



SMU

SINGAPORE MANAGEMENT
UNIVERSITY

IS450

Text Mining and Language Processing
Prof Swapna Gottipati

G2T6

AY 20/21 Term 2

Unravelling Insights from Hotel Reviews with
Classification and LDA models

Group Members:

Sean CHAI Shong Hee, sean.chai.2017

LOH Yu Jin, yujin.loh.2017

LEE Yuan Kang, yklee.2017

Jaslyn WONG, jaslyn.wong.2017

FOO Yong Long, ylfoo.2017

CHYE Soon Hang, shchye.2017

26 April 2020

| | |
|--------------------------------------|-----------|
| Introduction | 3 |
| Problem Statement | 4 |
| Solution Overview | 5 |
| Solution Details | 7 |
| Document Classification | 7 |
| Data Pre-processing | 7 |
| Model Solution | 8 |
| Sentiment Analysis | 10 |
| Topic Modelling | 11 |
| Data Pre-processing | 11 |
| Model Solution | 12 |
| Results & Analyses | 13 |
| Document Classification | 13 |
| Sentiment Analysis | 14 |
| Topic Modelling | 14 |
| Dashboard Rundown | 15 |
| Discussion & Gap Analysis | 18 |
| Document Classification | 18 |
| Sentiment Analysis | 18 |
| Topic Modelling | 19 |
| Gap Analysis | 19 |
| Future Work & Conclusion | 21 |
| Project Reflection | 22 |
| References | 25 |
| Appendices | 27 |

1. Introduction

The European tourism industry will be recording its “*seventh year of expansion*” (Karantzavelou, 2019) in the current economic cycle. However, the bottom line for hotels in Europe has been facing pressures for the past two years. Further analysis shows concerns especially in “*revenue from food and beverage, conference and banqueting and on a per-available room basis*” (Fox, 2020), based on the data from “Hotel Management”, a newsletter covering investment and development coverage worldwide. The team has done further secondary research by scraping results from the query “Europe hotel 2018-2019 customer satisfaction index” from google pages. The team found a slightly overall negative sentiment of -0.11559 (scale from -1 to 1).

Search

europa hotel 2018-2019 customer satisfaction index

www.jdpower.com › business › press-releases › 2018-north-america-h...

2018 North America Hotel Guest Satisfaction Index (NAGSI ...

Jul 24, 2018 - Now, as hotels look to push customer satisfaction levels higher, their focus ... has offices serving North/South America, Asia Pacific and Europe.

www.hotelmanagement.net › guest-relations › customer-satisfaction-h...


Customer satisfaction in hotels dips | Hotel Management


Apr 30, 2019 - Anbang is selling 15 of its 18 owned hotels, and developer Charles ... to the " American Customer Satisfaction Index Travel Report 2018-2019.

www.researchgate.net › publication › 275191198_A_customer_satisfacti...

A customer satisfaction index model for international tourist ...

Four-hundred and twelve customers of international tourist hotels were surveyed. ... in multi-airport regions: Implications for tourism destination. Article. May 2019 ... European Customer Satisfaction Index Model: Comparison of Evidences from ...





```
page 1 - Sentiment: -0.11895833333333335, Subjectivity: 0.5213392857142858
page 2 - Sentiment: -0.11069336219336222, Subjectivity: 0.3480274170274171
page 3 - Sentiment: -0.11409722222222221, Subjectivity: 0.3397837301587302
page 4 - Sentiment: -0.12893939393939394, Subjectivity: 0.3390075757575758
page 5 - Sentiment: -0.07679166666666667, Subjectivity: 0.4416527777777778
page 6 - Sentiment: -0.12840079365079363, Subjectivity: 0.39611507936507934
page 7 - Sentiment: -0.15444444444444447, Subjectivity: 0.39652777777777776
page 8 - Sentiment: -0.09876388888888889, Subjectivity: 0.5043472222222223
page 9 - Sentiment: -0.03323232323232324, Subjectivity: 0.37654040404040406
page 10 - Sentiment: -0.19535732323232324, Subjectivity: 0.46113888888888893
Overall Sentiment: -0.11596787518037521, Subjectivity: 0.4124480158730159
Sentiment -> negative vs. positive (-1.0 => +1.0)
Subjectivity -> objective vs. subjective (+0.0 => +1.0)
```

The team understands that service levels and improved offerings will lead to greater customer satisfaction that positively impacts profit. Utilising a dataset containing 515,000 hotel reviews retrieved from Kaggle, as shown in Appendix A, the team aims to solve the following problem statement.

Problem Statement

Growth of reviews are becoming too overwhelming for the European Tourism Association (ETOA) which are mainly made up of hotel owners. They are unable to effectively identify key concerns portrayed by customers from the vast amount of reviews which prevents them from obtaining actionable insights.

The two main analytics tasks done are Document Classification with Sentiment Analysis and Topic Modelling. In the former task, the team will analyse the sentiment of each sentence and affirm the overall sentiment of a review that is classified. This gives the business a breakdown of the more pressing reviews to take action on. The latter will complement and build upon the first task by providing topics of concern from a sentence level of granularity. By incorporating topic modelling, it gives business users a clearer picture as to the aspects of the hotel services the review is criticising.

Supported by comprehensive analyses, the team aims to communicate insights and recommendations to the ETOA members on the satisfaction levels and experience faced by customers. The insights gained can be utilised to enhance the competitiveness of the Hotel Industry in the Europe region.

2. Solution Overview

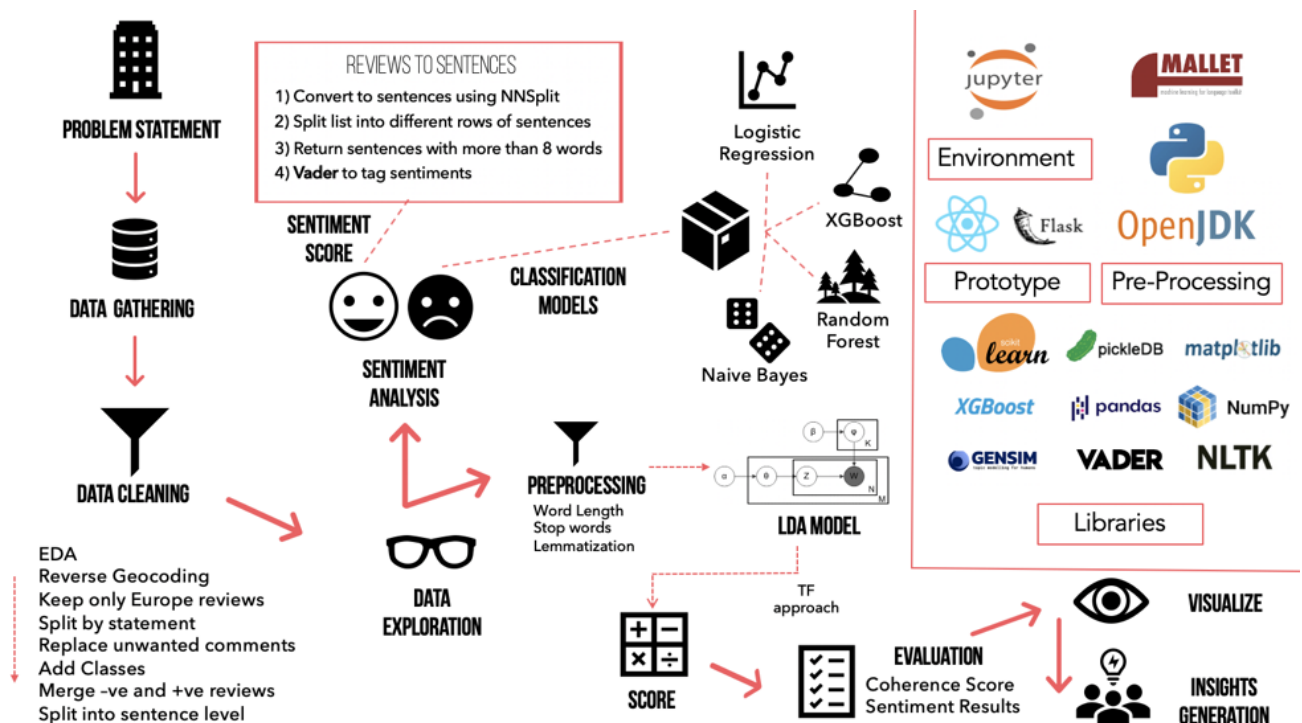


Figure 1: Summary of Process

With a large influx of reviews every month, a smooth data pipeline needs to be designed to analyse these reviews. The diagram above will give an overarching outline of the solution in tackling the mentioned problem statement. In greater detail, overall sentiment of a new review cannot be determined unless a trained classifier is used to predict the result. Since a review can have a mixture of positive and negative sentences, classification will only return the overall sentiment of the review but it does not capture the sentiments of individual sentences in the review. The purpose of sentiment analysis is to understand the sentiment based on each sentence rather than sentiment at a review level. Simultaneously, the Latent Dirichlet Allocation (LDA) topic tagging to each sentence was simply icing on the cake.

Pre-processing is crucial in ensuring high predicting power given a review. It also requires domain knowledge to fully understand our dataset and to decide on the steps to best train the classifier. On top of it, the team has to go beyond what was taught in class to explore the different classification models, the maths behind it, and tune their hyperparameters to best fit our data. Even after having the line of best fit, one has to ensure that the model is neither underfitting nor overfitting. Therefore, K-fold Cross Validation is done to ensure the fit of the model.

Implementing sentiment analysis was made difficult when there are no common delimiters like full stops and capital letters that can be identified to break reviews up into sentences. A temporary fix will be to import an external library - NNSplit (Benjamin, n.d.) to split reviews with a F-score of 0.963. The solution uses Long Short Term Memory (LSTM) technique to train its model to predict the splits. Another challenge is posed by the need to pre-process the data before dumping into the Vader model. Usage of stopwords and text-transformation using stemming or lemmatization were both put into consideration here. While data

massaging is supposed to remove noise and ensure consistency for the model to train on, it is found out that removing the stopwords and transforming the text made it more difficult for the model to identify the sentiment. As such, the verdict is to go ahead with the normal form. Lastly, some of the weaker polarity sentences were wrongly tagged. For instance, a weak positive sentence gave a negative sentiment, vice versa. This would greatly impact the outcome of the sentiment, resulting in business users to not have accurate sentiment of their hotels. Manual adjustments to the cutoffs between negative, neutral and positive needs to be calibrated.

As the topic modelling acts like an add-on to the sentiment analysis, bringing more values, it too requires NNSplit. The dilemma comes when the team has to decide on the N-gram used to train the LDA model. Though it was expected that useful bigrams like “air conditioning” and “high ceiling” are prominent, much to the team’s surprise, these words did not appear often in the top 10 keywords extracted from each of the topics. In contrast, these bigrams will only detriment the coherence score, the evaluation method used to measure the coherence of each topic to be understood by humans. As such, using unigram serves a better benchmark in identifying certain topics in the sentences.

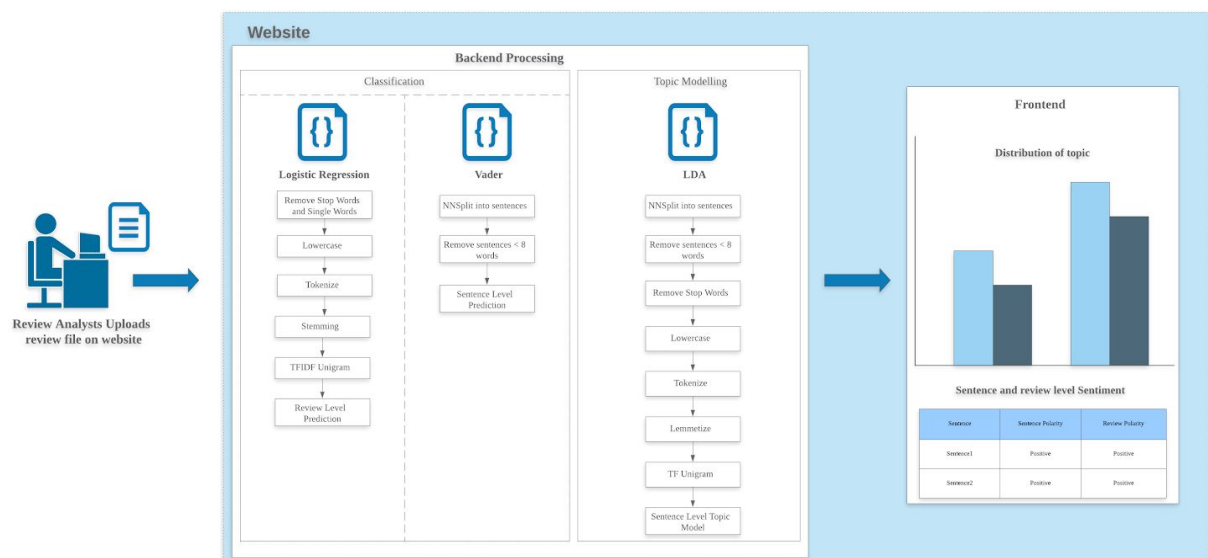


Figure 2: Overview of solution

In greater detail, the solution is built to ease business users in analysing new reviews. The above overview diagram will depict the different pre-processing methods used in the different algorithms.

3. Solution Details

Document Classification

Data Pre-processing

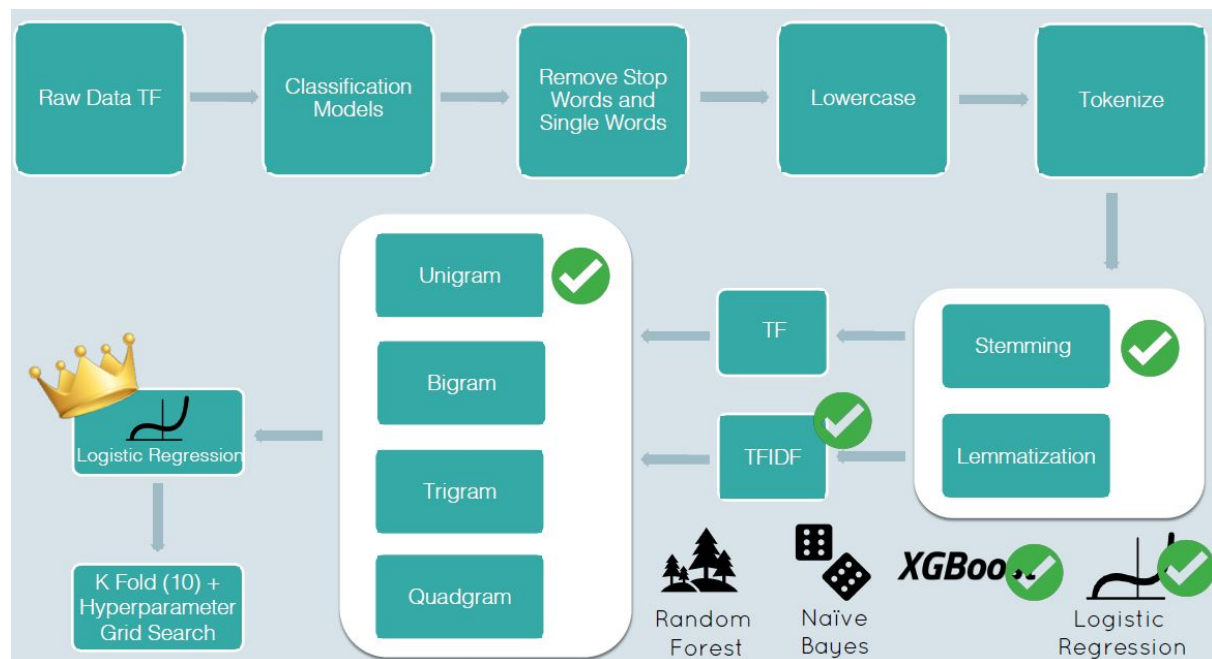


Figure 3: Overall Steps to Classification

Before carrying out classification on the preprocessed hotel reviews, the team ran 4 different classification models on the raw dataset. This was done to first determine the top 2 classification models to be used for the preprocessed dataset. Doing so eliminates the redundancy of training all 4 classification models multiple times at later stages of the classification task. The team trained the 4 following classification models on the raw data - Naive Bayes, Logistic Regression, XGboost and Random Forest. Logistic Regression and XGBoost yielded the best initial accuracy scores of 0.939 and 0.928 respectively. Logistic Regression had a F-score of 0.94, while XGBoost had a F-score of 0.93.

The team proceeded to carry out pre-processing by first removing stop words and single words. Single words consisted of words such as “NIL, Positive, Negative, Nothing, Blank” tagged with Positive or Negative labels. These words were not meaningful and hence did not add value to the classification task. The dataset is made case insensitive by converting all letters to lowercase and tokenization was carried out after.

The team created a stemmed dataset as well as a lemmatized dataset to run Logistic regression and XGBoost models on them. Doing so would allow the team to compare results obtained from stemmed and lemmatized datasets - the one with the best accuracy and F-score would be used for the final model.

The stemmed and lemmetized datasets were subsequently converted into both Term Frequency (TF) and inverse document frequency (TFIDF) models. XGBoost and Logistic

Regression were trained on both stemmed and lemmatized TF and TFIDF models, where a unigram, bigram, trigram and quadgram model were used for each of the TF and TFIDF models. Ultimately, Logistic Regression, trained using a stemmed TFIDF unigram model, yielded the best accuracy and F-score. With this model, the team carried out Hyper Parameter Tuning through Grid Search with a 10-Fold Cross Validation technique.

Model Solution

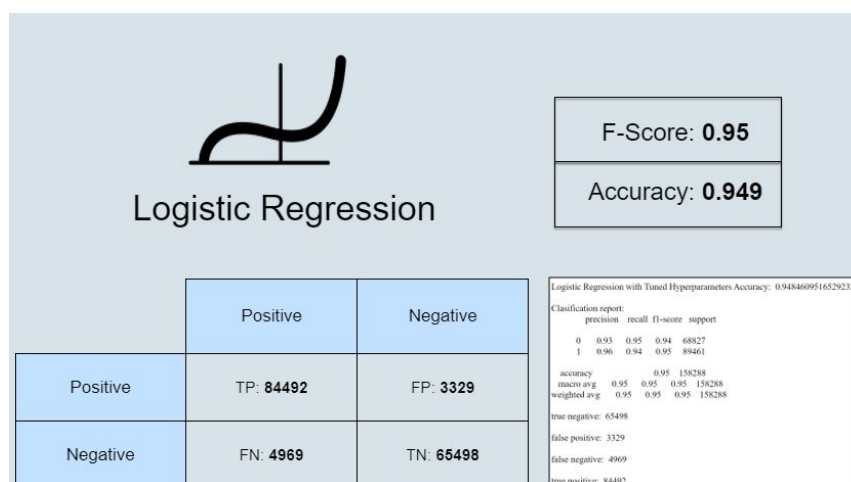


Figure 4: Logistic Regression Result

Logistic Regression was found out to be the most suitable classification model for the dataset consisting of Hotel Reviews. The teams' best result was obtained by training Logistic Regression using a stemmed TFIDF unigram model. To further tune and validate the results, a simple hyperparameter tuning with Grid Search with a 10-Fold Cross Validation was carried out.

Hyperparameter Tuning with Grid Search

The classification models were trained with default parameters determined by scikit-learn's modules. The following picture shows the hyperparameters tuned using Grid Search.

```
grid={"C":np.logspace(-3,3,7),
      "penalty":["l1","l2"]}
```

Figure 5: Hyperparameters for Logistic Regression

'penalty' specifies the norm used in penalization. "L1" and "L2" are regularization techniques used to prevent overfitting and stabilizing estimates. L1, also known as "lasso", can shrink some coefficients to zero to perform variable selection. On the other hand, L2, known as "ridge", shrinks all coefficients by the same proportions and does not perform variable selection where all variables are considered.

"C" specifies the inverse regularization strength. Small values specify stronger regularization.

```
tuned hyperparameters :(best parameters) {'C': 1.0, 'penalty': 'l2'}
```

Figure 6: Best hyperparameters for Logistic Regression

The best hyperparameter obtained for the Logistic Regression model had an inverse regularization strength of 1.0 and the penalty used was "ridge".

10-Fold Cross Validation

Cross validation is a resampling procedure used to get a better estimate of the accuracy and F-score of the Logistic Regression model. 10-Folds is used to ensure that estimates from the Logistic Regression model were unbiased. Ultimately, this reduces the likelihood of overfitting or underfitting.

```
logreg=linear_model.LogisticRegression()  
logreg_cv=GridSearchCV(logreg,grid,cv=10)  
logreg_cv.fit(xtrain_tfidf_unigram_stem,train_stem_y)  
logreg_pred = logreg_cv.predict(xvalid_tfidf_unigram_stem)
```

Figure 7: Code for 10-Fold Cross Validation with GridSearch

As can be seen from Figure 7, cv=10 specifies the number of folds for cross validation to be carried out.

```
Logistic Regression with Tuned Hyperparameters Accuracy: 0.9484609516529232  
  
Clasification report:  
              precision    recall  f1-score   support  
  
         0         0.93      0.95      0.94      68827  
         1         0.96      0.94      0.95      89461  
  
    accuracy              0.95      158288  
  macro avg              0.95      158288  
weighted avg              0.95      158288  
  
true negative: 65498  
false positive: 3329  
false negative: 4969  
true positive: 84492  
  
tuned hpyerparameters :(best parameters) {'C': 1.0, 'penalty': 'l2'}
```

Figure 8: Results from 10-Fold Cross Validation with GridSearch

The best accuracy score of 0.949 and F-score of 0.95 was obtained through Logistic Regression after tuning parameters and validating them with a 10-Fold Cross Validation.

Sentiment Analysis

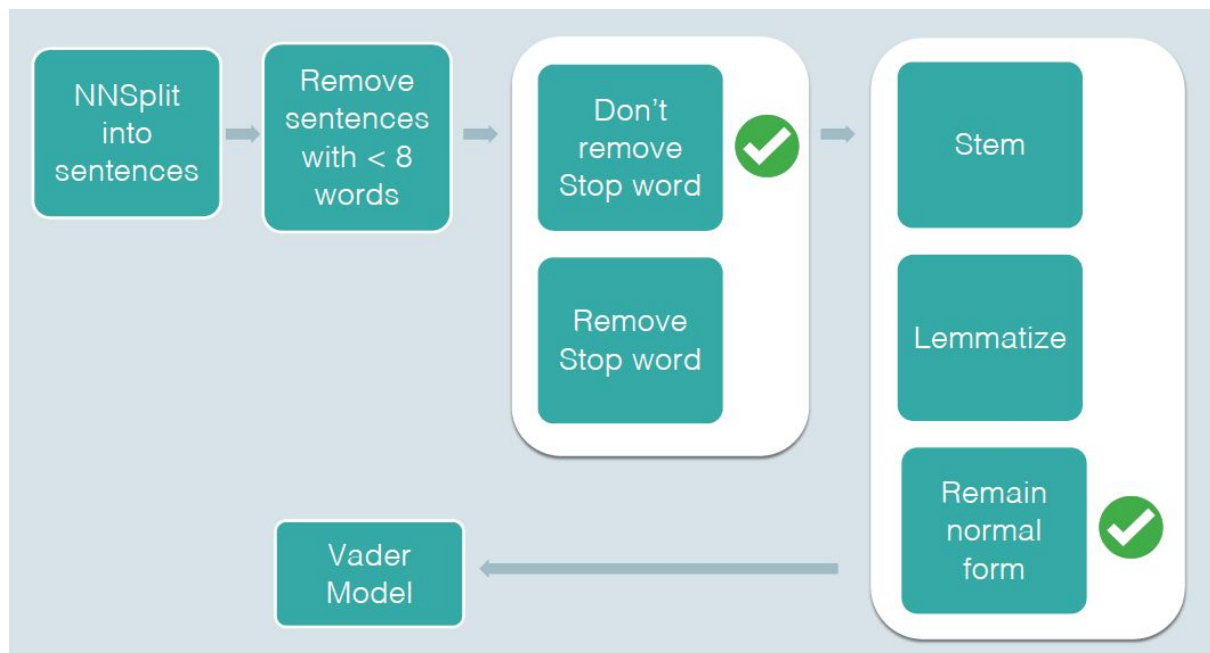


Figure 9: Overall Steps to Sentiment Analysis

Before tagging the individual sentence with a sentiment, a split of sentences from the reviews has to be done, and that can be achieved by using the library NNSplit as mentioned above. This will allow sentence level pre-processing and analysis. Following that, removing the sentences that have less than 8 words will retain all the more meaningful sentences. Sentences with length less than 8 are usually short phrases that may not make sense especially when we are relying on the NNSplit library to do the splitting.

To decide whether to remove stopwords or don't remove stopwords, both ways have been tested and the results proved to be better without removing the stopword. Following which will be Stemming, Lemmatizing or Retaining it in its normal form. Upon trying all the combinations, the best result achieved was to retain normal form which is not to stem and lemmatize.

To summarize the entire pre-processing steps, the first step is to split the review into sentences and remove sentences with less than 8 words. After this, the sentence data will be fitted into the Vader model and Vader will tag a sentiment, which is positive or negative to the sentences. It also provides a polarity to the sentiment so that it can show how positive or how negative the sentence is.

After all, the objective is to find out the sentiment of the reviews. Therefore, aggregating each of the sentences into a review will give the review level sentiment. Using the polarity of each sentence, the aggregation will be done and if the over the polarity of the review is more than 0.05, it will be positive, in between 0.05 to -0.05, it will be neutral and lastly less than -0.05, it will be negative.

As Vader is an open-source pre-trained model specialised for twitter dataset, the model may not be suitable in all use cases. In this project, the nature of review data may differ from the

twitter dataset used to train the model, hence the polarity returned may differ or be inaccurate. One approach taken to overcome this is to manually adjust the threshold by observing the output. This, however, is not perfect because ultimately, the problem still lies in the nature of the trained dataset

In essence, the purpose of sentiment analysis in this project is to understand the sentiment on sentence-level so that it can give a better classification to the reviews as some sentences may have an opposite sentiment from the review.

Topic Modelling

Data Pre-processing

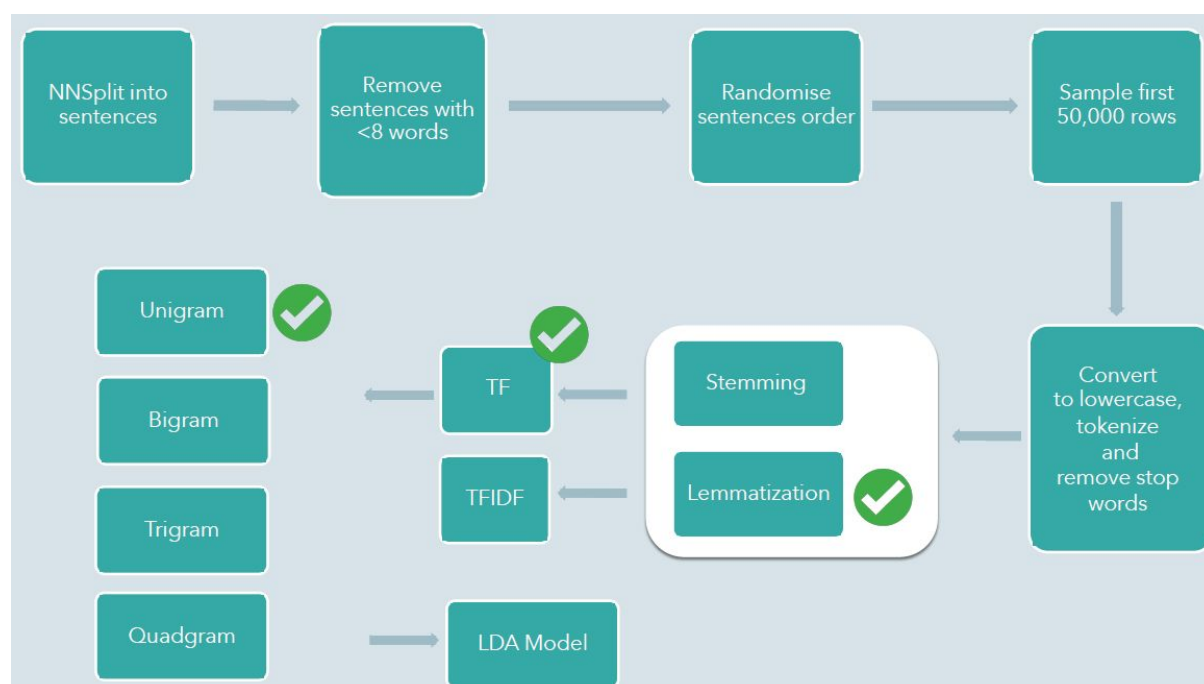


Figure 10: Overall Steps to LDA Model Pre-processing

As mentioned before, topic modelling helps give business users an idea of the hotel aspect the customer is concerned about in a glance without individually going through the sentences. The pre-processing comes with different steps starting with NNSplit, which will break reviews into sentences just like how it is done when doing sentiment analysis. From 515,000 rows of reviews, which eventually expanded to more than 1.9 million rows from the NNSplit, it will take forever to train the LDA model based on the entire dataset. With that, it is logical to take 50,000 sample sizes to train the model. Further pre-processing steps such as lowercasing, removal of stopwords using NLTK's English stopwords list and removal of non-alphabets using regular expression are done to standardise the words. With domain knowledge relevant to the hotel industry which is also not captured in the NLTK stopwords list, more stopwords like "hotel" are appended to the NLTK list to ensure a cleaner dataset.

Separately, stemming using Gensim and lemmatization using WordNetLemmatizer is used to ensure the root form of the words. This will allow TF as well as TFIDF to better identify keywords for the frequency distribution. As every word is equally important in our topic modeling, we do not want to penalise common words. On top of this, the number of words in

our sentences does not deviate too much from each other. This further discourages the need for TFIDF weighting to be applied to the words. Therefore, it is more compelling to take TF over TFIDF.

Moving forward, having N-gram variations may seem logical to identify hidden features that can be valuable for our analysis. As such, unigram is used to cross-check against bigram, trigram and quadgram to check for its utility. Ultimately, unigram is still retained due to its coherence score attained. Another main reason is that the number of bigrams appearing is all-time low, let alone for trigram and quadgram.

After the data pre-processing, the data is finally funneled into the LDA model to deduce the topics for these reviews.

Model Solution

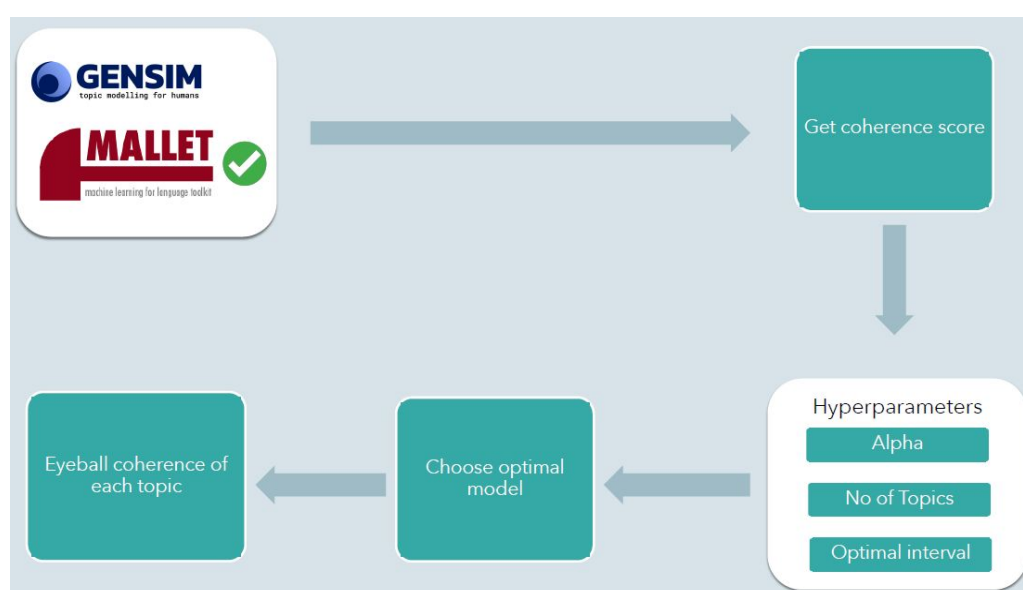


Figure 11: LDA Model Selection

The team then fit the data into the LDA model from Gensim to be compared to the one in Mallet. The main evaluation matrix to determine the optimal model is the coherence score as it will score the quality of the topics generated. For example, the topic food with words such as “breakfast”, “food”, “restaurant”, “service”, “choice” is more coherent compared to a topic price with words including “day”, “time”, “paid”, “card”, “extra”, “charge”.

By tuning the hyperparameters, mainly the Alpha, number of topics and optimal interval, the team trained the model differently to yield the optimal coherence score. It is found that the model using LDA from Mallet with the Alpha of 5, number of topics of 15 and optimal interval of 25 return us the best coherence score of 0.5050783 (the higher the more coherence).

4. Results & Analyses

Document Classification

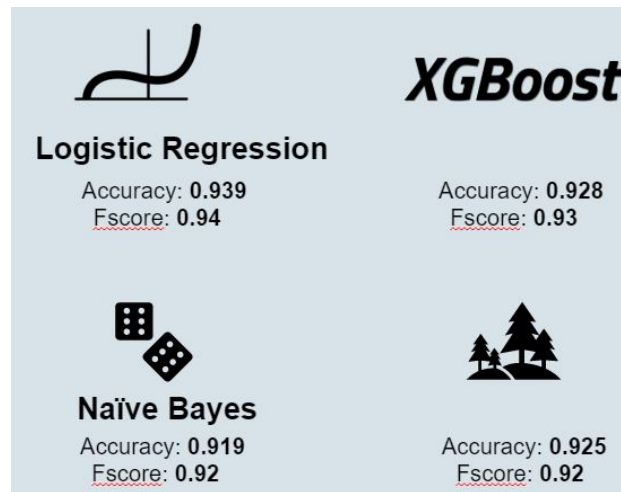


Figure 12: Classification Results

Training Logistic Regression on the raw data yielded an accuracy score of 0.939 and a F-score of 0.94. XGBoost was the runner-up, yielding an accuracy score of 0.928 and F-score of 0.93. Meanwhile, Random Forest and Naive Bayes had F-score of 0.92, and accuracy scores of 0.925 and 0.919 respectively.

Logistic Regression with Tuned Hyperparameters Accuracy: 0.9484609516529232

| | | | | |
|-----------------------|-----------|--------|----------|---------|
| Clasification report: | | | | |
| | precision | recall | f1-score | support |
| 0 | 0.93 | 0.95 | 0.94 | 68827 |
| 1 | 0.96 | 0.94 | 0.95 | 89461 |
| accuracy | | | 0.95 | 158288 |
| macro avg | 0.95 | 0.95 | 0.95 | 158288 |
| weighted avg | 0.95 | 0.95 | 0.95 | 158288 |

Figure 13: Final results of Logistic Regression for Classification Task

The final classification model was obtained by training Logistic Regression through a stemmed TF-IDF unigram model. The team conducted a simple tuning of hyperparameters and a 10-Fold Cross Validation to achieve a final accuracy score of 0.949 and the F-score was 0.95.

Sentiment Analysis

| Sentences | Polarity |
|---|----------------------|
| the scenery great with seaview right front eye | 0.6249 |
| however the window not clean for enjoy the view | -0.5975 |
| food good and tasty fish fresh with nice prese... | 0.7906 |
| the service quality good and efficient and the... | 0.6908 |
| Final Aggregated Sentiment | ('positive', 0.3772) |

Figure 14: Vader's Sentiment Analysis Results

As the Vader model is a pretrained model using twitter data, the result performed pretty logically as seen in Figure 14. Every sentence is tagged with a polarity that is representative of the actual meaning and from each polarity, the team combines them and produces a more aggregated sentiment result based on the aggregated polarity. However, since it is a pretrained model, there is a limitation of slangs and sarcasm which will not perform as well and those are now within the team's control.

Topic Modelling

After generating and tuning the model to find the best number of topics to use, the model with the higher coherence score of 0.5050783 was chosen. The score has an optimal number of 15 topics and the distribution of words in each topic are shown in Appendix B. As observed in Appendix B, some topics are classified quite accurately such as topic 1, 2, 3 and 7 being room comfort, food, reception service and accessibility. However, for some topics, there was more than one feature that was put into the same topic. An example will be topic 13 which talks about the temperature, light and the wifi of the room. In addition, topic 0 were having keywords such as 'stay', 'star', 'place', 'visit', 'thing', 'time', that do not link well with each other.

With a coherence score of 0.5050783, this means that the topics are not very distinct from each other and thus there were some overlaps with the words in the different topics like 'stay', 'breakfast', 'room'.

To give better insights for the business users, each topic was tagged to a topic name instead of just giving the topic number. This is so that when business users are looking at the interface, they are able to make sense of the kind of topics that each sentence review is giving. The names of the topics are as follows:

- 0: 'general',
- 1: 'room comfort',
- 2: 'food'
- 3: 'reception service'
- 4: 'room necessities'

- 5: 'price'
- 6: 'overall stay experience'
- 7: 'accessibility'
- 8: 'room noises'
- 9: 'staff'
- 10: 'room size upgradable'
- 11: 'location amenities'
- 12: 'hotel amenities'
- 13: 'room lighting and temperature'
- 14: 'booking'

Dashboard Rundown

Assuming that the business users have no technical background, it will be challenging for them to apply the models developed earlier to their review analysis workflow. To satisfy the business requirements for hotel owners, the team has developed a one-stop solution, in the form of a dashboard. The following section will elaborate on a proposed workflow to integrate the dashboard to the analytics segment of the hotel business.

To analyse fresh customer reviews, hotel staff will first collate the reviews into a CSV file. This CSV file will then be uploaded into the dashboard with a click of a button, seen in Figure 15.

Sentiment Analysis for Customer Reviews

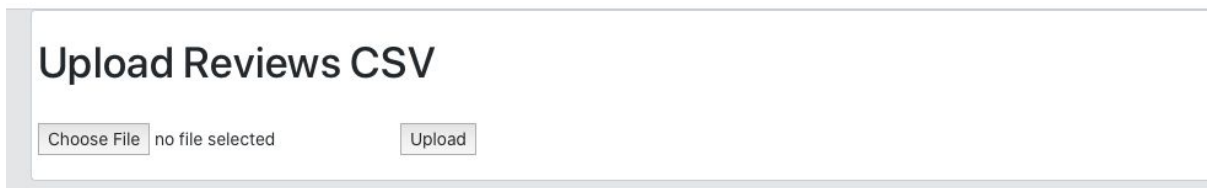


Figure 15: Upload Section of the Dashboard

With the uploaded CSV filled with customer reviews, the dashboard will run the classification process on each review. This process is supported by the Logistics Regression Classification model trained based on previously labeled customers. At this stage, the hotel staff can get rough sensing on customer sentiments towards their hotel based on the review level classification, for example shown in Figure 16.

Classified Reviews

| REVIEW_ID | REVIEW | CLASSIFICATION |
|-----------|---|----------------|
| 1 | This hotel is being renovated with great care and with an appreciation for its unique structure and location My spacious and comfortable room had a large double paned glass window onto the lush greenery of the park The breakfast selection was spectacular All considered this was a great hotel for the price and I plan to return | Positive |
| 2 | It was very good very historic building that s why I chose it | Positive |
| 3 | This hotel is awesome I took it sincerely because a bit cheaper but the structure seem in an hold church close to one awesome park Arrive in the city are like 10 minutes by tram and is super easy The hotel inside is awesome and really cool and the room is incredible nice with two floor and up one super big comfortable room I ll come back for sure there The staff very gentle one Spanish man really really good | Positive |
| 4 | Great onsite cafe Amazing building Park location Amazing Bobby Gin and Tonic | Positive |
| 5 | We loved the location of this hotel The fact that it is set in a Park away from the busy centre of dam square was great The tram system was brilliant and easy to handle The hotel is lovely and the bed was comfy Staff were very friendly and helpful and familiarized themselves with us when they realized we travelled from Ireland | Positive |
| 6 | Public areas are lovely and the room was nice but the window was broken and the drains in the bathroom smelt Its an old building and clearly has old building issues | Negative |
| 7 | I liked the hotels history And for such an enormous hotel I can t imagine how many girls were orphaned I liked their breakfast It was a quirky mix of fruits pastries meats and fish breads and the regular bacon and eggs Juices tea and coffee | Positive |

Figure 16: Classified Reviews Section of the Dashboard

However, review level sentiment might not be enough to fully understand the customer's true feelings towards the hotel. Many reviews will contain more than one sentence and each sentence can have different topics and sentiments. As such, it will be sensible to drill another level down to unravel more actionable insights. For that, the dashboard handles it with a two-pronged approach.

Firstly, the dashboard will first break each review into sentences. With the sentences, the dashboard will perform topic extraction with the trained LDA model. These topics will be collated and displayed to the hotel staff to have an understanding of the areas of concern among the customers.

Secondly, the dashboard will reveal the sentiment of each sentence by applying the Vader model. At this point, the dataset has a detailed sentiment and topic distribution. This information is then presented to hotel staff in the form of a stacked bar chart, as shown in Figure 17.

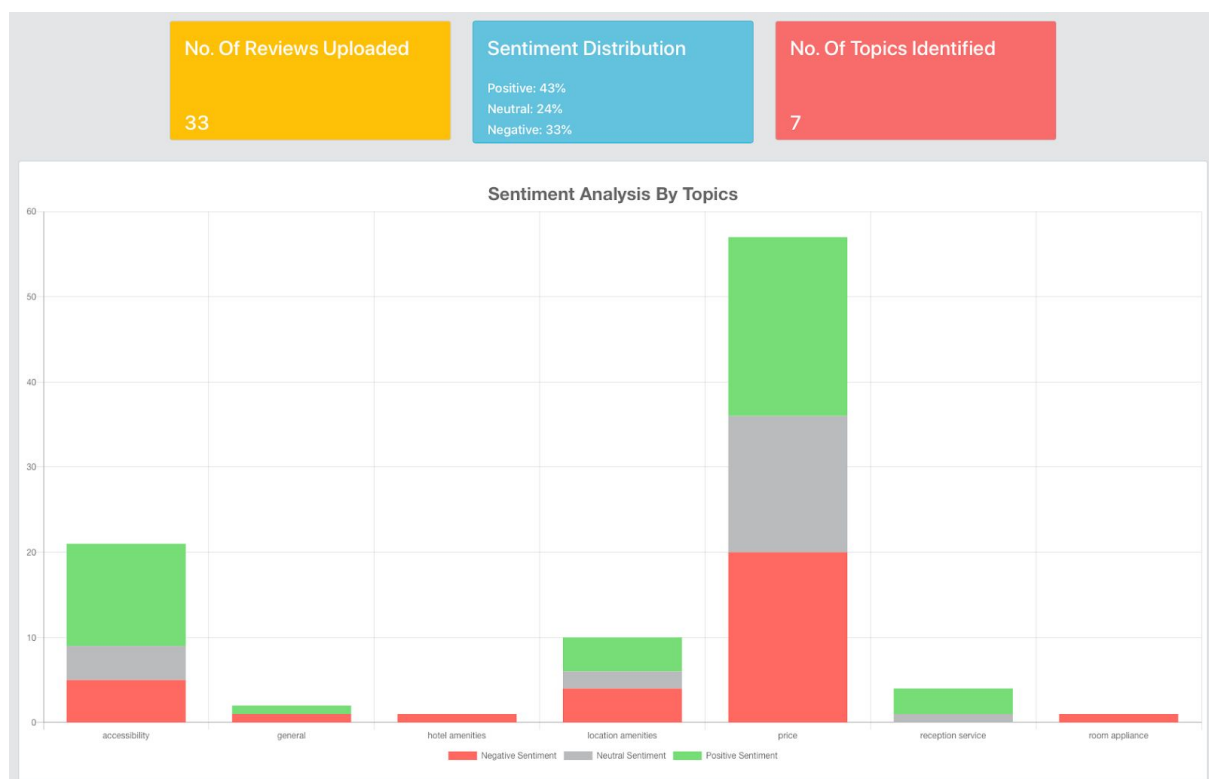


Figure 17: Stacked Bar Chart to illustrate Topic and Sentiment Distribution

For a more detailed analysis, the dashboard will render a data table to display individual sentences with the polarity score, sentiment and topic identified, as shown in Figure 18. This table will allow the hotel staff to read through the reviews in a more detailed manner to pick up more information.

Detailed View of Sentiment Analysis (Sentence-Level)

| REVIEW_ID | SENTENCE | TOPIC | POLARITY | SENTIMENT |
|-----------|--|--------------------|----------|-----------|
| 1 | this hotel is being renovated with great care and with an appreciation for its unique structure and location | price | 0.891 | Positive |
| 1 | my spacious and comfortable room had a large double paned glass window onto the lush greenery of the park | reception service | 0.5106 | Positive |
| 1 | all considered this was a great hotel for the price and i plan to return | price | 0.6249 | Positive |
| 2 | it was very good very historic building that s why i chose it | price | 0.4927 | Positive |
| 3 | i took it sincerely because a bit cheaper but the structure seem in an hold church close to one awesome park | reception service | 0.7684 | Positive |
| 3 | arrive in the city are like 10 minutes by tram and is super easy | price | 0.8519 | Positive |
| 3 | the hotel inside is awesome and really cool and the room is incredible | price | 0.7713 | Positive |
| 3 | nice with two floor and up one super big comfortable room | location amenities | 0.875 | Positive |

Figure 18: Data table to display more details on each sentence

With such an easily integrated dashboard, this will help hotels sieve out actionable insights more effectively and enable them to have a faster response time to maintain and grow the support among their customers.

5. Discussion & Gap Analysis

Document Classification

Unable to Recognise Different Polarities within Review

Classification returns the overall polarity of reviews. However, there may be different sentiments for each sentence within a review. Classification is not able to capture sentiments of individual sentences within a review. For example, a review may contain both positive and negative sentiments within it, but the classification model only returns a positive or negative overall sentiment based on the review and may neglect some strong opposite sentiment. Identifying overall sentiments of reviews may not be useful because sentences within the review might be talking about different topics. Since hotel owners want to identify key areas of interest, it is more useful to find sentiments of each sentence in a review, where these sentiments can be tagged to a topic through topic modelling. In this case, using the Vader model can help to analyse each sentence polarity which may be a better solution.

Sentiment Analysis

Inaccurate Polarity

As mentioned in the solution details, one of the crucial gaps is the inaccuracy of the polarity, which can be overcome by manually adjusting the threshold. However, this solution may not be perfect as the underlying problem is still with the type of dataset used to train the model. The actual cutoff between the polarity may fluctuate when new data is fitted to the model, deeming the arbitrary adjustment inaccurate over time.

Unable to Capture Slang and Sarcasm

As review data do have times that users might input slang or sarcasm, the vader model will be unable to capture those. There may be some slang or sarcasm in the twitter dataset that is used to train the vader model, however, slang and sarcasm may vary from different countries and nationality which result in the model unable to capture each of it. Hence, the vader model may be inaccurate when there is slang or sarcasm within the review.

Suggestions for Improvement

Since the identified problem is the nature of the dataset. One suggestion to improve the result will be training a model using the review's sentence dataset. This will produce a more fitted model for the review use case and will likely produce a better outcome.

Topic Modelling

Choice of Evaluation Matrices

As our tasks revolve around bringing values for the business owners, by having new reviews analysed immediately, the coherence score may not be the best option as it only looks at the current basket of words in the topic without accounting for new unseen data. Using perplexity as the other evaluation matrix may be more relevant as it looks at the coverage of the topics; a topic with better coverage will better determine the unseen data be allocated into the right topic. Since it is harder to tune the model for better perplexity, both perplexity and coherence scores can be captured and evaluate the goodness of the model.

Gap Analysis

Data Limitations

The lack of punctuation in the reviews is detrimental to our analyses as the sentences broken up may not make complete sense in the human eye. The incoherent sentences will only cause more confusion for the models as the data is not completely generated by a human, making less linguistic sense. Ideally, the dataset should be a series of sentences in a review as it is understood that a customer making a review has multiple concerns.

Hardware Limitations

The team started with a dataset that has 1.9 million records after NNSplit. However, due to the running and computing time to process the large dataset, the team decided to reduce the size of the dataset to 50,000. In this case, the hardware limits the team to utilise all the 1.9 million records of the dataset which may potentially help to improve the performance of all the models and analysis done in this project.

Another instance will be the usage of NNSplit which is computational heavy as it took the team approximately 12 hours to extract the 1.9 million sentences out of the 515,000 reviews. This may not be an ideal solution if not for the lack of other alternatives to split up reviews. As such, the team has to make decisive decisions to not commit to computational heavy algorithms.

Performance between Logistic Regression and Naives Bayes Classifier

Logistic Regression, as compared to Naive Bayes Classifier, was expected to have a better result because it does not assume independence between predictors (in this case, the words in our dataset). Correlations between predictors are taken into account for Logistic Regression, hence better prediction scores. Nevertheless, higher prediction scores for Logistic Regression might be a result of overfitting, something that we overcame using a 10-Fold Cross Validation.

Performance between Logistic Regression and XGBoost Classifier

XGBoost Classifier and Logistic Regression had very close accuracy and F-scores. It was expected that XGBoost Classifier would yield better prediction scores because it is a decision-tree based ensemble algorithm that uses a gradient boosting framework. It was

expected that XGBoost Classifier would yield better scores as compared to Random Forest, which was observed from our results. Logistic Regression's higher scores might be due to better probability calibration as compared to XGBoost. Logistic Regression yielding a better score might also be due to the lack of resampling for our models. The team did not carry out Cross Validation during the comparison of model results.

6. Future Work & Conclusion

The team can explore different topic modelling techniques like the Hierarchical Dirichlet Process (HDP) model which is modeled using the Stick Breaking Construction. This can be analogous with the Chinese Restaurant process (Gensim, n.d.). The main difference between the HDP and LDA is that the HDP infers the number of topics from the data while LDA predefines the number of topics. This model may improve the result with the data used in this project as it infer the topics differently from LDA.

Vader is used for sentiment analysis, but overall sentiment of each sentence in our dataset might not be very accurate because Vader is trained on datasets obtained from Twitter. Sentiment scores for words in Vader are not trained using domain specific datasets (such as our Hotel Reviews dataset). Instead of using Vader, the team can look to use classification models to classify sentiments of reviews at a sentence level. This can be achieved by manually tagging sentiments of every sentence in our training dataset and subsequently training a classifier model on the tagged dataset.

To further improve scores of our 3 different tasks, spelling checks can be done on our dataset. Spelling checkers correct misspelled words. In this way, misspelled words will not become a separate term when our dataset is converted into a bag of words model. The team can also consider training our model to split reviews into sentence level since NNSplit might not correctly split our current dataset into sentences.

Lastly, the team can consider conducting topic summarisation to summarise key findings of each topic based on locations of hotels, since our team has already successfully geotagged locations of each hotel in our dataset.

7. Project Reflection

Sean CHAI Shong Hee

This project has allowed me to put results obtained from text mining tasks into contextual use. As opposed to merely exploring how models can be used to generate results, I have learnt how to put these generated results into contextual use. By creating a end-to-end solution, this project has helped me understand the basics of what goes into the making of machine learning systems used in the business world. I'm glad that the text mining project required us to focus on the business user, as this helped me understand how text mining can benefit the end user. Overall, text mining has been a very challenging module (it was especially difficult to grasp concepts for LDA), but it has been well worth the effort and time spent to understand the various concepts because these concepts will be very beneficial for us in the future.

LOH Yu Jin

The main takeaway from this project is to understand the necessity of having a data pipeline to digest the text related data. This is essential as different analytical techniques are needed to fulfil certain tasks. For instance, sentiment analysis will complement classification for a more insightful analysis; classification alone will not bring about this value. This can be further value-added by tagging a topic inferred by the LDA model to the data. Likewise, a document summarisation can be done to give more insights to the data. As such, different kinds of techniques can be implemented to the same data while providing values to the targeted audience. With the long list of tasks in the pipeline, individuals in our self-managed team come forth to take up each part automatically. Once done, he or she will move on to the next step without waiting, and this is what is great about this team. In essence, a high level of teamwork and trust can be seen from the team. Lastly, the project is an excellent platform to increase exposure to text mining analyses. It provides an opportunity to apply theoretical understanding to a live set of data.

LEE Yuan Kang

Text mining has always been the most important knowledge to know and acquire. During internships or other projects from different modules, there will always be text handling tasks that require pre-processing or performing some analysis to understand the text. This project allows me to put the theory I have learnt in class and online to practical use. Many times, I thought the most important component for text mining will be the analysis or insights we gained such as the general idea of one list of reviews which is what our project has done. However, after this project, I learnt that gaining the general idea of the review is just the output and its about 20% of all the work done, what matters is the pre-processing step where we have to ensure that the text are processed in a way that the machine can best understand the context of the text. With the right preprocessed steps taken, the output, analysis or insights generated will then be more accurate or make more sense for business to make decisions.

Jaslyn WONG

Text mining is a must know especially for business analysts as there is a lot of data out there that uses texts. Through this module and project, I have gained a better understanding of how text mining is being used such as to understand a context of the text as some texts consist of noises and sarcasms. Like in our project, we can provide meaningful insights to the business users, for them to understand the sentiments and topics of the reviews that people are giving. Through this course, I have also learnt that a clear use case is essential to be able to do real-world text mining work as you do not want to build something that does not provide meaningful insights for the business users.

In every text mining work, there are also undeniably some challenges like in our project, where we faced the problem of no punctuations in our hotel reviews when we had to split it into sentence level. This is something that increases the time for pre-processing the data that is given as we have to find out a different way to split the reviews into sentence level. In our learning process, we also took more time in understanding the different models especially for LDA so that we can better understand the work behind the models. Lastly, our team is a team that is efficient and never fails to provide each other guidance along the way when either of us faces any problem. Overall, it was a great learning experience of the course and this will be beneficial for my future work.

FOO Yong Long

Truthfully speaking, the content covered in text mining is overwhelming for someone without any prior experience. I struggled to grasp the concepts during classes, especially the LDA models. Text mining is a whole new domain as it deals with non-numeric data. Unlike numbers which simply convey facts, text data comes with rich information within it such as emotions and areas of concerns. However, the pre-processing stage to uncover such insights is way tougher due to the complexity nature of the dataset such as sarcasm, inconsistent structure and noises.

However, by working/catching up on labs after classes and project experience, I was able to understand and practice the theories in class and implement them to solve a business issue. Other than the underlying statistics and logic that governs the model, I've learned how to pre-process the data and fit it into the training models to better get the desired output I need to solve the business issue. I foresee using this in my future internships and combining it with other visualization tools to produce dashboards for my clients/managers.

The whole overall learning experience was a smooth one despite the heavy content thanks to project work. By splitting into mini-teams and working on different aspects of the project, we can be subject-matter experts on different aspects of the project and channel our efforts into mastering different aspects. By sharing and guiding one another on our tasks, we learned faster overall as a group. My teammates received the most credit for this with their selfless attitude and thus, I am sure that we have benefited greatly from this project and Text Mining.

CHYE Soon Hang

This project has been a great opportunity for us to apply the theory we learnt from the weekly lectures. Before the project, it was challenging for me to grasp the concepts learnt in class. But the project allowed us to relate the techniques to real-life text-mining problems like dealing with cumbersome hotel reviews. As someone who learnt more from practical experiences, this project has been very helpful. Apart from applying the concepts learnt in class, the project has challenged us to come out with a suitable business solution to allow our business users to tap on the wonders of text-mining techniques with minimal technology background. And this has provided a great learning opportunity for us to learn beyond text mining techniques to create a viable user-interface for our business users. Lastly, our team has displayed great synergy and this has been one of the most fun and rewarding project experiences I have.

8. References

- Akici, F. (2017, March 1). A Non-Technical Introduction to LDA Topic Models with an Application in Hotel Reviews. Retrieved from <https://www.linkedin.com/pulse/non-technical-introduction-lda-topic-models-hotel-reviews-fatih-akici/>
- Benjamin, M. (n.d.). NNSplit. Retrieved from <https://pypi.org/project/nnsplit/>
- Enes P. (2018, April 3). Grid Search with Logistic Regression. Retrieved from <https://www.kaggle.com/enespolat/grid-search-with-logistic-regression>
- Fox, J. T. (2019, May 29). Revenue levels drop at Europe's hotels. Retrieved from <https://www.hotelmanagement.net/own/revenue-levels-fall-at-europe-s-hotels>
- Ganegedara, T. (2019, March 27). Intuitive Guide to Latent Dirichlet Allocation. Retrieved February 2, 2020, from <https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158>
- Gensim. (n.d.). Gensim: topic modelling for humans. Retrieved from <https://radimrehurek.com/gensim/models/hdpmodel.html>
- Han, H. J., Mankad, S., Gavirneni, N., & Verma, R. (2016, September 2). What Guests Really Think of Your Hotel: Text Analytics of Online Customer Reviews . Retrieved from <https://scholarship.sha.cornell.edu/cgi/viewcontent.cgi?article=1003&context=chrreports>
- Karantzavelou, V. (2020, January 2). European hotel industry: Robust fundamentals with pockets of opportunity in 2020-21. Retrieved from <https://www.traveldailynews.com/post/european-hotel-industry-robust-fundamentals-with-pockets-of-opportunity-in-2020-21>
- Oheix, J. (2018, December 18). Detecting bad customer reviews with NLP. Retrieved January 10, 2020, from <https://towardsdatascience.com/detecting-bad-customer-reviews-with-nlp-d8b36134dc7e>
- Shivam B. (2018, April 23). A Comprehensive Guide to Understand and Implement Text Classification in Python. Retrieved from <https://www.analyticsvidhya.com/blog/2018/04/a-comprehensive-guide-to-understand-and-implement-text-classification-in-python/>
- Wang, C., Paisley, J., & M. Blei, D. (n.d.). Online Variational Inference for the Hierarchical Dirichlet Process. Retrieved from <http://proceedings.mlr.press/v15/wang11a/wang11a.pdf>

Yordanova, S and Kabakchieva, D. (2017, January 5). Sentiment Classification of Hotel Reviews in Social Media with Decision Tree Learning. Retrieved from https://www.researchgate.net/publication/312520895_Sentiment_Classification_of_Hotel_Reviews_in_Social_Media_with_Decision_Tree_Learning

9. Appendices

Appendix A: Hotel Reviews Dataset

| Location | Date/Time | Review (Text Comments) | Review Attributes (Dimensions) | Review Attributes (Measures) |
|---------------|-------------|------------------------|--------------------------------|--|
| Longitude | Review Date | Negative Review | Hotel Name | Additional Number of scoring |
| Latitude | | Positive Review | Reviewer Nationality | Review Total Positive Word Counts |
| Hotel Address | | | Tags | Total number of reviews reviewer has given |
| | | | | Reviewer Score |
| | | | | Days Since Review |
| | | | | Review Total Negative Word Counts |
| | | | | Total number of reviews |
| | | | | Average Score |

Location

- lat: Latitude of the hotel
- lng: Longitude of the hotel
- Hotel_Address: Address of the hotel

Date/Time

- Review_Date: Date when reviewer posted the corresponding review.

Review (Text Comments)

- Negative_Review: Negative Review the reviewer gave to the hotel. If the reviewer does not give the negative review, then it should be: 'No Negative'
- Positive_Review: Positive Review the reviewer gave to the hotel. If the reviewer does not give the negative review, then it should be: 'No Positive'

Review Attributes (Dimensions)

- Hotel_Name: Name of Hotel
- Reviewer_Nationality: Nationality of Reviewer

- Tags: Tags reviewer gave the hotel.

Review Attributes (Measures)

- Average_Score: Average Score of the hotel, calculated based on the latest comment in the last year.
- Review_Total_Positive_Word_Counts: Total number of words in the positive review.
- Review_Total_Negative_Word_Counts: Total number of words in the negative review.
- Reviewer_Score: Score the reviewer has given to the hotel, based on his/her experience
- Total_Number_of_Reviews_Reviewer_Has_Given: Number of Reviews the reviewers have given in the past.
- Total_Number_of_Reviews: Total number of valid reviews the hotel has.
- days_since_review: Duration between the review date and scrape date.
- Additional_Number_of_Scoring: There are also some guests who just made a scoring on the service rather than a review. This number indicates how many valid scores are without reviews in there.

Link to Dataset: <https://www.kaggle.com/jiashenliu/515k-hotel-reviews-data-in-europe/data#>

Appendix B: Topics distribution based on LDA Mallet model

| General | Room Comfort | Food | Reception Service | Room Necessities |
|----------------|---------------------|-------------|--------------------------|-------------------------|
| Stay | Room | Breakfast | Staff | Room |
| Star | Bed | Food | Stay | Shower |
| Hotel | Comfortable | Bar | Reception | Coffee |
| London | Clean | Drink | Made | Water |
| Recommend | Nice | Good | Service | Bathroom |
| Place | Bathroom | Restaurant | Time | Tea |
| Visit | Comfy | Free | Feel | Bath |
| Back | Good | Choice | Check | Facility |
| Thing | Modern | Service | Make | Towel |
| Time | Spacious | Price | Special | Toilet |

| Price | Overall Stay Experience | Accessibility | Room Noises | Staff |
|--------------|--------------------------------|----------------------|--------------------|--------------|
| Day | Good | Station | Room | Staff |
| Night | Location | Location | Door | Friendly |
| Price | Great | Walk | Night | Helpful |
| Pay | Breakfast | Close | Noise | Location |
| Extra | Room | Minute | Floor | Excellent |
| Time | Excellent | Metro | Noisy | Great |
| Paid | Nice | Walking | Morning | Reception |
| Card | Staff | Tube | Sleep | Extremely |
| Money | Clean | Train | Window | Front |
| Charge | Service | Min | Street | Desk |

| Room Size Upgradable | Location Amenities | Hotel Amenities | Room Lighting and Temperature | Booking |
|-----------------------------|---------------------------|------------------------|--------------------------------------|----------------|
| Room | Location | Room | Room | Room |
| Bed | Great | Bar | Air | Time |
| Small | City | View | Work | Check |
| Double | Restaurant | Area | Working | Asked |
| Bit | Close | Pool | Window | Day |
| Single | London | Nice | Hot | Booking |
| Suite | Area | Top | Bit | Told |
| Booked | Good | Lobby | Cold | Arrived |
| Space | Place | Amazing | Light | Reception |
| Upgraded | Parking | Great | Wifi | Pm |