1. Introduction

As an online clothing retailer, understanding customers' thoughts is an important stage to improve their business. A recent decline in customers' decisions to recommend their products has been noticed. This report aims to find out factors affecting customers' recommendation level in order to figure out aspects they need to improve using data analysis. The report has four stages: (1) understand data by exploratory data analysis to extract important variables graphically, and deal with missing values; (2) build a logistic regression model as a benchmark model. In this step, a validation analysis is provided in order to evaluate our benchmark model and analyse strengths and weaknesses for further improvement; (3) two attempts are applied to get an optimized model with a better validation result for model improvement; (4) find out how recommendation is affected by departments and product types, and if it is affected by age. Suggestions are provided for the company to attract more customers based on the results we obtain.

2. Data Understanding

2.1 Missing Values

To better understand data, the first step is data cleaning. One of the biggest problems in data cleaning is to deal with missing values. In this part, we will first identify missing values, then apply methods to deal with missing data, last provide explanations and reasons in terms of the treatments we've done.

After checking the missing value, it can be found that most of the missing data are stated under the text and the title of the customer's review, which the number counts are 602 and 2638, respectively. What's more, only 9 missing values are under the division, department which the products are in and the type of the products.

As the number counts of missing values in variables "Division", "Department", and "Type" are small, in most cases, they can be deleted. We don't delete them since the reason why data goes missing in these three terms may depend on other fully observed terms (e.g. Age). Deleting missing values requires an assumption of Missing Completely at Random (MCAR), which states that missingness is completely unsystematic (Rubin, 1976). However, we can assume that missingness in these three terms is independent by themselves. For example, a customer who does not have any preference of types does not depend on another customer's preference of types. Thus, we can make Missing at Random (MAR) assumption which states

that missingness is related to one or more other measured variables but is independent to itself (Rubin, 1976). As a result, a single imputation approach can be applied. This approach is to fill missing values in rather than removing them. We use this method since it keeps the full sample size which leads to lower bias rather than the deleting method, but it will also cause other different kinds of bias which will be discussed later. Since the variable types of these features are different, we will apply specific treatments for them.

2.1.1 Fill in Missing Values with Mode

As there are different categories under the division, department which the products are in and different product types, we can assume that the missing data for them can be filled by the most common one which is the mode imputation.

2.1.2 Fill in Missing Values with a Blank

In order to decrease the influence of the missing value under the text data, we choose to replace them simply by a blank. Because using any other words like “NaN” to fill in the missing data would possibly affect the performance of the model since it will become a feature by text analytics.

2.2 Exploratory Data Analysis

Exploratory Data Analysis (EDA) is an approach to give us an initial view of data. It helps us gain a better understanding of the data, thus provides an intuition for feature engineering.

2.2.1 Dependent Variable

Our target variable is whether the product is recommended or not. Figure 1 shows the number of customers who recommend the product and do not recommend the product. The numbers of customers who recommend the product are 13,547, while the numbers of customers who do not recommend the product are 2,893. We find that we have imbalanced classes with a high proportion of customers who recommend the product, whereas the proportion of not-recommended customers is low. This observation is useful for model evaluation and will be discussed later.

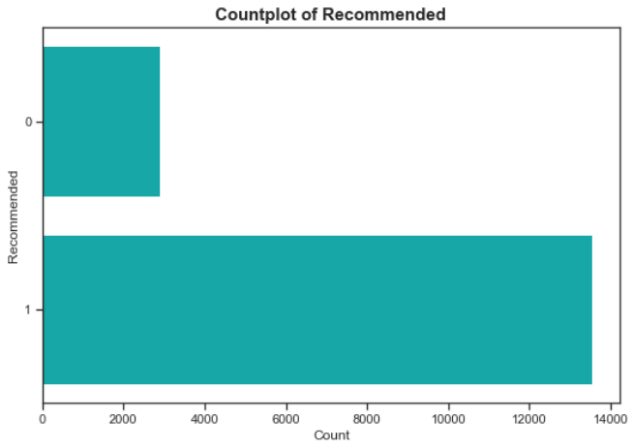

Figure 1. Countplot of Recommended

2.2.2 Descriptive Summary

The descriptive summary presents some descriptive analysis such as mean, standard deviation, maximum, minimum, skew, and kurtosis of "Age", "Rating", "Positive Feedback Count" (Table 1). The mean of the age group is 43 with a maximum of 99 and a minimum of 18. The skew and kurtosis of "Age" suggests that data is right skewed. The mean of the rating score is 4.2 in range of 0 to 5 which probably suggests that most customers recommend the product. Among 16,440 counts of positive feedbacks, the mean is only 2. It probably means less customers prefer to like comments. A detailed analysis will be provided in terms of each variable.

Table 1. Descriptive Summary

	Age	Rating	Positive Feedback Count
count	16440.000	16440.000	16440.000
mean	43.196	4.204	2.564
std	12.273	1.106	5.911
min	18.000	1.000	0.000
25%	34.000	4.000	0.000
50%	41.000	5.000	1.000
75%	52.000	5.000	3.000
max	99.000	5.000	122.000
skew	0.532	-1.328	6.843
kurt	-0.081	0.846	77.875

2.2.3 Independent Variable – Rating

"Rating" is the product rating given by customers. Figure 2 shows the number of customers at each rating score. Most customers rate the product as 5. To understand the rating criteria, we need to see the relationship between rating score and recommendation level. Shown in Figure 3, as rating score increases, recommendation level is increasing. We can conclude that a rating score of 4 or 5 indicate a highly recommended level, while a rating score of 1 or 2 means the product is not recommended, a rating score of 3 is in the middle. Rating criteria is useful to judge customers' recommendation level for analysing their preferences on departments and product types. We find that the correlation between rating score and recommendation level is 0.79 which means they are quite related to each other.

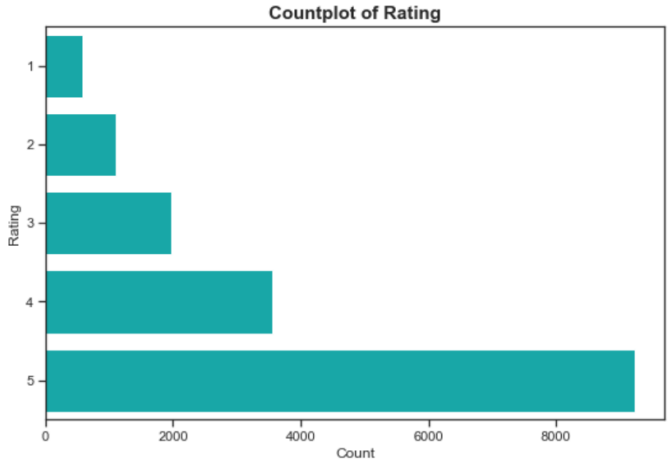

Figure 2. Countplot of Rating

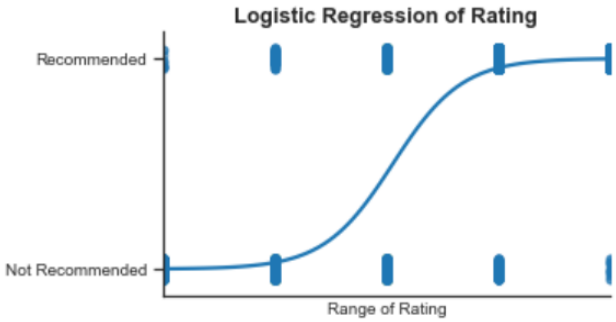

Figure 3. Logistic Regression of Rating

2.2.4 Independent Variable – Age

For the data of the age group, we aim to test whether it is normally distributed by plotting a distribution histogram. Figure 4 shows that most customers are distributed at 35 - 45 years

old. There is a positive skewed distribution with tails on the right-hand side. This also can be supported by the descriptive summary in 2.2.2.

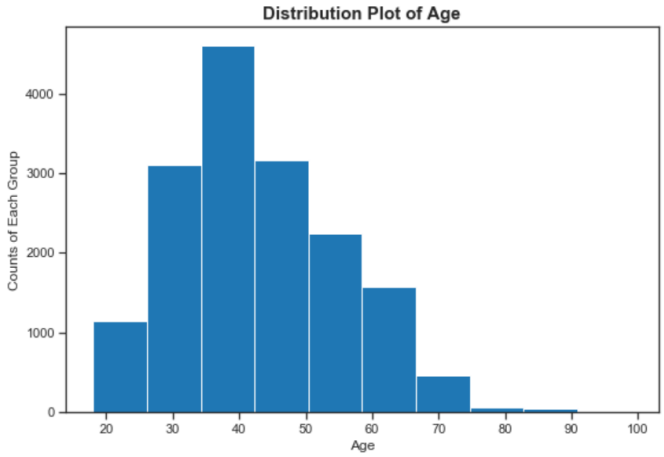

Figure 4. Distribution Plot of Age

Figure 5 shows the recommendation level in each age group. As age increases, the recommendation level is increasing. This means that old people tend to recommend the product while young people are more fastidious. In addition, since the number of old customers is relatively small, there exist some outliers in the old age group. The feedback provided by old customers is also less than feedback provided by young customers. However, we find the correlation between age group and recommendation level is only 0.03 which means they are not quite related.

Figure 5. Logistic Regression of Age

2.2.5 Independent Variable – Department and Type

From Figure 6, it can be observed that the most popular sales of departments are tops, where the number of customers who recommended tops is also the most. However, the percentage of customers who recommended tops is not more than that of dresses. This indicates that tops may be an important factor which affects recommendation level.

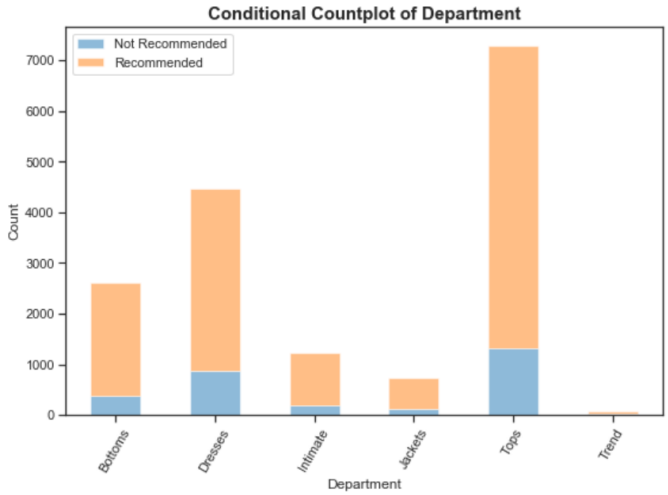

Figure 6. Conditional Countplot of Department

By looking at Figure 7, the most popular sold of type is the dress but not the top. The reason for the high sales volume of tops is probably because of its variety of types. Since the sales of tops are disperse, the factors that may affect recommendation level could vary. In contrast, sales of dresses are more concentrated and the number of customers who recommend dresses is more stable.

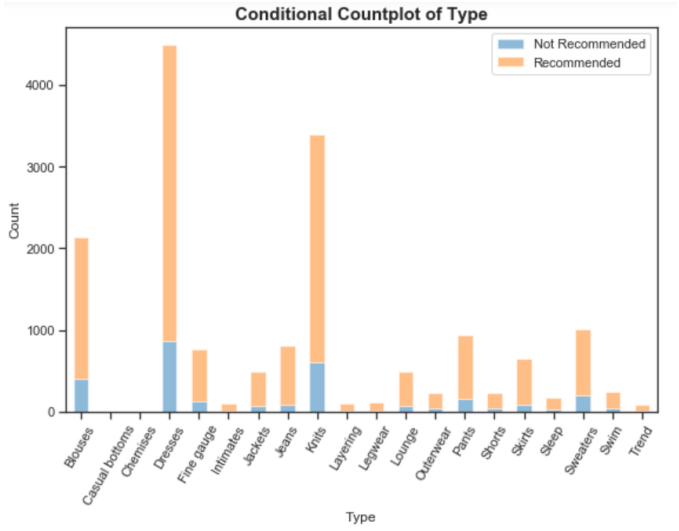

Figure 7. Conditional Countplot of Type

As discussed, the rating score is high when customers tend to recommend the product and the department. According to Figure 8, most customers give 5 scores to dresses, jeans and knits, the numbers of the highest rating are almost double than other scores. In Figure 8, the department of tops is most preferred by the customers and has the highest score of 5 compared to other departments.

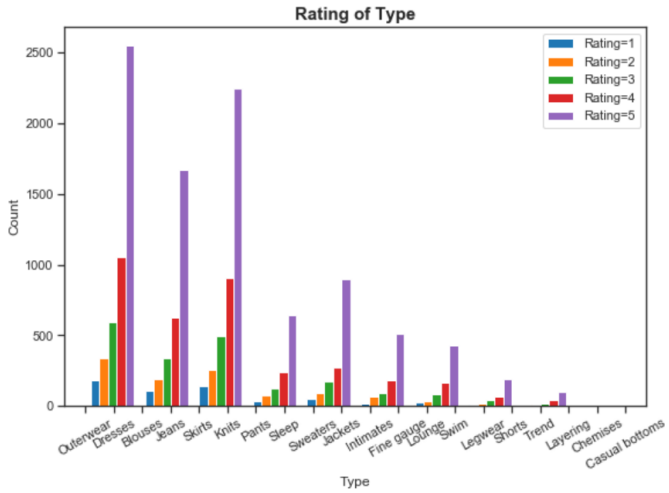

Figure 8. Rating of Type

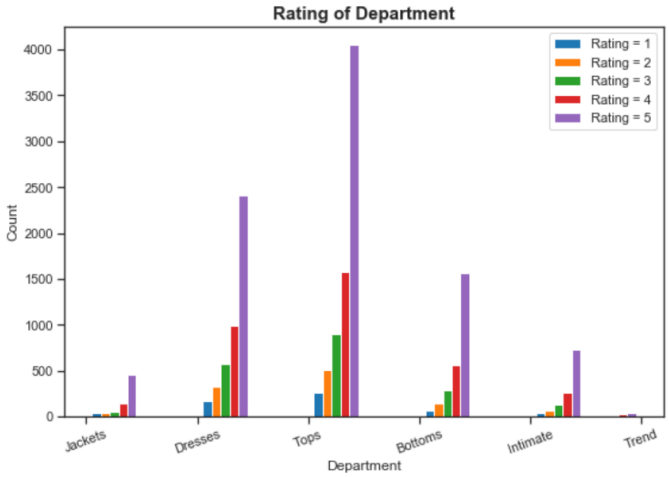

Figure 9. Rating of Department

3. Methodology

3.1 Benchmark Model

3.1.1 Logistic Regression

To predict whether the product is recommended or not, we need to build a simple logistic regression model as it constrains the probability between zero and one rather than the unconstrained linear regression model. In this case, the response variable is {Recommended, Not Recommended}, where Y = 1 if it is recommended and Y = 0 if it is not recommended. We only determine text data under the review text from the customers belonging to which class to build a simple logistic regression.

We use the Bag-of-Words method to transfer text data into a numerical representation for the customer reviews. This method treats each word as a feature and counts how many times the word exists in a string. In this case, we set “500” for “max features” which means we are looking at top 500 types of words, and all other parameters set to default. Then we build a logistic regression model with “solver” set to “liblinear” which means L1 penalty and all other parameters set to default.

3.1.2 Validation Analysis

For model evaluation, we use a confusion matrix to see the performance of our model. A confusion matrix counts the number of true negatives, false positives, false negatives and true positives. For example, true positives are the number of customers who predicted to recommend the products are actually recommend while false positives are the number of customers who predicted to recommend the products are actually not recommend. Our purpose is to see the percentage of our prediction that is actually correct. As shown in Figure 10, 95.1% of the customers who are predicted to recommend our products actually recommend our products.

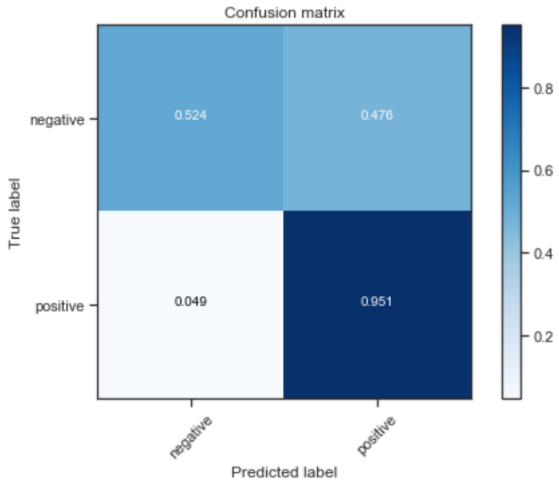

Figure 10. Confusion Matrix of Benchmark Model

However, as shown in the EDA process, we have imbalanced classes with a large number of samples in “Recommended” class. The rate for samples in “Recommended” class is less meaningful since it would be always big. Therefore, we introduce F1 score which is the harmonic mean of precision and recall in order to evaluate our model. In this case, precision shows the rate that customers predicted in “Recommended” class are correctly predicted. Recall shows the rate that customers actually in “Recommended” class can be predicted. F1 score can be calculated using the formula:

$$F1\ score\ =\ 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

We use harmonic mean since it penalizes outliers. However, F1 score here is still for "Recommended" class, so we choose a weighted average F1 score as our evaluation metric. F1 score is more useful than accuracy since we have imbalanced classes.

3.1.3 Analysis of Validation Results

The weighted average F1 score is 0.869 where it reaches its best value at 1 and worst value at 0. Since 86.9% of our prediction is accurate, our benchmark model did a good performance but still needed to be improved. Strengths and weaknesses will be discussed in the following.

Strengths

We use logistic regression to predict whether the product is recommended or not. The advantage of logistic regression is that it is useful and simple that does not require many computation resources, and it can be easily interpreted. Our performance can be measured easily using logistic regression as well. Compared to linear regression, logistic regression constrains the probability between zero and one which is more suitable for our case. In many cases, a simple model performs better than a complex model since a complex model may estimate many parameters, the coefficient of those parameters might be sensitive.

Weaknesses

There are still some drawbacks for our model. One drawback is that we only use customers' reviews as our predictor. When dealing with this text data, we use the Bag-of-Words method to see the top 500 types of words, but among these 500 types of words, not all of them are important. Some words like "just", "size", "small" are indistinct to show the recommended level. As some parameters chosen are not important, the numbers of customers who are predicted to recommend products may not be accurate enough. Ignoring useless words is the thing we need to consider improving our model.

Another weakness is that too many parameters may lead to an overfitting model. In our model, 500 words have been used as variables. An overfitting model leads to low bias with higher flexibility, but it does not mean the model will perform well. It may lead to higher variance with high training error. Therefore, getting an optimal fitted model with appropriate number of parameters is the way to improve our model.

Final weakness is that when building the benchmark model, the hyperparameters of logistic regression are not optimized. We only set “solver” to “liblinear” and keep all other parameters default. The “liblinear” solver may not be the best, there are other hyperparameters such as the inverse of regularization strength, the type of penalty, whether to fit an intercept and the class weight that can use cross validation to be optimized.

3.2 Improving Benchmark Model

3.2.1 Improvement Attempts

3.2.1.1 First Attempt

Before the first attempt of the improvement, the missing values in the dataset have been simply filled. However, there are some variables like the division, department and type of the products which are not presented as numerical and cannot be used in the model. Thus, we use one-hot encoding to transfer the strings into integers. What’s more, as there are 20 types of products, we create a new class named “other” to sum the classes which are less than 20.

Next, as we want to see whether different age groups will influence the customer’s decision, we divided the age of the customers into three groups: customer’s age from 18 to 30 will be included into the age group of young, customer’s age from 31 to 60 will be included into middle age and customer’s age from 61 to 99 will be included into an old age group . Then, we get dummies for those age groups and add three new features of “Young”, “Middle”, and “Old”, where 1 means that the customer’s age is under that group and 0 means they are not.

Finally, for text analytics, since we find that using the Bag-of-Words method to deal with the customer’s review has lots of disadvantages, we change the method to Term Frequency Inverse Document Frequency (TF-IDF) so that we can get more important words than that we get by using former method. According to Ramos (2003), the TF-IDF method determines the relevance between a given word and a particular document. The TF-IDF numbers of a common word in one document will be higher than that of articles or prepositions. Compared to the benchmark model that there are so many meaningless words of feature noise to the model, we want to make the features more specific and positive. Thus, we reduce the max features from 500 to 20 for getting the 20th most relative word frequency in customers’ review and the review text’s title excluding meaningless words like “the” or “a”. What’s more, since there are some repetitive words in the review text and review title, we calculate the mean frequency of the same words and pick some positive ones to set as new features.

3.2.1.2 Second Attempt -- Optimized Model

After the first modification, it can be seen that the new age-group features we add have negative coefficient with the model. Thus, we decide to delete these features for better performance. What's more, the feature of product ID is dropped as we found that it is meaningless as for only presenting the Id for each product. For the positive feedback count from the customers, there are too many zeros under this feature. In order to see the impact of these zeros, we create a new variable which represents whether the positive feedback is zero or not.

Another big improvement in the second attempt is that we add the sentiment analysis to the customers' review text and the title of the products. Because we think the opinion from customers is the most straightforward representation for whether they decide to recommend a product. However, the TF-IDF method can only recognize the frequency of the words instead of the sentiment of the words. We use Valence Aware Dictionary and sEntiment Reasoner (VADER), it can recognize special sentiment like typical negations, contractions as negations or some conventional use of punctuation to signal which increases the sentiment intensity. Although VADER is mostly used to analyse Tweets, we think it can also be used for our online reviews for the products.

Conclusion of Improvement Attempts

After these two attempts, we can find from Table 2 that the model performance becomes better than that in the first attempt and the benchmark. Although the separate groups for the age does not work well for the improvement, using the TF-IDF method to deal with the text data and adding the sentiment analysis do a good job in increasing the score. We are quite satisfied with the improvement and precision for the features and then we will go next to the model validation.

Table 2: Model Comparison of Improvement Attempts

	Weighted Average F1 Score
Benchmark	0.869
First Attempt	0.940
Second Attempt (Optimized Model)	0.942

3.2.2 Model Validation

GridSearchCV has been used to choose the hyperparameters in logistic regression for the optimized model. GridSearchCV let us combine an estimator with a preamble to grid search to tune hyperparameters (Data Science, 2018). The method selects the appropriate grid search parameter and then uses it with the estimator selected by us (Data Science, 2018). The hyperparameters chosen to optimize are the inverse of regularization strength (C), the type of penalty (L1 or L2), whether to fit an intercept and the class weight. After GridSearchCV, the inverse of regularization strength (C) is chosen to be 526315789.4736937. The class weight is chosen to be {0: 1, 1: 1} which means "Recommended" class and "Not Recommended" class both have one vote. Not fitting an intercept and L2 penalty are also chosen as the best estimators. Other hyperparameters are default.

After fitting the best estimators, the optimized model has been used in the test set to validate its performance. We use a confusion matrix to see the performance of our model. Our purpose is to see the percentage of our prediction that is actually correct. As shown in Figure 11, 96.5% of consumers who are predicted to recommend the products actually recommend our products.

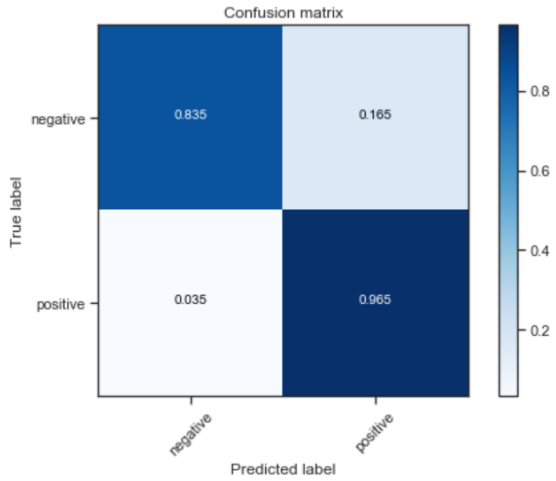

Figure 11. Confusion Matrix of Benchmark Model

As discussed in 3.1.2, we choose the weighted average F1 score as our evaluation metric since we have imbalanced classes. The optimized model's weighted average F1 score is 94.18% (Table 3). It classifies 867 observations out of 4932 to be "Not Recommended" and 4065 observations out of 4932 to be "Recommended", among which 94.18% of the observations are correctly predicted.

Table 3. F1 Score and Classification Count of Optimized Model

	F1 Score	Count
0	0.835	867
1	0.965	4065
Weighted Average	0.942	4932

3.3 Model Comparison

K-fold cross validation has been implemented on the training set to compare the performance of benchmark model and optimized model. The procedure has a single parameter called k which refers to the number of groups to be divided into for a given data sample (Brownlee, 2018). We set K=5.

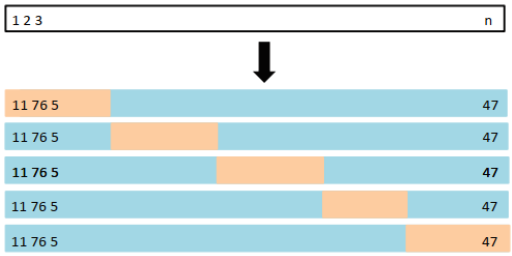

Figure 12. K-Fold Cross Validation

1. We randomly split the training sample into 5 folds of roughly equal size.

2. For each fold k ϵ {1, …, 5}, we estimate the model on all other folds combined, and use k as the validation set.

3. The cross-validation accuracy is the average accuracy across the 5 validation sets.

Table 4. Model Comparison of Benchmark Model and Optimized Model

Model	CV Accuracy
Benchmark Model	0.877
Optimized Model	0.942

As shown in Table 4, the optimized model (94.2%) has higher cross validation accuracy than the benchmark model (87.7%), which means the optimized model has a better ability to capture the pattern in the data than the benchmark model. Since the cross-validation error is calculated by (1 - accuracy), the cross-validation error for benchmark model and optimized

model is 12.3% and 5.8% respectively. This means the optimized model has less chance to misclassify whether the product is recommended or not.

Chosen the optimized model, its performance which is measured by accuracy on the test set is calculated as 94.2%. Since misclassification rate is calculated by (1 - accuracy), the misclassification rate of the benchmark model is 5.8%. This means given 100 observations, only 6 observations are misclassified by the optimized model.

4. Recommendation

According to the previous analysis and comparison, the optimized model has a better performance. The rating of products has a high correlation with whether customers will recommend the products or not. The higher the rating is, the more possible they will recommend the products. What's more, it seems that customers who are under the age group from 35 to 45 years old are more likely to buy clothes online. Compared to younger customers, older people prefer to recommend the company to their friends or colleagues. If the company wants to improve their net promoter scores, they need to recognize what their customers prefer and analyse the customer's opinion from their review text and title.

According to Figure 13, positive effects for customers' decision making are rating scores, the word "perfect" which is extracted for the review text and title from the customers, and the sentiment from the review title. However, some types of clothes like pants, shorts and jeans are not preferred by the customers and will not be recommended by them.

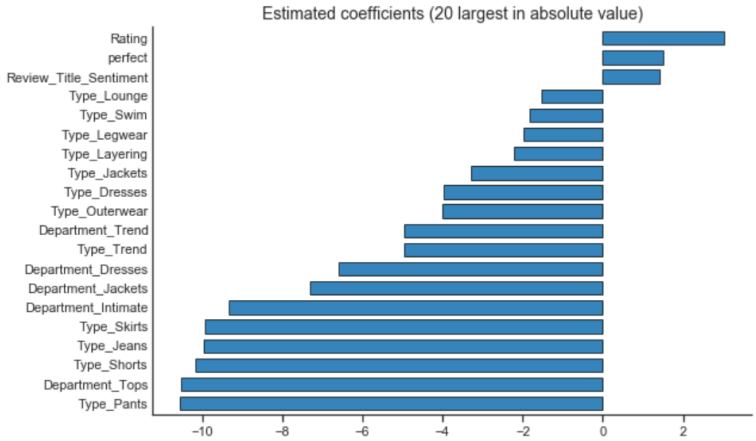

Figure 13. Important Coefficients of Optimized Model

The variable “perfect” indicates that if customers put this word in their review text or review title, they will highly recommend the products they buy to their friends or colleagues. This means when some customers see the word “perfect” in others’ review text and review title, they tend to have a positive preference for the product. The company is suggested to put review text or title with the word “perfect” at the top of the review lists, so that it can be highlighted when customers review the feedback. Therefore, more customers are expected to buy the product after viewing the positive feedbacks.

High rating score means more customers are giving positive feedback. The result shows that customers see rating score as an important component to judge the product. As the current rating score has a positive effect on the promoter score, the company is suggested to collect more high rating scores in order to show high recommendations from customers. This can be implemented by, for example, providing a discount next time for customers who are giving 5 score of rating this time. The company is also suggested to show their high rating score on the homage to attract more customers.

The sentiment analysis is the one we add for the company to have a clear view of the customers reviews and help them to improve the promoter score by accepting their opinions. Medhat et al. (2014) state that these reviews are important to the business holders as they can make decisions according to sentiment analysis results of customer’s opinion about their products. It can also be proved that the sentiment of the review title of the customers do have a positive impact on our models. The company can decide to add this sentiment analysis in their future to make better marketing decisions.

The reason for a decline in the net promoter score is because the types of clothes, especially the bottoms, are not recommended by customers. We observe that bottoms such as skirts, jeans, shorts and pants are mostly not recommended by customers. In addition, the sale quantities of bottoms like dresses are large. There are some problems in the procurement strategy for bottoms. To avoid receiving more negative feedback from customers who bought these types of clothes, the company is suggested to either stop producing bottoms or make an improvement for bottoms in terms of size, style, quality, etc.

We have observed that most customers shopping from online clothing retailers are in the middle-aged group. Old customers are relatively less thus providing less feedback. Young customers prefer giving more reviews but are critical than old customers. However, shown by

the result, the age group is not a significant component to affect customers' opinions. It is not quite related to the promoter score.

5. Conclusion

The company should show customers high rating scores as well as highlight the customers' reviews which included the word "perfect", and aim to let more customers recommend their products. To improve their offering, the company should improve their production on bottoms to avoid additional negative reviews. Moreover, the interesting factor of customers' levels of preference shown by sentiment analysis needs to be involved in the company's consideration since it is quite related to customers' decision making.

6. Reference

Brownlee, J. (2018). *A Gentle Introduction to k-fold Cross-Validation*. https://machinelearningmastery.com/k-fold-cross-validation/

Data Science. (2018). *How to use the output of GridSearch?* https://datascience.stackexchange.com/questions/21877/how-to-use-the-output-of-gridsearch

Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. *Ain Shams engineering journal*, *5*(4), 1093–1113.

Ramos, J. (2003). Using tf-idf to determine word relevance in document queries. *Proceedings of the first instructional conference on machine learning*, Vol. 242, 133–142.

Rubin, D. (1976). Comparing Regressions When Some Predictor Values Are Missing. *Technometrics*, 18(2), 201–205. https://doi.org/10.1080/00401706.1976.10489425