SENTIMENT ANALYSIS OF BT BROADBAND REVIEWS

Karolina Mikiciuk, Radhika Kumar, Elizabeth Arevalo, Isobel Blythe, Sunaina Parvathi

ABOUT THE PROJECT: AIMS

Our project aims to answer the following questions:

1. Which features of broadband products are most important to customers?
2. Which areas of England have the greatest proportion of negative reviews?
3. Which rating criteria (satisfaction/customer service/ speed/ reliability) does BT broadband perform best and worst in?
4. How can BT find a niche within the broadband service market?/ How does BT compare to its main competitors?

Introduction to the problem

There are many broadband services on the market, provided by a variety of different companies and our company (BT) might benefit from performing analysis on the customer review data. This will allow BT to be aware of how customers view our products and hence we will be able to take customer feedback to improve our services.

Our approach to solve the problem

We aim to use a combination of machine learning (NLP) models and statistical analysis. The NLP model will be trained on the reviews themselves, while the metadata such as the ratings, date and location lend themselves well to analysis of their underlying distributions, time series analysis, as well as graphical visualisations. Hence, our project employs a range of analytical techniques.

Data Sources

We have utilised the following site as the source of customer reviews: https://www.broadband.co.uk/broadband/providers/bt/reviews/.
This specific site was chosen for a few reasons:

1. The site has over 3200 reviews about BT across 80 pages, which is the greatest number of reviews from all the sites we considered.
2. This specific site was the most suitable one for web scraping, as other sites required payment in order to use their APIs that exceeded our project budget. Some sites ban web scrapers.
3. It has a wider range of sentiments and review ratings than any other

site we considered (such as Amazon or Trustpilot). Some of the other sites had only a few reviews, all of which were extremely negative. This is expected, as dissatisfied customers are more likely to leave a review than those who are satisfied with the service. However, this means that we have to bear in mind that our analysis has an inherent bias towards negative reviews, as reflected by our data, as well as other review sites.

We collected the review content, review location, review date and review ratings (for satisfaction, customer service, speed and reliability) data for our project from the aforementioned site.
We also used an API to validate the data from the location column - we called the API and saved the results in gb.csv and uk_locations.csv

Data Collection

We opted for automating our data collection process by building a web scraper, as this is a more reliable solution than copy-pasting each review from the website individually in a manual way. The code of our web scraper is contained within the broadband-dot-com-scraper.py file, along with the corresponding utils.py for the functions used by the scraper code.
Our initial version of the web scraper was not able to scrape the review ratings and did not ensure data reliability; when saved in a csv, the review content itself did not always correspond to the actual location of the reviewer. This was caused by some reviews whose reviewers left the location field empty, hence no location data was returned for this review, in turn causing the location of the next review to be assigned to the first review. Thus, the initial data collected was not valid. We solved this problem with the second iteration version of the web scraper, which was able to scrape review ratings and ensured data reliability and validity by returning a triple tilde ~~~ if the field was left empty. Our detailed data cleaning analysis and EDA is what allowed us to see that something is wrong with our data, making us rewrite the web scraper.

Data Cleaning and EDA

The pieces of data that had to be cleaned were the location names and the review content (for the sentiment analysis NLP task). Investigation of null values was used multiple times throughout the project, whenever major changes were implemented in the code in order to ensure data validity and

reliability. This was achieved by comparing the results of the analysis to our previous analysis results and by reasoning the plausibility of these results.

1. Investigating empty fields and null values

In order to ensure high quality of EDA, we performed data cleaning, identifying, deleting and replacing the inconsistent and incorrect information from our data set. Our process involved:

- Checking for null (missing) values across all rows and columns.
- Dealing with the missing values by deleting or replacing it with suitable values.
- Identifying and dealing with duplicate values in our dataset.

Implementation and results of our data cleaning policy:

Null (None) and duplicate values were observed in our dataset and both the observations led us to investigate the cause and hence re-writing/making changes to our original code to scrape data from the website. There were a few null (NaN) values in the location column and it was observed that these values were also missing from the source of our data.

The first stage of data cleaning made us rewrite our web scraping code and generate a new set of clean compact data (.csv) files. Data from all these .csv files are extracted and loaded into a data frame for cleaning and analysis purposes. We repeated the above data cleaning steps after fixing our web scraper to find the null, missing and duplicate values in our data frame.

1. '~~~' is observed in our columns : This is removed by replacing it with empty string ‘’
2. Null/NaN values are observed
   content: 3, location: 58, date: 0, satisfaction: 360, customer_service: 396, speed: 419, reliability: 431
3. There are now no duplicate rows observed in our dataset.

The reviewers are free to leave the location and content fields empty. They are also free to not leave a rating for any of the rating categories. The date field is always present, as it’s automatically generated by the site.

2. Wordcloud EDA

The purpose of the wordcloud is to visualise the most frequently used words in a body of text, in this case our customer reviews. The larger the font size of

the word in the wordcloud, the more it was used. The purpose of this is to ascertain the focus of the reviews, and it could be used to address the issues customers are most concerned about. Putting aside BT, which was the most used word overall, we can see that "time", "service" and "broadband" were highly prevalent. This shows that efficiency and quality of service are important to customers, and either that broadband is the most prevalent product - or the one which raises the most issues. The wordcloud does not answer all of these questions, but it can give initial insight into areas to evaluate.

3. Cleansing the location data

Several functions were created using pure python with the purpose of getting the locations in a standard format. Most of them were created to accommodate the specials needs of the data as the initial exploratory analysis shown that the location column has rows with following attributes:
-Postcodes only (full postcodes or partial postcodes)

-UK or United Kingdom only and UK or United Kingdom plus a location

-Location with partial postcode or full postcode

-Non-valid locations

-Special characters only or special characters with the location

In order to achieve a clean data the following steps were taken:

1. Non-specific values like ‘UK’ or ‘United Kingdom ‘were replaced for unknown.
2. Special characters were removed
3. When only postcodes were provided a free API was used to get a location.

-To validate the location those were compared with a csv file that contains cities, towns, counties and countries of the UK and if the location was not found in the file it was changed to unknown.

4. Review cleansing for sentiment analysis

The necessary steps required to prepare our review data for sentiment analysis were the following:

1. Convert all of the words to lowercase.
2. Remove punctuation.
3. Remove excessive whitespace.
4. Use a SpaCy model to convert all non- these steps can be found in src/nlp_pipeline.ipynb. These steps are required for the NLP models to work properly and this is a standard procedure across the industry in preparing the data for model training and later analysis.

The Sentiment Analysis Model

The sentiment analysis was done using a pre-trained model called VADER (Valence Aware Dictionary and Sentiment Reasoner) - it's a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media, and works well on texts from other domains.

To use VADER, an instance of nltk.Sentiment.SentimentIntensity.Analyzer was created and .polarity_scores() was used on the reviews. After this a dictionary of different scores was obtained. The compound result was used to classify whether a review was positive (compound > 0), neutral (compound == 0) or negative (compound < 0).

The Statistical Analysis

The Statistical Analysis part of the project, was to primarily, understand and interpret the data in order to uncover patterns and trends that would give us an insight into BT's business. The methodology we adopted involved identifying key questions that we aimed to answer, we achieved this by ordering, manipulating, and interpreting raw data from various sources to turn it into valuable insights. This allowed us to present the data in a meaningful way. Post which we carried out an exploratory analysis during which we mined and investigated the data to find connections and generate hypotheses and solutions for specific questions. To conclude, we followed-up with a diagnostic analysis to gain a firm contextual understanding of why something happened, a good example of this would be an increased customer base because of EE acquisition. The next steps of the analysis, which falls outside of the scope of our project timeline, would be to carry out predictive and prescriptive analysis to develop informed projections and practical business strategies.

Areas for improvement

One of the areas for improvement is extractive summary - while in our version control we have code that uses the Rake library to perform keyword extraction, as well as code using the Gensim library for topic modelling of the reviews, the first approach didn't yield any useful insights and thus was abandoned. Meanwhile, the Gensim library in the end required too much computation, rendering our code useless with the amount of computational resources available to us. Hence, both approaches didn't make it into the final version of the project. One way to improve would be to upgrade on resources available to us to take advantage of the Gensim code. Yet another way to improve our NLP model is to add Named Entity Recognition and a model to correct misspellings. This would increase the accuracy of our NLP model and give us extra insights. Some of the reviews identified as positive were in fact sarcastic - in the future we could use a more sophisticated sentiment analysis model that accounts for sarcasm. Some evidently neutral words were also counted as positive, which introduces a bias in our analysis towards classifying reviews as positive.