noise removal - involves stop word removal and normalization - involves stemming, lemmatization.
2. Convert text to vector of word of counts
3. sort terms in descending order based on term frequencies to identify top words and this forms the keywords

For the review, the following are the keywords.

Review:
irritable bowl syndrome diarrhea worked great take twice day needed juice side effect noticed one case nausea passed hour make sure drink plenty fluid avoid major constipation

Keywords with TFIDF:
(u'take twice day', 0.244)
(u'drink plenty', 0.234)
(u'bowl', 0.243)
(u'great take', 0.236)
(u'take twice', 0.234)

5 Next Steps

We have tried two approaches for supervised learning to predict sentiment from review. Tried Logistic regression and then LSTM. Removing stops words and using tri-gram model really helped. We feel, LSTM model can to be improved after tuning.

We have tried two approaches for summarization, sentence extraction and key word extraction using TF-IDF and textrank algorithms. Our Dataset lacked the Annotated Corpora. Due to this, we were unable to use machine learning techniques such as LSTM to do a classification type summarization. In future, this work can be extended to generate an annotated corpora which will greatly benefit the research community.

Acknowledgments

We sincerely thank course assistant Jon Kotker for providing support and guidance for the project and Professors Dorsa Sadigh and Moses Charikar for introducing us to the field of machine learning.

References

We referred to the following sources as we learnt about NLP and completed this project

[1] https://blog.insightdatascience.com/how-to-solve-90-of-nlp-problems-a-step-by-step-guidefda605278e4e.

[2] https://arxiv.org/abs/1301.3781.

[3] https://towardsdatascience.com/your-guide-to-natural-language-processing-nlp-48ea2511f6e1.

[4] http://colah.github.io/posts/2015-08-Understanding-LSTMs/#fnref1.

[5] https://ml-cheatsheet.readthedocs.io/en/latest/logistic-regression.html#multiclass-logisticregression.

[6] https://www.analyticsvidhya.com/blog/2018/11/introduction-text-summarization-textrank-python/

[7] https://medium.com/analytics-vidhya/automated-keyword-extraction-from-articles-using-nlp-bfd864f41b34

[8] Text Summarization Techniques: A Brief Survey Mehdi Allahyari et al - https://arxiv.org/abs/1707.02268