Nov, 2018 Project Code : P9

Analyzing Consumer Purchasing Behaviour from Product Reviews, Ratings and Description

End-Term project report submitted for

BACHELOR OF TECHNOLOGY PROJECT (PART-I)

IN

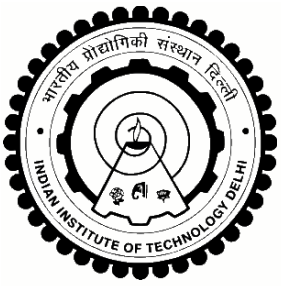
MECHANICAL ENGINEERING

Submitted by

MANAS JOSHI Entry No: **2015ME10108**
SHUBHAM SINGLA Entry No: **2015ME10383**

Under the guidance of

Prof. Kiran Seth & Prof. Achal Bassamboo



Department of Mechanical Engineering
Indian Institute of Technology Delhi
New Delhi 110 016

ACKNOWLEDGEMENTS

We would like to express our sincere gratitude to my supervisor Prof. Kiran Seth and Prof. Achal Bassamboo for providing their invaluable guidance, comments and suggestions throughout the course of this project. We would like to specially thank Prof. Achal Bassamboo for constantly motivating us to work harder Prof. Julian McAuley for providing us the relevant data to start with for the project.

Also, we would like to thank Ms. Anshul (working with Amazon) for her invaluable suggestion and help in scraping the data from Amazon. We would like to thank Ruomeng Cui and Dennis J. Zhang for guiding us towards web scraping from Amazon internal servers.

ABSTRACT

Many online retailers provide real-time inventory availability information. This affects the consumer purchasing behaviour. Example of this is Amazon lightning deals, which are run for limited amount of time with huge discounts. These deals display the real-time percentage of products purchased by the consumer. Based on the inventory availability information, we can predict the consumer behaviour. Apart from this, each product contains reviews as well as ratings for each product which further affect the consumer purchasing decision for the product. Our main objective is to find out how the consumer makes decision when all this information about a particular product is available. As a first step, we need to find the evidence whether review text and review rating together provide different kind of information or one is contained in the other. It seems quite obvious that reviews will contain a lot more information as compared to ratings and the information that ratings might convey will be overshadowed by reviews. This might seem obvious to normal human but for a machine learning program trying to predict the different pieces of information from reviews, it is quite likely that machine learning algorithm won't be able to convey all the information captured by the reviews. From our preliminary analysis, we found that both reviews and ratings capture much more precise information as compared individually. Our next step is to study the behaviour of deals which are re-run by the amazon and understand how the consumer behaves to such sudden change in the deal characteristics.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

Different flash deal websites such as Amazon, Flipkart have become a popular means of selling products online. These websites provide heavy discounts to consumers along with valuable feedback from other consumers. Sellers also find it easy to advertise their products and sell them at discounted prices. To make it more transparent, Amazon provides real time inventory information by means of Amazon's lightning deals. Thus, it is important to understand whether this piece of real time information provided by Amazon by the means of lightning deals affects consumer decision making power and hence, affects their purchasing decision. When making a purchasing decision, consumers generally look at how good the deal is, what is the average rating given to the product, how many people have reviewed it and are the reviews good or not. After analyzing all this information, consumers make decision whether to purchase the product or not. Hence, purchasing decision of a consumer is being influenced by a lot of factors. If the inventory for a particular product is less and the time is also decreasing, this might create a pressure on the consumers to purchase the product. It pushes them into the trap of thinking that the product is good since it has been purchased by a lot of people. Also, the time creates a pressure on the consumer to quickly make the decision before the deal ends. In this thesis, we would like to study how a consumer reacts in such situation. This real time inventory information along with reviews, ratings and description about a product affects the consumer purchasing behaviour and can be analyzed during the lightning deals closely. Consumer infers the quality of deal, that is, whether to purchase it or not, using the available information. By observing past purchasing decision from inventory information and the product reviews, customers can conclude and update their belief about the quality of the deal. This is particularly helpful when they are not sure about the deal or the product before hand. In a lot of cases, customers have a prior idea about the quality of deal. In such cases, customer tend to ignore the information conveyed on the product page. By showing a product's real time

inventory, Amazon's lightning deals are an ideal research context to investigate how the customers behave to different types of information conveyed to them. Lightning deals differ from traditional online sales in mainly two ways. Firstly, lightning deals contains fixed amount of inventory which is consistently displayed revealing product availability. Secondly, lightning deals are generally sold at hefty discounts for a fixed time frame. This creates a pressure among consumers to quickly make the decisions and buy the products. As a first step towards this, we want to analyze the information captured by the review text and rating and see whether both convey the same type of information or not. It seems quite obvious that reviews will contain a lot more information as compared to ratings and the information that ratings might convey will be overshadowed by reviews. Once this fact is established that the review text along with rating provide much richer source of information, we would like to study their combined effect on consumer purchasing behaviour. There are a few deals on Amazon for which the deal time and is increased in between the deal. We would like to study the change in sales for such products and how consumers react to sudden jump in the time. In our initial findings, we observed that review text when combined with rating helps in determining the helpfulness more accurately. This confirms the fact that we should consider both the variables simultaneously and also figure out a way to measure the effect of review through sentiment learning. We collected 1437 data for lightning deals such as how much deal is claimed for each 30s interval out of which 14 deals have time increasing characteristic. After establishing the above same fact, we tried a simple regression model to see how the sales for the products change whose deal time is increased abruptly by Amazon with the time. We have done just a preliminary analysis for the same and the results look promising.

CHAPTER 2
LITERATURE SURVEY

Past studies have explored the impact of inventory information on consumer demand and how to use the inventory information as a strategic lever to reshape demand. Past literature attributes this to herding behaviour. [5] Ruomeng Cui, Dennis J. Zhang, and Achal Bassamboo show that when inventory availability information is given to customers, higher prior percentage claim information attracts more sales in the future. They also show that a higher product rating amplifies the learning momentum and a high actual discount will weaken the learning momentum. They have also shown that a higher product rating will improve the learning momentum. They have used a difference in difference analysis to analyze the consumer behaviour and prove the hypothesis stated above. We try to extend and contribute to the literature by considering the review text along with the rating to deduce consumer behaviour. No one has tried to include both review text and ratings along with other parameters considered by previous mentioned paper to analyze consumer purchasing behaviour. To consider the effect of reviews, [8] Raymond Y.K Lau and Wenping Zhang, Peter D. Bruza along with K.F. Wong tried to determine the positivity and negativity of reviews. They tried to learn the domain specific sentiment lexicons for predicting the product sales. They have shown the feasibility of deriving the sentiment metrics and predicting the sales. Sentiment analysis is one of the most important step in analyzing the review text and the tools provided by the above mentioned authors are quite helpful for our analysis of product sales. Another paper by [6] Eun-Ju Lee and Soo Yun Shin examined how the quality of online product reviews affects the participants' acceptance of the reviews as well as their evaluations of the sources. They show that this effect varies with different product types which can be studied by classifying the products broadly into two categories - experience goods and search goods. None of the authors have tried to include the review text to consider the changes in sales of a particular product. Our attempt is to show firstly that review text along with review rating is an important parameter to gain

useful information and later on, we want to show that for predicting the sales as well as consumer behaviour, review text is an important metric. Stock and balachander (2005) and Debo and Van Ryzin (2009) have shown that the companies have adopted different approaches such as ‘scarcity strategy’ and ‘asymmetric inventory allocation strategy’ to determine the product quality and increase their profits.

Deep neural networks have been found to perform better than conventional regression models across a number of different applications [13]. Since the development of recurrent neural networks, or RNNs, these architectures have been heavily used to classify, process and predict natural language data. Also the introduction of long short-term memory [12], or LSTM, have transformed speech recognition and synthesis, machine translation, language modeling, and image captioning. Given the growing popularity of deep learning for natural language processing applications, we build a deep network consisting of several LSTM layers for predicting product review helpfulness.

From the above literature review, we conclude the fact that people have tried predicting the consumer purchasing behaviour without considering both the review text and rating. These papers offer different innovative operation strategies in response to the consumer’s previous purchasing encounters or their decision making strategically. In our thesis, we have tried establishing the fact that review text along with rating is an important metric. We have also tried to study the deals which are abruptly increased by Amazon.

CHAPTER 3
PROJECT OBJECTIVES AND WORK PLAN

Problem Description and Motivation

E-commerce websites such as Amazon, Flipkart, etc. are becoming popular day by day. These websites are now within reach of everyone. Since the discount offered by these websites is often hefty, everyone enjoys buying new products from these websites. It is a great area of research to study the effects on consumer behaviour due to review text, review rating, product description and real time inventory information conveyed in Amazon lightning deals. If the effects of such things are known, customized suggestions can be shows to customers such that they buy the products. Also, the products having poor sales can be analyzed and the reason behind low sales can be rectified to improve the sales for a particular product. Not only this, it will help in improving the operations by increasing the products in stock for which sales is expected to be high. As can be inferred from the literature review, most of the previous work is focussed on analyzing the review ratings, the price and product description statistics for normal products. In present times, it becomes quite important to consider the review text alongside other parameters to analyze consumer purchasing behaviour and also analyze the products which suffer from abrupt change in deal timea along with their effect on consumers.

Objectives

1. Infer that review text along with rating give more useful information as compared individually.

2. Use the reviews, ratings, real time inventory availability to study the consumer purchasing behaviour for normal products and the products for which deal time is increased abruptly and comparing them.

Methodology

To analyze this behaviour, we collected the data for different Amazon products with the help of a person who has previously worked with Amazon and some of the data was provided to us by Prof. Julian Mcauley, UC San Diego. To prove our claim that review along with rating is a better parameter, we analyzed the raw data available on Amazon website itself and the data we obtained. On analyzing the data manually, helpfulness of a review seemed to be a good metric to prove our claim. We obtained a certain number of features divided on the basis of subjectivity of review and readability. These features were passed through linear regression and a random forest classifier, while we also passed the review text through a deep neural network to predict the helpfulness of a review. Similar kind of experiment was done with ratings to predict the helpfulness. Apart from this, a combination of both reviews and ratings was also used to predict the helpfulness. Our results claim the inference that review text and ratings combined are a better metric to extract relevant information as compared individually. On analyzing the data further, we found that for some deals, amazon increases the deal time abruptly in between. We have done a preliminary analysis on the same to compare this sudden increase in deal time with normal deas and see how their sales are driven by this abrupt change.

CHAPTER 4
THEORY

We have briefly discussed and summarized the different machine learning models used to achieve our objective.

Linear Regression

This is the most simple and yet, powerful algorithm to predict a value which is continuously distributed with the help of features that are independently distributed across the feature space. It tries to fit a hyper place across all the points and learn the coefficients so as to minimize the R-squared values.

$$y = w_0 + w_1 * x_1 + w_2 * x_2 + \ldots\ldots + w_n * x_n \quad \ldots (1)$$

where $w_0, w_1 \ldots\ldots w_n$ are the coefficients learnt by the model across feature space denoted by $x_1, x_2 \ldots\ldots x_n$.

Random Forest Classifier

A random forest is a machine learning estimator that fits a number of decision tree classifiers on different sub-samples of the dataset. It uses averaging to improve the accuracy of the test set and at the same time avoids overfitting of the data.

Deep Recurrent Neural Model (LSTM)

Recurrent Neural Networks are used for solving sequential problems, i.e, problems which have some kind of sequence in them such as sentiment analysis from review text. Since, RNNs don't keep the previous data with them, gated RNNs were introduced which are also incapable of keeping the context of data over long range. LSTMs provide a solution to these problems. Long short term memory, commonly known as LSTM keep

some of the past information intact with the help of different gates known as input gate (i_t), forget gate (f_t), update gate (o_t) and memory cell (c_t). The governing equations for LSTM are given
below along with a block diagram -

$$f_t = \sigma_g(W_f * x_t + U_f * a_{t-1} + b_f) \quad \text{... (2)}$$
$$i_t = \sigma_g(W_i * x_t + U_i * a_{t-1} + b_i) \quad \text{... (3)}$$
$$o_t = \sigma_g(W_o * x_t + U_o * a_{t-1} + b_o) \quad \text{... (4)}$$
$$c_t = f_t * c_{t-1} + i_t * \sigma_c(W_c * x_t + U_c * a_{t-1} + b_c) \quad \text{... (5)}$$
$$f_t = o_t * \sigma_h(c_t) \quad \text{... (6)}$$

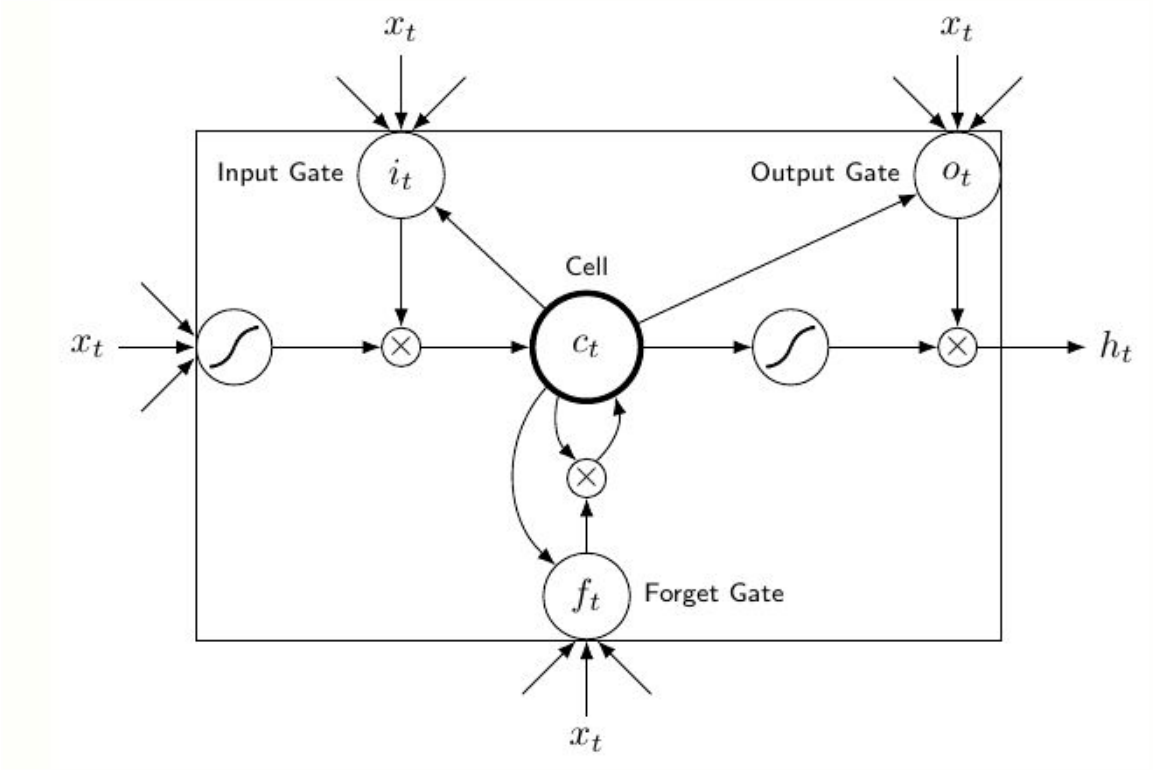


Figure 1 : Schematic of LSTM Cell

CHAPTER 5
PROBLEM STATEMENT - I

We attempt to classify a whether a product review would be helpful to the consumers or not. We define the following measures.

Figure 2 : Sample Amazon Product review

Amazon has a voting system whereby community members provide helpful votes to rate the reviews of other community members. Previous peer ratings appear immediately above the posted review, in the form, "[number of helpful votes] out of [number of members who voted] found the following review helpful." These helpful and total votes enable us to compute the fraction of votes that evaluated the review as helpful. An example is shown in Figure 2.

So for a given review *r*, let n_p be the number of users that vote the review as helpful(positive) and n_n be the number of users that vote the review as helpful(negative). The helpfulness ratio f_r is defined as :

$$f_r = \frac{n_p}{n_p + n_n} \quad \text{... (/*e1*/)}$$

For reviews with a non-zero amount of votes, the value of f_r is in the range [0, 1].

Amazon Dataset

We use the Amazon 5-core product review dataset provided by McAuley et al.[1] . For every review along with the text, we also have the following metadata: reviewer ID, product ID, reviewer name, helpfulness rating ([number of helpful votes] out of [number of members who voted], overall rating, review summary and review time. This data is provided for 24 categories of products for eg. electronics, books, healthcare etc. A sample review is shown in Figure 3.


```
{
  "reviewerID": "A2SUAM1J3GNN3B",
  "asin": "0000013714",
  "reviewerName": "J. McDonald",
  "helpful": [2, 3],
  "reviewText": "I bought this for my husband who plays the piano.
He is having a wonderful time playing these old hymns.  The music  is
at times hard to read because we think the book was published for
singing from more than playing from.  Great purchase though!",
  "overall": 5.0,
  "summary": "Heavenly Highway Hymns",
  "unixReviewTime": 1252800000,
  "reviewTime": "09 13, 2009"
}
```


Figure 3 : A sample review from the review dataset

Model 1

Ghose et al. [4] showed that a threshold of 0.6 for marking a review as helpful minimized the false positive and false negative error rates. In other words, if more than 60 percent of the votes indicate that the review is helpful, then we classify a review as “helpful.” Otherwise, the review is classified as “not helpful”.

Model description

Readability Features

We measured the number of spelling mistakes within each review, and we normalized the number by dividing with the length of the review (in characters). Further, we measured the length of a review in sentences, words, and characters.
Beyond these basic features, we also used empirical metrics for measuring the readability of a text, while none of them is perfect, the computed measures correlate well with the actual difficulty of reading a text. Specifically, we computed the following:

- Automated Readability Index

$$ARI = 4.71\frac{characters}{words} + 0.5\frac{words}{sentences} - 21.43 \quad \text{... (8)}$$

- Coleman-Liau Index

$$CLI = 0.0588L - 0.298S - 15.8 \quad \text{... (9)}$$

where, *L* is the average number of letters per 100 words

S is the average number of sentences per 100 words.

- Flesch Reading Ease

$$FRES = 206.835 - 1.015\frac{words}{sentences} - 84.6\frac{syllables}{words} \quad \text{... (10)}$$

- Flesch-Kincaid Grade Level

$$FKGL = 0.39\frac{words}{sentences} - 11.8\frac{syllables}{words} - 15.59 \quad \text{... (11)}$$

Subjectivity Features

Pang and Lee [3] described a technique that identifies which sentences in a text convey objective information, and which of them contain subjective elements. In particular, objective information is considered the information that also appears in the product description, and subjective is everything else. Using this definition, we then generated a training set with two classes of documents:

A set of "objective" documents that contain the product descriptions of each of the products in our data set.

A set of "subjective" documents that contains randomly retrieved reviews.

We trained the classifier using a Dynamic Language Model classifier with n-grams (n = 8) from the LingPipe toolkit [9] which is based on Bayes Classification of text.

After constructing the classifiers, we used the resulting classification models in the remaining, unseen reviews. Instead of classifying each review as subjective or objective, we instead classified each sentence in each review as either "objective" or "subjective," keeping the probability being subjective $P_r subj$ for each sentence s. Hence, for each review, we have a "subjectivity" score for each of the sentences.

Based on the classification scores for the sentences in each review, we derived the average probability AvgProb(r) of the review r being subjective defined as the mean value of the $P_r subj$ values for the sentences s_1, s_2 ... , s_n in the review r. Since the same review may be a mixture of objective and subjective sentences, we also kept of standard deviation DevProb of the subjectivity scores $P_r subj$ for the sentences in each review.

Non-Neural Model

Next we fit a simple regression model given by:

$$Helpfulness = \beta_0 + \beta_1(Readability) + \beta_2(AvgProb) + \beta_3(DevProb) \quad \ldots(12)$$

Further, we then ran a Random Forest Classifier on the given features.

Deep Neural Network

Google pretrained News 300 dimension Word2Vec embeddings were used to map a a review to a vector space. The review was initially truncated to a length of 200 with all stopwords and punctuations removed. The LSTM layer learns a mapping from this space 200 length sequences of 300-dimensional vectors into 100 dimensional vector space. More concretely, each sentence (represented as a sequence of word vectors) $x_1,...,x_T$, is passed to the LSTM, which updates its hidden state at each sequence-index and encodes all the information. This is then passed to 2 Fully Connected layers reducing the dimension to 50 then 2. We used dropout regularization in every layer to prevent overfitting. The model was made dense since we observed high bias in our results. A validation dataset was used to employ Early Stopping.

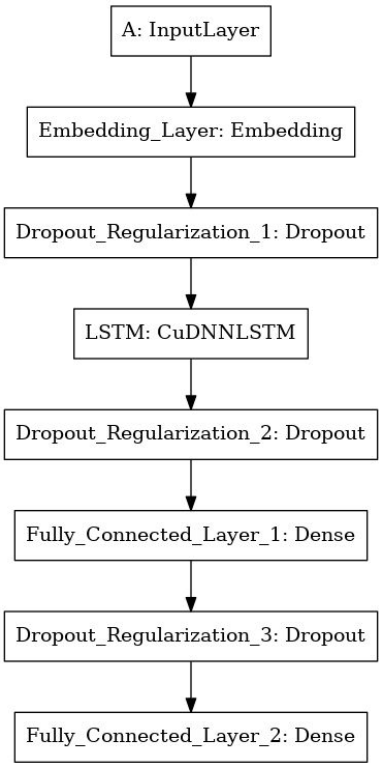


Figure 4 : DNN Network Architecture

Model 2

With results of previous models being unsatisfactory, so, we tried a different approach.

Related works suggest that the reviews in the central region are prone to noise, making the classification task at some boundary difficult, therefore two different thresholds must be chosen, one for classifying review as helpful and another one for classifying it as not helpful.[11]

We choose the thresholds such that, a review is classified helpful if $f_r > 0.80$ and not helpful if $f_r \leq 0.20$.

Also, James et al.[11] suggest removal of skew in the dataset hampers the performance. Therefore, a deskewing operation must be performed. Since this requires large dataset, rather than experimenting on domain-specific reviews we collect data for all categories available. This also helps our analysis since our final experiment will be conducted across all types of products.

We collect all 5-core data available for all the categories. We further filter the reviews, such that all the reviews have total helpfulness votes more than 5. This gives us 12.4 lakh reviews. On observing the distribution of this data, it is seen that it highly skewed towards right.

Hence we perform a deskewing step, selecting a subset of reviews from 5 buckets of the histogram to achieve an approximate uniform data. This leaves us with 4.5 lakh reviews. Figure 2 shows the distribution of review data before and after the deskewing operation.

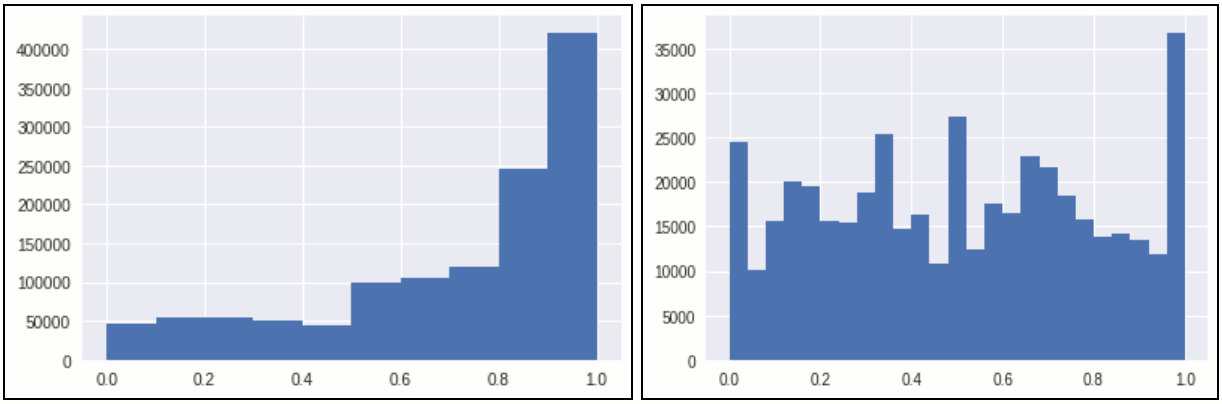


Figure 5 : Dataset distribution of helpfulness ratio: before deskewing(left) and after deskewing(right)

Next, two subsets of 2 lakh reviews were taken, one for a training a helpful review classifier and another one for not helpful review classifier.

Model description

Metadata Features:

- *Product Popularity(1)* - This is approximated by the number of reviews the product has received in our dataset.
- *Reviewer history(1)* - The number of reviews that the particular reviewer has written across all products.
- *Product category(22)* - Data was taken from 23 categories. This is one-hot encoded as a feature
- *Overall rating(4)* - One hot encoding of the overall rating given by the reviewer

Semantic features:

- Number of tokens in the normalized text
- Mean and standard deviation of token lengths
- Number of characters in the review text
- Word2Vec ratio - This represents the fraction of tokens present in the 100000 most frequent words in the Word2Vec model. This is a proxy for f correctly spelled and commonly identifiable english words.
- Part of Speech(POS) tag distribution - This consists of ratio of occurrences of different POS tags - adjectives, adpositions, adverbs, conjunctions, nouns, numerals, verb and unknown.

TF-IDF featurization

We use term frequency-inverse document frequency (*tf-idf*) to feature the review text in our dataset.[7] It was originally developed for information retrieval but it has been proven to be effective for natural language processing tasks. [10]
Suppose we a collection of *M* documents, *C = {d_0 , d_1 , d_2 … } ,* each document d_i containing n_i tokens *{t_0 , t_1 , t_2 … } .* For each term t_j in document d_i we calculate the following:

$$tf - idf(t_j, d_i) = tf(t_j, d_i) \times idf(t_j) \quad \text{...(/*e11*/)}$$

$$tf(t_j, d_i) = \frac{f(t_j, d_i)}{n_i} \quad \text{...(/*e12*/)}$$

$$idf(t_j) = log(\frac{1 + M}{1 + df(t_j)}) + 1 \quad \text{...(/*e12*/)}$$

Further before featurization, the words are stemmed using the Snowball Stemmer, then we compute the tf-idf vectors for each review using the top 12000 unigram, bigram and trigrams.

LSTM Output

Each token in review text is encoded to a 300 dimensional vector using Google pre trained Word2Vec model. The normalized text is padded(truncated accordingly) to a length of 200. This is then passed through two stacked LSTM layers which encode the review text to 50 dimensional vector.

Fully Connected Layers

Next the output from previous layers is concatenated and passed through 3 dense layers with hidden units 75, 50 and 30 having Leaky Relu activation function.

We used Dropout regularization in every dense layer to prevent overfitting. The model was made dense since we observed high bias in our results. A validation dataset was used to employ Early Stopping. We use the binary cross-entropy loss and the l2 regularization term at our final sigmoid layer as our objective function, which is optimized over training iterations using the Adam algorithm. Figure 3 shows the schematic diagram of the architecture.

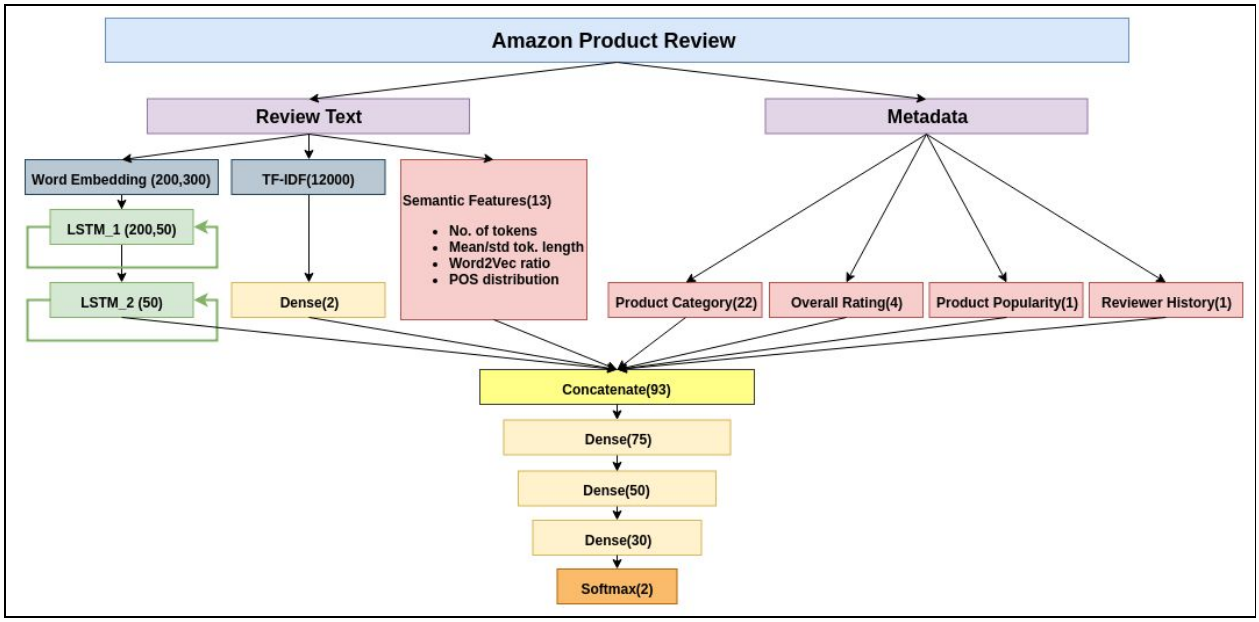


Figure 6: Architecture of Deep Neural Network

CHAPTER 6
PROBLEM STATEMENT - II

We attempt to analyze the sales of special kind of lightning deals and their impact on consumer.

Data Scraping

Our first task is to scrape the data of lightning deals i.e the percentage of deals claimed and the time remaining for each product every minute from the Amazon website.

We first attempt to screen scrape the from Amazon pages. Screen scraping is the process of collecting screen display data. For fast scraping, we use requests library in python but this resulted in Amazon giving blocking the IP due to high number of requests per minute. Next we turn to Selenium, which is a web based automation tool. It is able to scrape the data without getting blocked as it emulates a browser but since we need to crawl through pages to collect the data, we had a limit on number of pages that can be crawled in a minute. Also, this method was dependent on how Amazon decides placement of different products on different pages. Therefore, we had multiple cases where the product had missing values or addition of products in middle of day.

We needed an efficient way of getting product data every minute for large number of fixed deals. This was achieved by sending AJAX requests to Amazon server using XMLHttpRequest and Fetch Api. We were able to get per minute data for at least 300 products within 5 seconds using this.

We further had difficulties where the data collected was corrupted or had some missing data points . A number of iterations were required to get a good quality data.

Modelling Deals

We observe a set of deals which the deal time increases abruptly while the deal is in progress. We try to study them and the consumer behaviour as a secondary task. We scrap the data from Amazon India of 1437 lightning deals spread across 6 days from November 4, 2018 to November 10, 2018. Out of these 1437 deals, 847 deals were started again or observed a sudden change in deal time. Further, 14 out of 847 deals were of our interest. We couldn't carry out a proper analysis but we have tried to establish the fact that these deals behave differently as compared to the normal deals. Due to a sudden spike in the remaining deal time, we expect to observe a decrease in sales. We ran a simple linear regression model to predict the deals claimed for 30 minute and 1 hour time period across all the deals and all the times for which deals were running. This generated 63548 data points in total and 162 data points for the special kind of deals.

Linear Model

For preliminary analysis, we have run a simple regression model given by,

$$y_{i,t} = w_0 + w_1 \times C_{i,t} + w_2 \times x_{1t} + w_3 \times x_{2t} + w_4 \times x_{3t} + w_5 \times x_{4t} + w_6 \times z_{i,t} + w_7 \times z_{i,t} \times t_{rem} + w_8 \times t_{rem}$$

...(16)

Here, $y_{i,t}$ denotes the deal claimed for product i for time interval of t

$C_{i,t}$ denotes the deal claimed at the beginning of time interval for product i

x_{1t}, x_{2t}, x_{3t} and x_{4t} denote the number of reviews, average rating, actual discount and deal discount for the product i respectively.

$z_{i,t}$ is a dummy variable that accounts for the time changing attributes of the deal. It attains a value of 1 for all the times after which the deal time was increased and 0 otherwise.

CHAPTER 7
RESULTS AND DISCUSSIONS

Model 1

Traditional Models

1. **Helpfulness Ratio as a Continuous Value**

Table 1 - Linear Regression

Features	**Mean Square Error**	**R^2 value**
Review_Features	0.03	0.06
Ratings	0.03	0.15
Review_Features + Ratings	**0.03**	**0.20**

2. **Helpfulness Ratio as a Discrete Value (0 - Not Helpful, 1 - Helpful)**

Table 2 - Other Models on Class 1

Models	**Precision (Class 1)**	**Recall (Class 1)**	**F1 Score (Class 1)**
Random Forest on Review_Features	0.929	0.988	0.985
Logistic Regression on Ratings	1	0.926	0.961
Random Forest on Review_Features + Ratings	0.936	0.96	0.95

For both the above tables, we have observed very good results but we later observed that our data was skewed, so these results wrongly portray that good dependence of helpfulness ratio on the feature vector clear from performance measures of Class 0

Table 3 - Other Models on Class 0

Models	Precision (Class 0)	Recall (Class 0)	F1 Score (Class 0)
Random Forest on Review_Features	0.296	0.059	0.098
Logistic Regression on Ratings	0	0	0
Random Forest on Review_Features + Ratings	0.3	0.177	0.22

Data Distribution

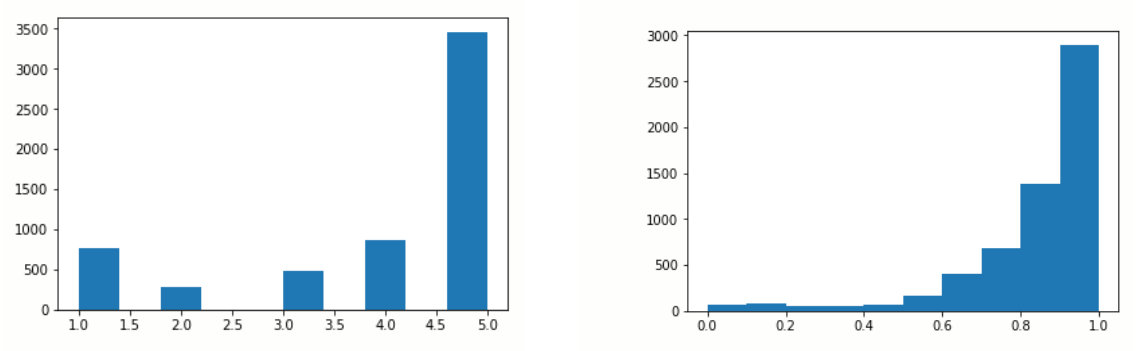


Figure 7 : Dataset distribution of ratings (left) and helpfulness ratio

Deep Neural Network

The new dataset created has the following distribution (Binary Classes)

Table 4 - Dataset

Dataset	Not Helpful Reviews	Helpful Reviews
Train Data	8595	11068
Dev Data	2121	2795
Test Data	2693	3452

Accuracy on Test data = 67%

Table 5 - DNN Model on Class 0

	Precision (Class 1)	Recall (Class 1)	F1 Score (Class 1)
DNN on reviews	0.69	0.65	0.67
Logistic Regression on Ratings	0.67	0.63	0.65

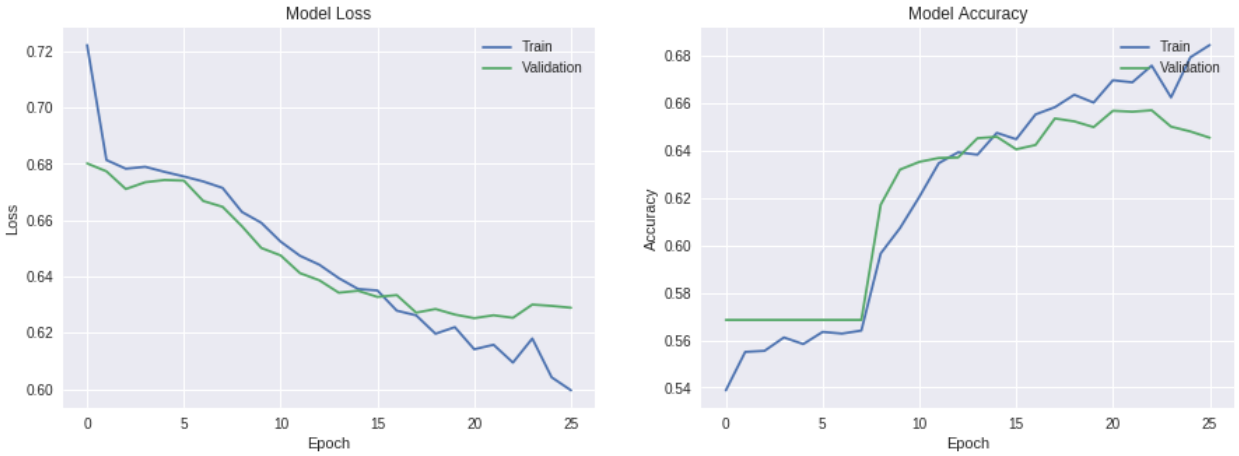


Figure 9 : Model Training : Model Loss vs Epochs(left) and Model Accuracy vs Epochs(right)

Table 6 - DNN Model on Class 1

	Precision (Class 0)	Recall (Class 0)	F1 Score (Class 0)
DNN on reviews	0.587	0.632	0.608
Logistic Regression on Ratings	0.56	0.60	0.58

Though the deep neural network outperforms ratings in predictive power, but it is far from satisfactory.

It was observed that for our Amazon dataset, the total number of unique words present were 80474 out of which 42552 were not present in the Word2Vec model, since they

were either spelling mistakes or brand names. These words are assigned a zero vector in our model.

We try to use a different set of word embeddings called Fasttext provided by Facebook Research which is capable of forming vectors of unknown words. These embeddings rather than storing a vector for every word, they store the vector for n-grams of words and a vector representation of a word is created using these.

The results don't show any significant changes, with similar precision and recall values.
Accuracy on Test data = 65%

We further try to use a spell corrector to correct the spellings in the reviews. This also performs poorly.
Accuracy on Test data = 65%

Model 2

The deep neural network model for classifying good reviews gives a **test set accuracy of 80.5%** and **AUC of 0.88.** The bad review classifier gives a **test set accuracy of 83.7%** and **AUC of 0.90.**

While when trying to classify good reviews with just the ratings, gave a **test set accuracy of 76.5** and **AUC of 0.79** and similarly classifying bad reviews gave a **test set accuracy of 77.5** and **AUC of 0.81 .** This classifier just finds a separating hyperplane for prediction.

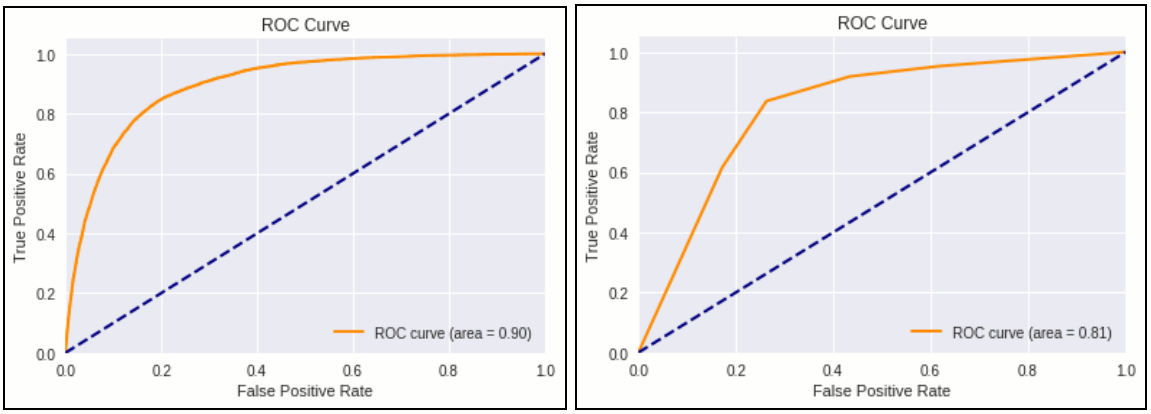


Figure 10 : ROC curve for bad classifier: ratings and review features(left) and rating features(right)

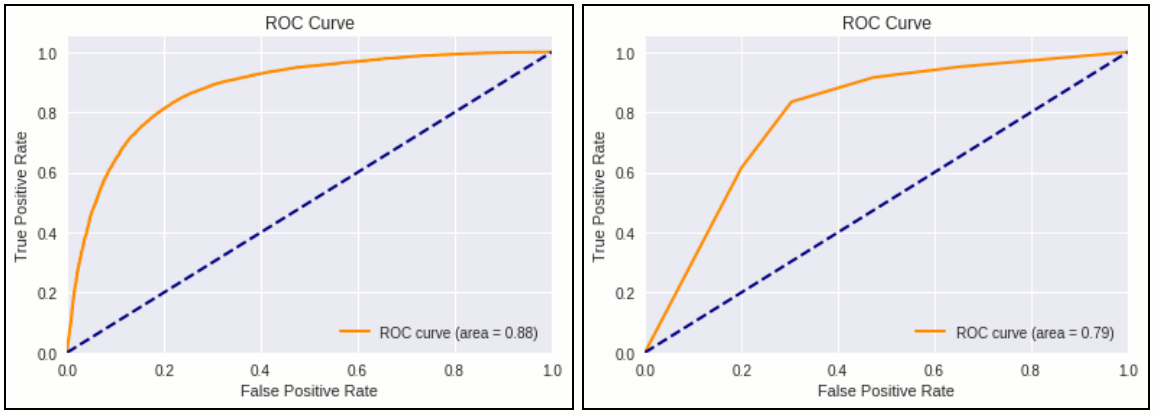


Figure 11 : ROC curve for good classifier: ratings and review features(left) and rating features(right)

Result Summary

Table 7

Features	**Test Accuracy**		**AUC**	
	Good Classifier	Bad Classifier	Good Classifier	Bad Classifier
Only Ratings	76.5	77.5	0.79	0.81
Reviews and Ratings	80.5	83.7	0.88	0.90

It can be seen that reviews and ratings give us higher predictive power compared to just ratings.

This shows us that this approach for analysing reviews is better compared to the previous one. This also favours our hypothesis that the information in reviews and ratings is more than the ratings alone.

Model 3

OLS Estimator

We ran an OLS Estimate to predict the deals claimed across a time period using Difference in Difference approach. The results obtained are not satisfactory and have been mentioned below. We will be working on collecting more data that would be useful to us and do a rigorous analysis on that. We ran the OLS on 164 data points which is skewed data set since it contains 24 data points of one type and 140 data points of the other. These 164 points were chosen out of 63467 data points.

Table 8 - OLS Results

Variables	Coefficient	Std error	t-value	P > | t-value |
x_0	-7.153	8.685	-0.824	0.411
x_1	-0.0616	0.033	-1.885	0.061
x_2	0.0012	0.001	2.157	0.033
x_3	1.5437	1.753	0.881	0.380
x_4	0.0358	0.108	0.332	0.740
x_5	0.0939	0.126	0.747	0.456
x_6	2.6091	3.9	0.669	0.504
x_7	-2.85×10^{-7}	1.43×10^{-7}	-0.2	0.842
x_8	-1.05×10^{-8}	3.98×10^{-8}	-2.641	0.009

R-squared Value - 0.114

CHAPTER 8
CONCLUSIONS AND FUTURE WORK

From our experiment, it can be concluded that both review text and review rating combined provides a better measure of helpfulness of reviews. While analyzing the consumer behaviour, we can't ignore any one of them since both of them individually provide substantial information to predict the consumer purchasing behaviour. We are able to increase our accuracy for predicting the helpfulness of review text using both the rating as well as review text. There is a significant increase in accuracy while involving both the review text and rating as compared individually.

On Analyzing the scraped data from Amazon, we observed some special deals for which the deal time increases abruptly while the deal is in progress. We did a preliminary linear regression analysis on these deals and observed that their sales probably change due to increased deal time and hence, consumers behave differently to the phenomenon.

As a part of future work, we would try to determine the effect on sales of these special deals and the change in consumer purchasing behaviour in much more rigorous form. We will try to observe how the consumers react to such deals and how is their purchasing behaviour different as compared to the normal deals. We will also try to establish some connection between the sales pattern of a normal product and these special products. This would help in managing the operations better and more efficiently.

CHAPTER 9
REFERENCES

[1] Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering, R. He, J. McAuley, *WWW*, 2016

[2] W.H. DuBay, The Principles of Readability, Impact Information, http://www.nald.ca/library/research/readab/readab.pdf, 2004.

[3] B. Pang and L. Lee, “A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts,” Proc. 42nd Ann. Meeting of the Assoc. for Computational Linguistics (ACL ’04), pp. 271-278, 2004.

[4] Estimating the Helpfulness and Economic Impact of Product Reviews: Mining Text and Reviewer Characteristics Anindya Ghose and Panagiotis G. Ipeirotis, IEEE, 2011

[5] Learning from Inventory Availability Information: Evidence from Field Experiments on Amazon; Ruomeng Cui, Dennis Zhang, Achal Bassamboo, Forthcoming Management Science, 2016, https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2868218

[6] When do consumers buy online product reviews? Effects of review quality, product type and reviewer’s photo; Eun Jee, Soo Yun Shin IEEE 2014

[7] Christopher D. Manning, Prabhakar Raghavan and Hinrich Schütze, Introduction to Information Retrieval, Cambridge University Press. 2008.

[8] Learning Domain-Specific Sentiment Lexicons for Predicting Product Sales; Raymond Y.K Lau, Wenping Zhang, Peter D. Bruza, KF Wong, IEEE 2011

[9] Alias-i. 2008. LingPipe 4.1.0. http://alias-i.com/lingpipe (accessed October 1, 2008)

[10] T. Joachims. A probabilistic analysis of the rocchio algorithm with tfidf for text categorization. Technical report, DTIC Document, 1996.

[11] Predicting Amazon Product Review Helpfulness. UCB, 2016

[12] S. Hochreiter and J. Schmidhuber. Long short-term memory. Neural computation, 9(8):1735–1780, 1997.

[13] Y. Kim. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882, 2014.