Master of Technology (Intelligent Systems)

ShopBot

for earphones and headphones

Project Report

Team Members

Shashank Nigam	A0198469A
Lin Wenqi	A0198435R
Ng Mei Ying	A0198546L
Edmund Leow Kwong Wei	A0198458H

Shop assistant icon modified from original by Freepik from Flaticon

Contents

1 Executive Summary

Natural language processing is an emerging filed with new breakthroughs made each day. Building a truly conversational user interface is a perfect example of an application of natural language processing. It involves use of natural language processing and understanding starting from information extraction, knowledge representation to understanding the user query and giving an appropriate response for the query asked. In this project we study the application of natural language processing using Headphones and Earphone recommendation as a use case study. The goal of the project was to build a truly conversational user interface which understands the user queries and responds with a recommendation of headphone or earphone as per his/her requirements. The task involved building an inventory of valid products which involved extracting information from different available sources and valid datasets, understanding and extracting the most common features that qualify a headphone or earphone, ranking the earphones/headphones based on the features extracted and sentiments provided by user reviews. The task involved information extraction, name entity recognition for identification of different brands, model and features, topic modelling for extracting different features and sentiment analysis based on aspects. Owning to the uniqueness of the data set extracted we conducted the aspect-based topic modelling using a deep learning framework involving attention networks. We further built upon a conversational user where we trained model to identify user intents and respond accordingly. The project is available as web-based user interface where the user can converse with the chatbot deployed. The project has given an opportunity to learn and implement the different components involved for building truly conversational system.

2 Business Problem

In today's world, traditional shopping is declining year-on-year, while e-commerce has been rising and is expected to reach 2.05 billion buyers globally by 2020 [1]. This has led to a much greater variety of products at lower prices, as multiple sellers across different countries compete for a slice of the market [2].

However, in a bid to differentiate their products, retailers often advertise their features with marketing jargon that adds to consumer confusion [3]. In addition, with the plethora of online marketplaces [4], even after deciding on the "best item" item, the consumer needs to decide which platform has the "best" price. All these factors result in consumer confusion and decision paralysis, where consumers experience greater anxiety and stress, and sometimes "buyers' regret" after purchase, fearing that they have not chosen the "best" possible product [5, 6]. In 2016, a survey of

1000 Australians found 86 per cent believe too many products was making buying decisions harder [7]. Overwhelmed with choices and decisions, they may even delay or stop shopping all together [8].

In particular, there has been a surge in demand for headphones in the last few years. More than one million headphones were sold per day in 2018 and sales are expected to increase by two-fold in 2024 [9]. Consumers are no longer content with standard earphones that are bundled with mobile phones or MP3 players, and headphones and earphones have become lifestyle accessories for the trendy. Confronted with a large variety of product models, consumers may not always understand the different specifications or find it tedious browsing through countless items to find the product that best suits them.

3 Solution Approach

3.1 Objective

The project objective was to build a product recommendation chatbot for headphones and earphones to give users better "decision simplicity" in their shopping. The main intents are summarised in *Table 1*, and will be discussed in detail in *Chatbot Design*.

Table 1. Chatbot intents and descriptions

Chatbot Intent	Description
Product recommendation	Select suitable products according to rating and sentiment in product reviews
Explanation of terminology	Shortlist typical terminology and store the terms into a lookup table; if exact term is not found, word similarity between query and keys would be performed and closest match will be returned.
Product price and information	Webscrape prices from Amazon periodically (eg weekly/daily), and display price, features and website link to purchase in chatbot

We assumed that we were an e-commerce website that sold headphones (eg Treoo). Because of a very wide selection of products, our customers might face the "Paradox of Choice", and be overwhelmed with choices. In order to increase the conversion rate of web traffic to actual sales, a virtual shop assistant in the form of a chatbot, would be used to identify their needs and facilitate their choice-making decisions. In the event that our store did not hold stock, we would even recommend products from Amazon, so that we would become the one-stop website for headphones.

3.2 Datasets

Several datasets were used for this project. Significant data preparation and cleaning were required for deriving the inventory database of products. This would be described in detail in **Data preparation and cleaning.**

For “Explanation of terminology” intent, several websites were web-scraped to obtain a glossary of terminologies. For “Product Pricing” intent, Amazon website was web-scraped for price and delivery costs based on the list of products that were shortlisted in “Product Recommendation”.

The list of dataset sources is summarised in Table 2.

Table 2. Dataset sources and usage

Purpose	Dataset source	Dataset type
Sentiment Analysis	Amazon Reviews	Text
Information extraction	Product review websites	Text
Aspect extraction	Headphone and earphone reviews from Amazon	Text
Topic modelling	Headphone and earphone reviews from Amazon	Text
Short-listing of products	Shopping websites (Amazon, Treoo)	Text, product IDs
Explanation of terminology	Websites specialising in headphones (Headphones.com, CrutchField.com, Krisp.ai)	Text
Product Pricing	Amazon Website and Treoo Website	Prices

3.3 System Architecture

The chatbot was hosted on DialogFlow due to its ease of setup and wide integration with multiple platforms including Google Assistant, Slack and Facebook Messenger. However, intent classification and entity recognition were performed by Rasa NLU, as it allowed greater customisation in terms of processing and training.

In order to easily embed our chatbot into our website as a pop-up, we leveraged Kommunicate, as it also allowed easy integration with DialogFlow.

When a user enters a message into the chatbot platform, its API sends the message to DialogFlow. Instead of using DialogFlow’s NLP engine to process all intents and entities, we utilised it for only basic categorisation and sent the raw input to our flask webhook app for preprocessing and then forward it to Rasa NLU model to determine intents and entities. The appropriate response is formulated in the flask app and then returned to the user. The detailed system architecture is shown in *Figure 1*.

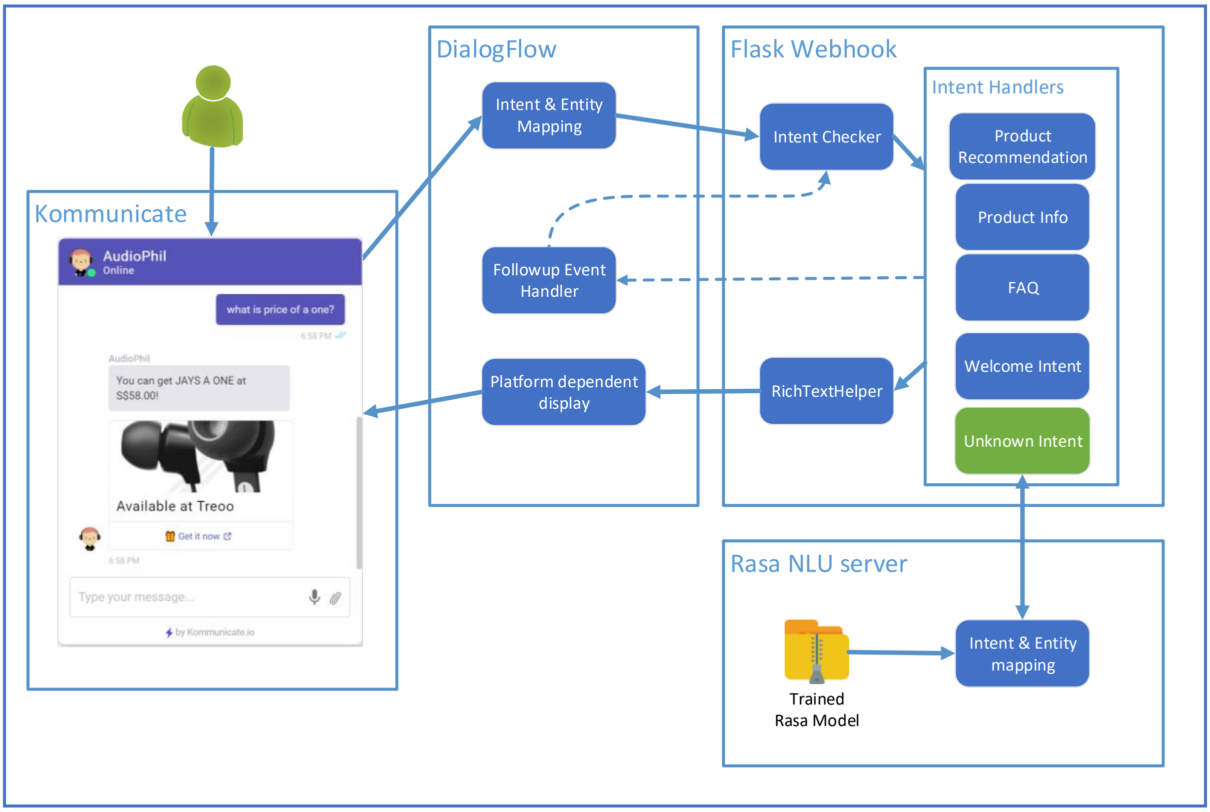


Figure 1. Overall System Architecture

Ideally, for deployment, the Flask app and Rasa NLU server should be hosted in Heroku server. However, there were several problems in doing so[1], and finally for this prototype, it was assumed that the Flask app and Rasa NLU server would be ran locally, and exposed to the internet with a public url provided through ngrok.

[1] See *Challenges and Limitations*

4 Data preparation and cleaning

For preparing a recommendation system to build a chatbot for headphones and earphones we needed an active inventory of product list which was present in Singapore market. For this we needed to scrap data from web to get a list of products available. From among different market available and due to limitations of web scarping we came up with a set of possible sites which could be used for scraping data. *Table 3* describes the use of each of the sites used for web-scraping.

Table 3. Data source site and the information extracted

Web site	Information extracted
Google shopping	Initial inventory of headphones available, their name, rating (if available), content information, seller information, and seller site
Treoo	An Inventory of 900 headphones and 700 earphones along with their specific attributes
Amazon	An Inventory of 506 product list, their related information and their respective product review with customer question answers were extracted

Below section describes the process followed in extracting and further storing the data in required knowledge-based format.

4.1 Product Inventory Extraction using Google Shopping

Google shopping maintains an inventory of different sponsored items, and forms a good starting point for information extraction. In addition to maintaining product list from different leading sellers (Amazon, Shoppee, Lazada etc), it also provides a localized search enabling querying of products available in a region. Each entry in the website further contains a summarized format of different products sold.

Figure 2. Information to be extracted from each item of google shopping list

Based on the data extracted, an initial product list, which formed the basis of information extraction for amazon dataset, was derived. The typical result of this extraction included the following fields shown in *Table 4*.

Table 4. Basis of information extraction

Label heading	Label description
Product Title	Title of the product
Product price	Price of the product
Product seller	Product seller Name
Product seller url	URL of the product website hosted by the sellers
Product Rating	Product rating in stars
Number of reviews	Number of reviews(optional some of products did not include rating in this site)
Product Description	A brief description as stated.

4.2 Data Extraction from Treoo.com

Treoo.com contains an active inventory of most of the headphones and earphones present for sale in Singapore market. With an active inventory of products such as headphones, earphones, accessories and other electronics, the site provided good possible candidate for preparing the inventory list. The website's information was used for preparing an inventory for headphones and earphones. The information extracted from the site was stored in the structured data format which could further be used by the NLP recommender system for effective use. Following approach was followed for extracting the information from the site. The site maintains a directory of items it is selling containing a link to the main page. The site was parsed in two steps. First phase extracting the set of possible products from the site followed by detailed parsing from the each of the individual site. After the inventory of items was created, some of the pre-processing was carried out which involved removing duplicates based on colour and other extra features. *Figure 3* describes the information extraction workflow.

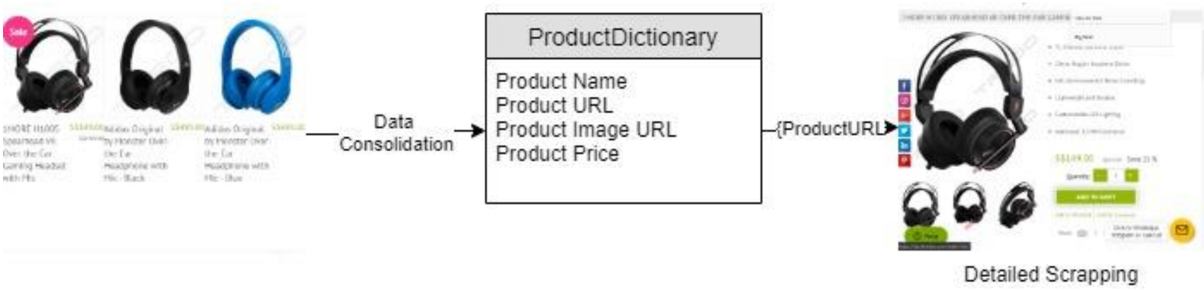


Figure 3. Web scrapping pipeline for Treoo

Detailed site provides information of the headphone/earphone such as

1. Brief description of the product
2. Detailed description of the product.

3. The product feature, technical specification, warranty information, box contents.

Although site contained good information about different products available, it did not contain the information of product reviews. The detailed information contained data in form of a dictionary of items where detailed description described the terms described in the brief product specification. This information was further exploited to prepare a dictionary representation of the information which was further used by the semantic parser. *Figure 4* describes the schematic representation of information extracted from Treoo.

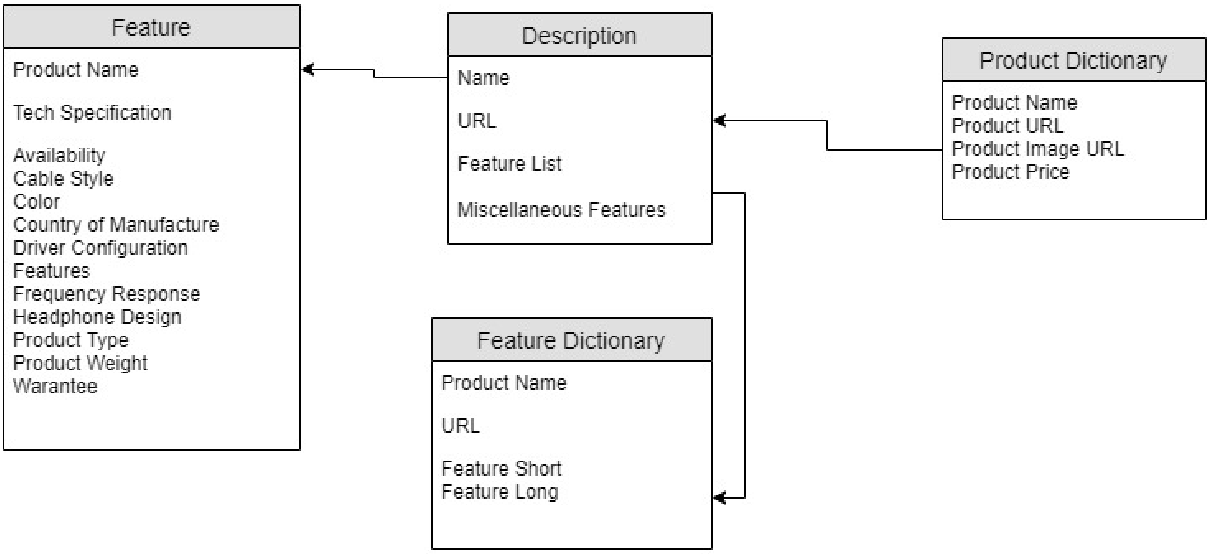


Figure 4. Schema after web scrapping Treoo

4.3 Data Extraction from Amazon.sg

With Treoo we had a considerable dataset of headphone and earphones, however we were lacking in information related to product reviews and general question answers asked by the users. Amazon website had a good source of such information. Also, the reviews provided were from genuine buyers, this prevented biased reviews towards a product. Amazon was mined for the following information

1. ASIN
2. Product Name
3. Product Image URL
4. Rating Information (Number of review and average rating)
5. Brief description
6. Product Information (Product Dimension, Item Weight, Shipping Weight, Manufacturing, ASIN, Item Model Number, Batteries, Best Seller Rank, Date First Published)
7. Review data (5 stars,4 stars, 3 stars, 2 stars, 1 star)
8. Review Detail (Individual review heading, review body, review date time, review rating)
9. Question Answer (Question asked, answer (if available))

Each of the web scraping request was carried out based on the ASIN or Amazon Standard Information Number. Each of these acted as a special identifier for identifying an amazon web product. Thus, for the schematic relationship which was constructed after the information extraction phase. The data extracted can thus be visualized as shown in *Figure 5*. Post web-scrapping the dataset was cleaned for the products who delivered only in Singapore.

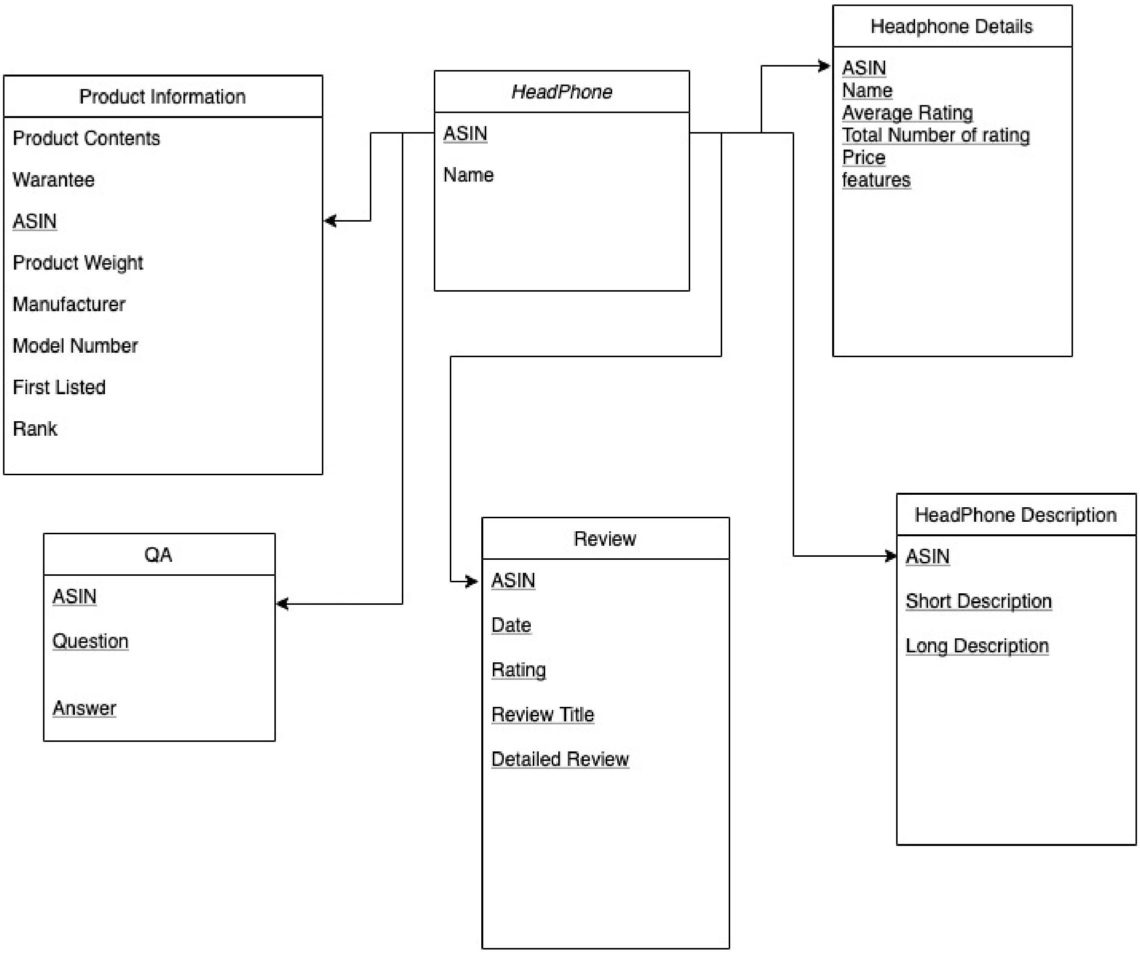


Figure 5. Amazon dataset representation

4.4 Extracting Product Model Features from Product Name

Each of the seller website have their own name convention for naming the products. This results in ambiguity while comparing the same product across different sites. A general format of specifying the product names would be

$$\{Brand\ Name\}\{Set\ of\ Feature\}\{Model\ Name\}\{Set\ of\ Features\}$$

Here the order of occurrence of BrandName, set of features, model name could be randomized. This prevented us from comparing same products across different sites effectively. This can be further illustrated as in *Table 5*, considering the same brand product from same and different models

Table 5. Comparing products across different sites

Seller	Product Name	Brand/Model Name	Cosine similarity (spacy_web_md)	Cosine similarity (corpus w word embedding)
Amazon.sg	Sony WH-CH700N Wireless Noise Cancelling Headphones, Black	Sony, WH-CH700N	1	1
Lazada	Paris Sony Wireless Bluetooth Noise Canceling Over Ear Headphones Headsets Wh-Ch700n	Sony, Wh-Ch700n	0.93	0.86
Amazon	Sony WH-1000XM3 Bluetooth Over-Ear Noise Cancelling Headphones	Sony, WH-1000XM3	0.91	0.73

Even though the model changed, there was not significant drop with the similarity score. This allowed us to compare two models with same brand but not able to distinguish between different models. Thus, for the better integrity of the data set, it was important to extract the brand name and model name from product name.

This was based on a rule-based system combined with the grammar structure of the model Name. We extracted the part of speech tag from the name using Spacy's parts of speech (POS) tagging. This was further combined with product model with some hand-crafted features to give product model name and brand description.

1. Based on the product name, brand name was extracted first. For this approach, consider the first word of the product name as the possible candidate of the brand. Based on this, a set of brand names were formed and refined. Other possible set of key indicators could be the key words such as from, by, is usually followed by the product name.
2. Feature extraction: After Brand name was extracted, the next step in the process was the extraction of the feature. This was followed by a set of grammar rules described below:
    a. [ADJ]* [NOUN]
    b. [NOUN]

c. [ADP] [DET]?[NOUN] : for features such as over ear, over the ear, in ear etc

d. [VERB][ADP]: for verbs in list such as cancelling

The possible set of candidates were further matched with glossary items to get accurate feature map. The ones which were not mapped accurately were removed from the text.

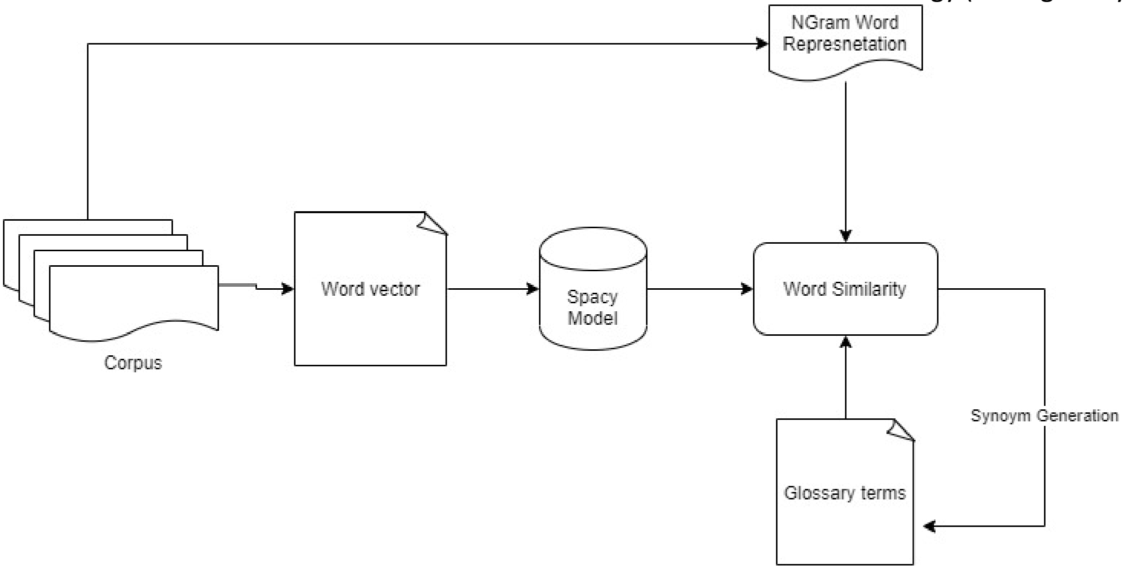
The text remaining after the extraction would indicate the brand model name if the string was not empty after step2. The process was validated with model names of few of the products obtained during the process. The extraction feature is summarized in `OrgnaizationNameExtraction.ipynb`. Thus, after the extraction of the brand and feature names we could get output similar to the one in *Table 6*.

Table 6. After brand and feature extraction

Product Name	Brand	Model	Feature
Sony WH-CH700N Wireless Noise Cancelling Headphones, Black	Sony	Wh CH700N	Wireless,noise cancelling, headphones, black
Sony WH-1000XM3 Bluetooth Over-Ear Noise Cancelling Headphones	Sony	Wh 1000XM3	Bluetooth, over ear, noise cancelling, headphones

4.5 Synonym Generator

For making the intent classification more robust for testing needed a possible candidates of the keywords similar to one being queried for the intent classifier. In order to achieve grounding in chat bot we find set of possible key words or synonyms used. Since the dataset we extracted was very relevant for the domain, we used similarity based on word vectors constructed from the corpus which included (product description, product details, review data, feature set, question answer data which was extracted). Based on the vocabulary of words we obtained, we extracted the word vectors if the similarity score was more than .60 on the unigrams and bigram of the word vectors extracted. The data was further used for training the intent classifier model.



5 Chatbot Design

This section discusses the different aspects of our chatbot design, including chatbot persona, entities, intents and conversation flows.

5.1 Chatbot Personality

Our target audience will be mainly young people within the age range 18-34., as these were deemed to be the people that had the highest chance of intending to purchase earbuds within the next 12 months [10].

There are 12 archetypes commonly used to define a brand persona, which are functionally identical to the 12 Jungian archetypes [11]. As our target audience are young people, we will stereotype them as Explorers, who are ambitious and always seek out new things. In order for them to identify well with our bot, we will mirror the personality of our audience [12] and also give it the same archetype Explorer, with a friendly, energetic and enthusiastic personality. This will be projected in its interactions with our users. For example, when returning the price for a specific item, it might respond with something like "Woohoo! It's your lucky day! 🤩 JAYS A ONE is available at S$58.00!"

In addition, in order to not be repetitive, there would be multiple variations for each response of our chatbot. This was catered for by having choosing randomly from lists at different parts of the response, thereby having many multiple permutations.

In coming up with a name, it was important that we had a memorable name that was linked to our core business of earphones and music, so that our users could easily remember it. We came out with

AudioPhil, as a play on the word “audiophile”, who is a person who is enthusiastic about high-fidelity sound reproduction. Phil is also uncommon enough while being an easy name to remember.

In coming up with an avatar picture, we wanted the users to have reasonable expectations of our chatbot. It was found that consumers expected chatbots to behave like human agents but wanted it to be clear that they are bots [13]. In addition, if the chatbot was too convincing as a human, people were more likely to speak quickly and less clearly, and have higher expectations for the system [14]. Hence, we adopted a cartoon-like avatar so that they will know immediately that it was a bot. At the same time, we wanted it to appear friendly, hence we opted for a human avatar. Since our chatbot is supposed to be like a virtual shop assistant, and since we recommend headphones, it would be nice if he had headphones on him.

With all these in mind, our final chatbot persona is given in *Figure 6*.

AudioPhil

Friendly • Energetic • Attentive

Age: 20
Occupation: Earphone Purchase Specialist
Location: Singapore
Archetype: Explorer (ENTP)

Interests and Passions

- Listening to music
- Audio history and latest technology in headphones
- Trying out the newest headphones on the market
- Helping people find their ‘headphone soulmate’

Quotes

“Hello there! I am Phil. Let’s work together to discover the perfect headphones for you today!”

Figure 6. Chatbot Persona

5.2 Intent Classification and Entity Recognition

Basic natural language processing (NLP) was first carried out using DialogFlow. However, it had several major limitations such as (i) unable to perform customised preprocessing and processing of inputs, and (ii) unable to easily input a large list of training and test phrases. Hence, we adopted a hybrid approach, where intents and entities were defined within DialogFlow, and a first-round basic intent classification and entity recognition was done. If DialogFlow fails to recognise the intent, second-round processing for intent classification and entity recognition was done by Rasa.

This allowed us to adopt the following custom preprocessing pipeline for user inputs as shown in *Figure 7*. By performing such preprocessing, we simplified the training examples that were required (eg. No need to consider Singlish; no need to consider both singular and plural tenses, etc).

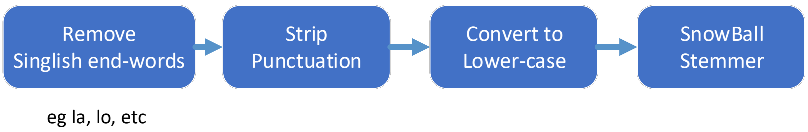


Figure 7. Custom pre-processing pipeline

Training example datasets were generated using a third-party tool [Chatito](https://rodrigopivi.github.io/Chatito/), by defining examples in a custom domain specific language (DSL), which simplified the generation of samples. A snippet of the chatito file used to generate training samples for "Explanation of terminology" is given in *Figure 8*. From the simple definitions below, training and testing datasets could be easily generated by iterating and permutating through the different combinations that were defined.

In order to streamline our workflow, a script (*_generate_lookup_files.py*) was also created to generate lookup tables for the entities of the intents, as well as to add these to the chatito file that would be used to generate training and testing examples.

```
import ./common.chatito

# https://rodrigopivi.github.io/Chatito/
# Chatito file for intent_whatis
%[intent_whatis]('training':'1000', 'testing':'1000')
    *[30%] ~[hi?] ~[name?] ~[could] I know what ~[is] @[term] ~[thanks?]
    what ~[is] @[term]
    can explain @[term]
    I would like to know about @[term]
    need to know @[term]
    tell me about @[term]
    why ~[is] @[term] ~[useful]
    *[5%] @[term]
```

Figure 8. Example snippet of chatito file

In Rasa NLU, incoming messages are processed by a sequence of components that will executed one after another in a so-called processing pipeline. The three main parts are (i) tokenization, (ii) featurization and (iii) entity recognition/intent classification/response selectors[2]. The following pipeline, supervised embeddings in *Figure 9* was used. This was so that word vectors could be customised for our specific domain.

[2] See https://rasa.com/docs/rasa/nlu/choosing-a-pipeline/#understanding-the-rasa-nlu-pipeline for details

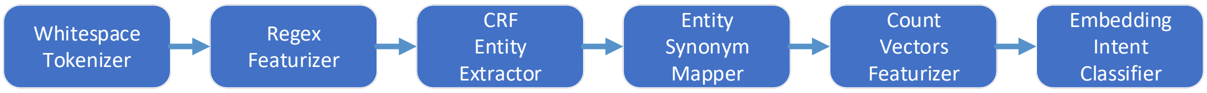


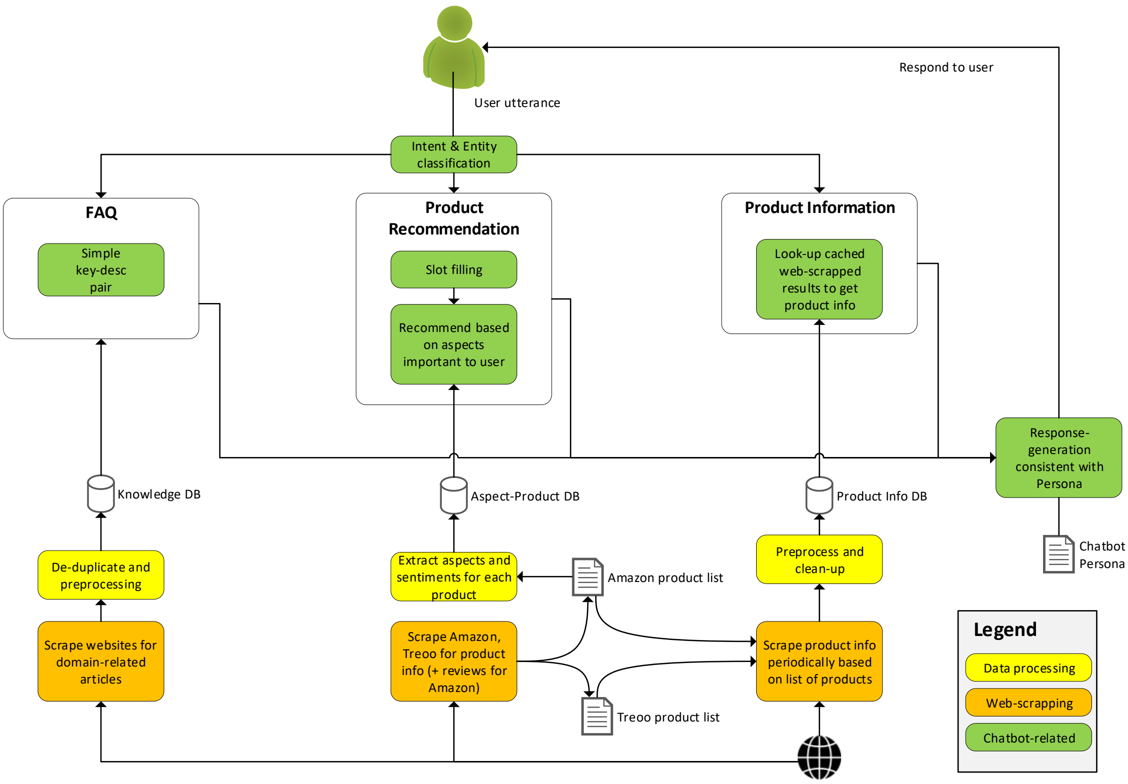
Figure 9. Rasa pipeline using supervised_embeddings

5.3 Intent Details

An overview of intents is shown below. There are three main intents for our system

- Product Recommendation
- Explanation of terminology (FAQ)
- Product Information

In general, each of the intents rely on a database that was scraped from websites as described previously.



5.3.1 Sentiment analysis Product Recommendation

Sentiment analysis can be performed at different levels depending on the degree of granularity required: document-level, sentence-level, and aspect-level. Here, we attempted to do sentiment analysis at the aspect-level to provide more information to consumers and aid their purchase decision. Aspect-based sentiment analysis involves the following subtasks: aspect term extraction, aspect term polarity, aspect category detection, and aspect category polarity. [15]

Datasets consisting of Amazon product reviews of earphones and headphones were obtained from Price API (), which provides e-commerce market data, and by web scraping. An unsupervised neural method was used to extract relevant aspects from reviews [16, 17]. It involves using an attention mechanism to filter neural word embeddings within a sentence which are then used to build aspect embeddings. We varied the number of aspect categories during testing and the aspect size was fixed to fifteen in the end as this produced semantically meaningful aspect categories. We could easily infer the different aspect categories and four categories that might be important to consumers were selected for further analysis. The four inferred categories are design, fit, price and sound. A limitation of this unsupervised approach was that noun phrase aspect terms could not be extracted.

We then manually screened the results and omitted certain implicit aspect terms in each category to focus on explicit aspects. A rule-based approach [18] based on dependency relations was then used to extract the opinion words associated with the aspect terms (*Table 7*). Stanford CoreNLP was to perform POS tagging and dependency parsing at the sentence level for each review. Minimal pre-processing was done to maintain the semantic relationship between words, and stop words were removed. The type dependency relations (TDR) nsubj, amod, dobj, nmod, acl and conj between governor and dependent were used to identify preliminary aspect-opinion pairs.

Table 7. TDR POS tag patterns

TDR	POS Tag Pattern		Aspect	Opinion
	Governor	Dependent		
nsubj	JJ/JJR/JJS	NN/NNS/NNP	dependent	governor
nsubj	VB/VBD/VBG/VBN/VBP/VBZ	NN/NNS/NNP	dependent	governor
amod	NN/NNS/NNP	JJ/JJR/JJS	governor	dependent
amod	NN/NNS/NNP	VB/VBD/VBG/VBN/VBP/VBZ	governor	dependent
nmod	JJ	NN	dependent	governor
acl	NN	JJ	governor	dependent
acl	NNS	VBP	governor	dependent
conj	NN/NNS/NNP	JJ	governor	dependent
conj	NN/NNS/NNP	VBZ	governor	dependent

Inspecting the results, some of the rules (Table 8) largely failed to retrieve correct aspect-opinion pairs and results obtained using these rules were discarded.

Table 8 Omitted rules

TDR	POS Tag Pattern		Aspect	Opinion	Example
	Governor	Dependent			

nsubj	VBZ	NN	dependent	governor	produces (VBZ) † the most astonishingly detailed sound (NN) ‡
amod	NN/NNS/NNP	VB/VBD/VBP/VBZ	governor	dependent	wired (VBD)headphones (NNS)
acl	NNS	VBP	governor	dependent	these headphones (NNS) try (VBP) to recreate every sound
conj	NN/NNS/NNP	VBZ	governor	dependent	the flat nature of the ear piece allows to you to lay down with the headphones on without too much bulk (NN) and the overhead band keeps (VBZ) the ear pieces in place

† opinion (POS)

‡ aspect (POS)

VADER (Valence Aware Dictionary for sEntiment Reasoning), a rule-based model, was used to determine the sentiment of the aspect. Word tokens between the aspect and opinion were used to classify whether the sentiment towards the aspect was positive, neutral or negative. Of note, it was observed that VADER sometimes did not recognize of the context of the words in which they appeared. For instance, "The lows are tight , crisp" was classified as having negative sentiment presumably because the word "low" has a negative meaning.

For each product, the proportion of positive, negative and neutral aspect terms in each aspect category was obtained. An aspect rating on a five-point scale was then calculated using the formula below:

Rating = P_{neg} * (5 / 3) + P_{neu} * (10 / 3) + P_{pos} * 5

P_{neg}: proportion of negative aspect terms

P_{neu}: proportion of neutral aspect terms

P_{pos}: proportion of positive aspect terms

The review ratings from consumers were also converted to a five-point scale for each product. When recommending products to chatbot users, the products were ranked based on the average of the aspect and review ratings.

5.3.1.1 Product Recommendation

Based on user's preferences, including whether he would like earphones or headphones, the connectivity type and desired brands, results are returned to the user. There are four different kinds of intents, three for the preferences and one for confirmation of selection. At least two and up to ten results are displayed. The sentiments for the aspects design, fit, price and sound are displayed, along with product name and price. The review rating from Amazon consumers is also provided.

5.3.2 Explanation of Terminology

In order to get a list of domain-specific terminologies, we searched for websites that had ready-made glossary terms for headphones, and finalised on glossary terms from three websites (See *Table 9*). In cases where there were repeated glossary terms, we just took the description that was shorter. Interestingly, the number of overlapping terms was small, which helped to expand our range of glossary terms, giving us a total of 116 terms (See *Figure 10*). All web-scraping was performed in the notebook `Glossary.ipynb`.

Table 9. Websites for glossary terms

Website	Website URL	Total terms
Headphones.com	https://www.headphones.com/pages/glossary	63
Crutchfield.com	https://www.crutchfield.com/S-P3Fy2Oh1DMm/learn/headphones-glossary.html	36
Krisp.ai	https://krisp.ai/blog/glossary-headphone-terms/	28

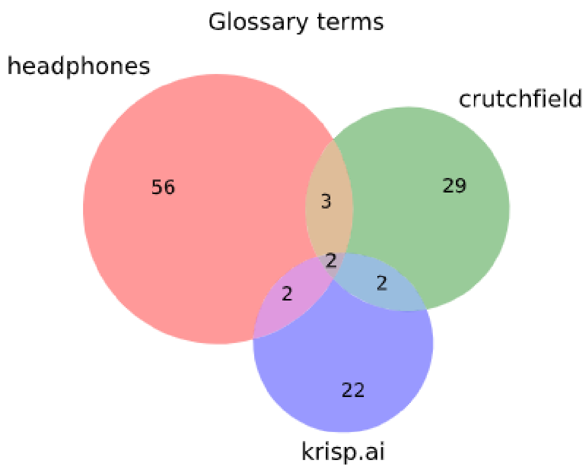


Figure 10. Venn diagram showing overlaps of glossary terms among the webscraped sites

If the search query was found in our list, we would return the description and source of information to the user. In the event that it was not found, we would perform similarity matching of query with terms in the database, based on word vectors using spacy model 'en_core_web_md' (English multi-task CNN trained on OntoNotes, with GloVe vectors trained on Common Crawl), and return the closest matched term. Otherwise, we would just tell the user that we do not know what that term is.

See **User Guide** for test scenarios for the intent.

5.3.3 Product Information

We had two lists of products from Treoo and Amazon Singapore. For Amazon, we realised that the international site had a lot of products that did not ship to Singapore but had more sellers, in different conditions (new or used), different delivery cost (free or not, and if not, price differed greatly depending on type of delivery).

In order to simplify these considerations, we instead took products from Amazon Singapore, which all had free delivery and had only new items. There were numerous items that did not state a brand (*See Figure 11 and Figure 12*). Filtering for only products that had a brand, model and price, we had a total of 87 brands consisting of a total of 230 models.

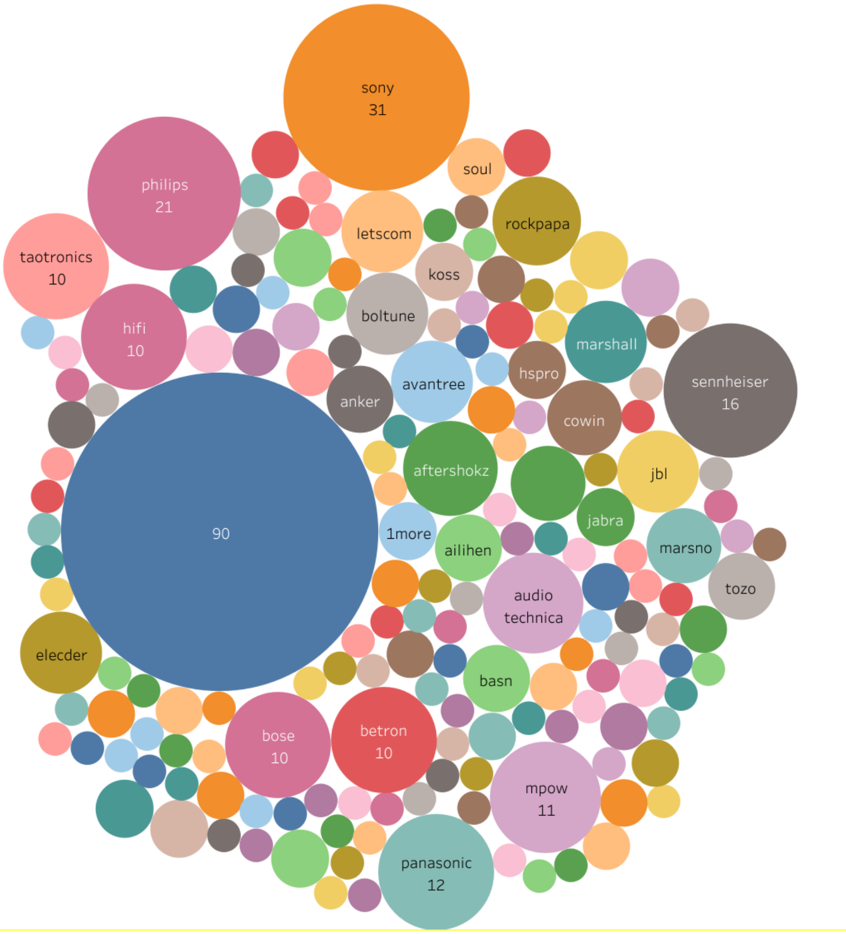


Figure 11. Packed bubble chart showing Amazon products before filtering

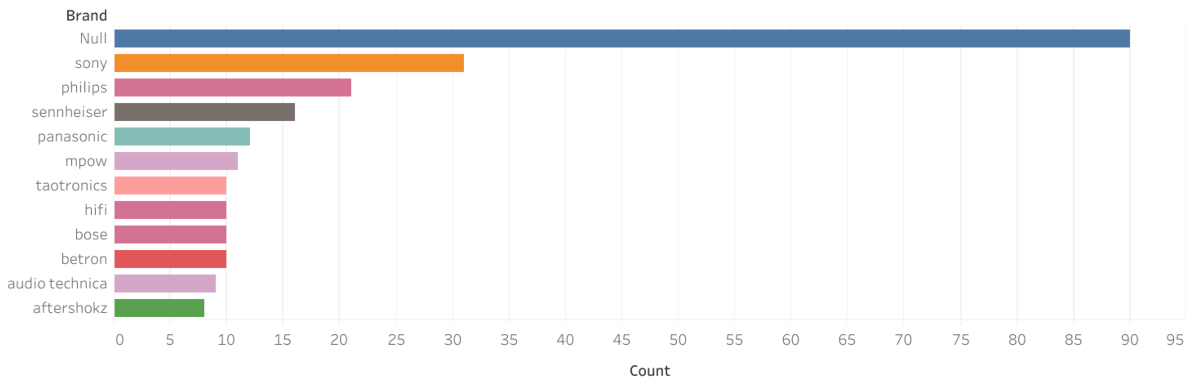


Figure 12. Amazon product frequency by brand before filtering

For Treoo, there was a much greater number of total models (*see Figure 13 and Figure 14*), but product name was very inconsistent. Some inconsistencies, using AKG models as examples, are shown in *Table 10*. Therefore, various pre-processing methods had to be done to standardise them. In total, there were 144 brands and 1643 different models.

Table 10. Example inconsistencies in Treoo product names for AKG brand

Example	**Comment**
AKG K52	Ideal format
AKG K 545	Extra space between K and 545

AKG K845BT Wireless Bluetooth Over-the-Ear Headphone	Features are also included in the description

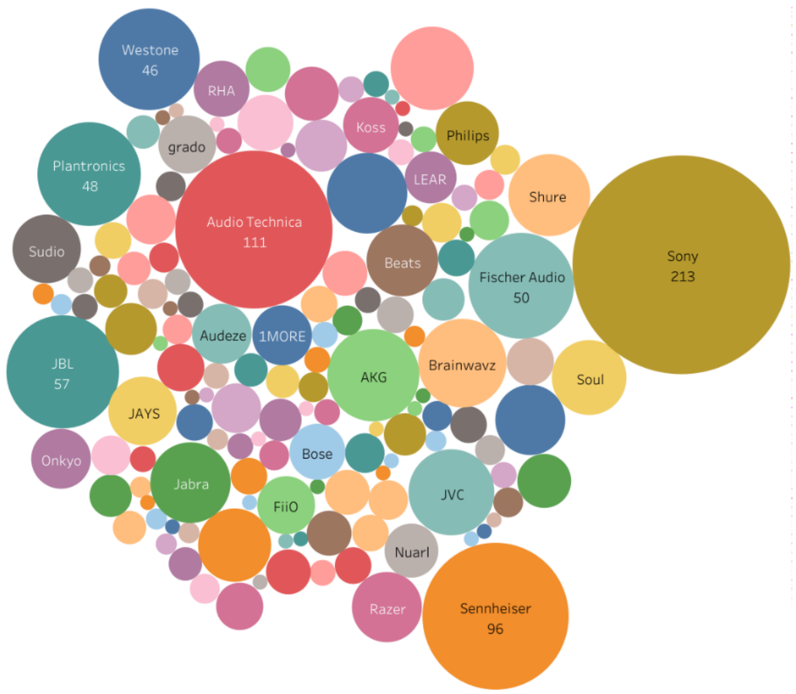


Figure 13. Packed bubble chart showing Treoo products before filtering

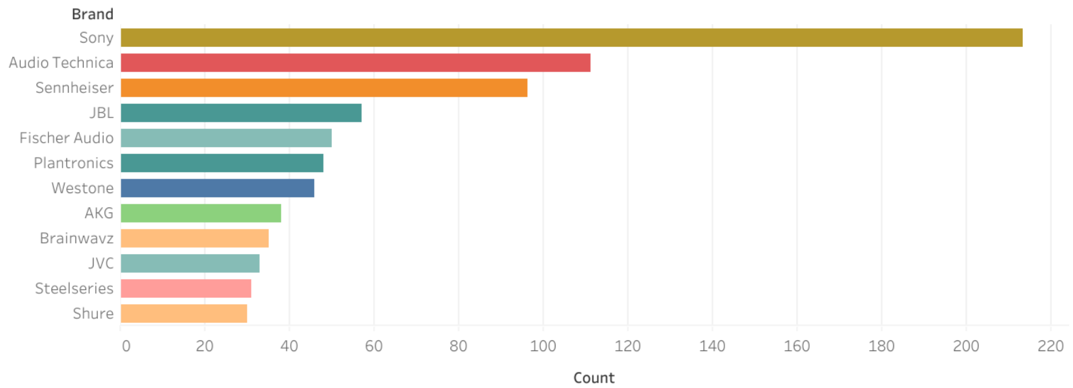


Figure 14. Treoo product frequency by brand before filtering

We combined both lists of brands and models and found that there were 29 brands that overlapped. However, there were only 24 models that existed in both Treoo and Amazon (See *Figure 15*). This small overlap suggested that there were possibly similar models that should be regarded as the same model. However, to capture more variation in user queries, and in the interest of time, we decided not to further process the model names.

For the same brand and model, we took the item with the lower price. It was found that when the same item existed in both Treoo and Amazon, Amazon was almost always the cheaper one.

All pre-processing was performed in the script file *_generate_lookup_files.py.*

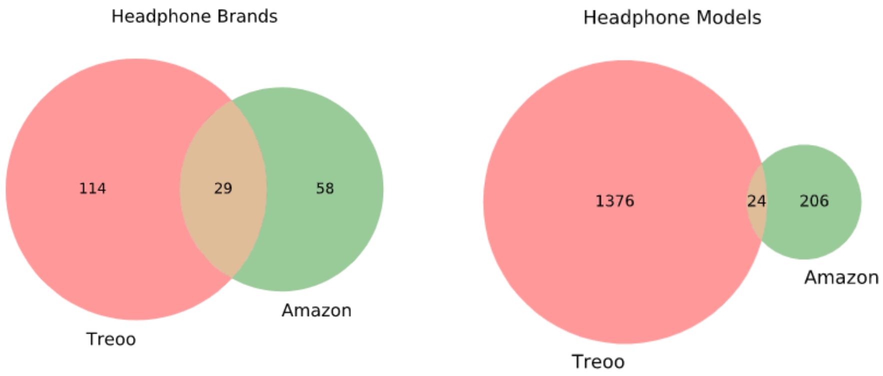


Figure 15. Venn diagrams showing Treoo and Amazon brands and models

In order to match the user's query, we would first check if a product model was given. If not, we would ask them for it. If the model is unique (ie. Only 1 brand has this specific model), we could then retrieve the results. Otherwise, we would ask them for the brand name as well. If a suitable brand-model pair was found, then results would be returned.

See **User Guide** for test scenarios for the intent.

6 Test Results

The intents for "Explanation of Terminology" and "Product Information" were trained on 1000 samples and tested on another 1000 samples each in Rasa NLU. Intent classification evaluation was performed via [rasa's in-built functionality]. The intent prediction confidence distribution showed that most samples were predicted with a very high confidence (*Figure 16*). The confusion matrix also showed that most intents were correctly classified (*Table 11*).

Table 11. Confusion table for intents

	greet	intent_price	intent_whatis
greet	100	0	0
intent_price	1	998	1
intent_whatis	0	0	1000

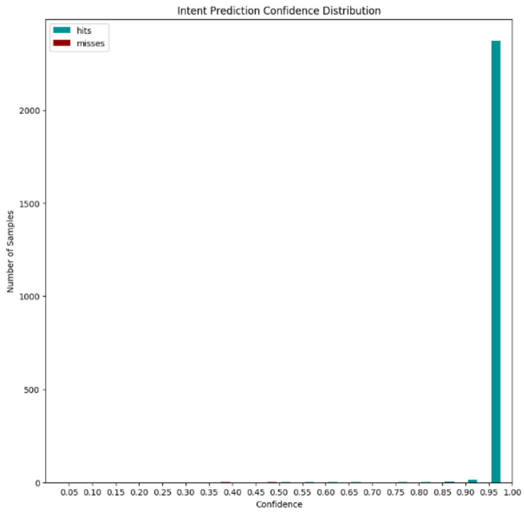


Figure 16. Intent Prediction Confidence Distribution

For “Product Recommendation”, intent classification and entity recognition were done mainly in Google DialogFlow, which did not require a large training set, and had no easy way to evaluate performance.

7 Challenges and Limitations

We encountered several issues with DialogFlow, Heroku deployment, and had some library conflicts while working on the chatbot and will elaborate on them in the following subsections.

7.1 Heroku Deployment

Initially, we deployed our system automatically to Heroku. However, later on in our development, we realised the limitations of the free tier of Heroku.

Firstly, we exceeded the maximum slug size limit of 500MB was exceeded, leading to a failed deployment. This was because we were using many different libraries especially in terms of NLP (eg spacy model and rasa). To circumvent this, the flask webhook was hosted on one app, and rasa NLU server was hosted on another.

Next, we exceeded the memory quota (errors R14 and R15) where our dyno in Heroku required too much memory. This was due to loading of too many large variables into memory, such as the spacy 'en_core_web_md' model (91MB), and various lookup tables for our querying of headphone models and brands. We tried to switch to the spacy ‘en_core_web_sm’ model and managed to deploy, but faced performance degradation for similarity matching.

After lots of time trying to fix the issue, we finally gave up in the interest of time, and decided to only cater for a local deployment. However, these errors would be non-existent if using higher tiers of Heroku such as the Performance M which has 2.5GB RAM.

7.2 DialogFlow

DialogFlow has an unconfigurable timeout of five seconds for webhook responses. This feature was to ensure that the chatbot responds within an acceptable time limit for natural conversation. However, this meant that we could not fetch real-time price information for the user via web-scraping, as it would exceed the time constraints. Hence, we resorted to using cached data for our results.

In an actual deployment, we would periodically web-scrape (eg weekly) and update our data cache to ensure that prices were not too different from the current price. An alternative was to subscribe to APIs that allowed the retrieval of product price and information from multiple sellers, which would be typically much faster than web-scraping. Unfortunately, there were no free sources of APIs for this, and we could not choose this option.

7.3 Python library conflicts

Initially, for “Explanation of Terminology” intent, we wanted to also provide context-based Question and answer functionality based on the description in the glossary, so that the user could ask follow-up questions. We utilised deeppavlov which had a pre-trained BERT model trained on the SQuAD dataset. Initial experiments yielded pretty good results.

However, later on, we realised that there were conflicts in dependencies with other libraries in our code. After spending some time debugging and trying to resolve them, we decided to forgo this functionality as it was a good-to-have.

8 Conclusions

We were able to build a conversation user interface which understands the user intents and recommends the product accordingly. This was accomplished by using different natural language understanding techniques. We employed the use of name entity recognition which exploited under hood part of speech tagging for an unsupervised approach for extracting different models, brands and features. We employed use of an unsupervised deep learning framework for extracting different aspects from the corpus collected. This involved use of attention network mechanism. Based on the aspect extracted we further qualified the different products on the aspects. We employed the use of aspect extraction on each of the review, and sentiment analysis for scoring the review extracted from amazon. A Vader system was employed for successful sentiment analysis. For building a successful user interface we employed use of Google Dialogflow and rasa NLU for intent classification and entity extraction. This was integrated with the knowledge base created for providing most suitable response for the user. For better user access we deployed the system as a web interface. Kommunicate.io was used as a bridge for between the chat bot and the web interface. With the implementation of the project we were success fully able to employ different natural language understanding techniques such as information extraction, name entity recognition, aspect mining, topic modelling, sentiment analysis, intent classification which allowed us to build a truly conversational UI for headphone/earphone recommendation. This application further could greatly hep a visitor to the website to narrow down his choices or get relevant information that he required. This can further help increasing sales conversion and increasing the revenue for the sellers as well. A conversational interface could act as bridge between the users and sellers allows users to buy the best product while keeping the competition fair.

9 References

[1] Oberlo, “Find out How Many People Shop Online in 2020,” [Online]. Available: https://www.oberlo.com/statistics/how-many-people-shop-online.

[2] “The growth of e-commerce means more choice and better deals,” The National, 2 Jan 2018. [Online]. Available: https://www.thenational.ae/opinion/editorial/the-growth-of-e-commerce-means-more-choice-and-better-deals-1.692313.

[3] R. Carter, “Lost in translation: The dangers of marketing jargon,” Fabrik Brands, 1 Aug 2019. [Online]. Available: https://fabrikbrands.com/the-dangers-of-marketing-jargon/.

[4] “The biggest online marketplaces that you should know,” IONOS, 6 8 2019. [Online]. Available: https://www.ionos.com/digitalguide/online-marketing/online-sales/the-biggest-online-marketplaces-that-you-should-know/.

[5] B. Schwartz, The paradox of choice : why more is less, New York: Ecco, 2004.

[6] L. M. Lyengar S, “When choice is demotivating: Can one desire too much of a good thing,” *Journal of Personality and Social Psychology,* vol. 79, pp. 995-1006, 2000.

[7] E. Han, “Analysis paralysis: Why shoppers spoilt for choice often regret purchases,” The Sydney Morning Herald, 14 Jul 2016. [Online]. Available: https://www.smh.com.au/business/consumer-affairs/analysis-paralysis-why-shoppers-spoilt-for-choice-often-regret-purchases-20160713-gq4rak.html.

[8] Knowledge@Wharton, “Online Shopping Choices: Less Is Sometimes Better Than More,” Wharton University of Pennsylvania, 17 Dec 2013. [Online]. Available: https://knowledge.wharton.upenn.edu/article/online-shopping-choices-less-sometimes-better/.

[9] A. A. &. Intelligence, “Global Earphones and Headphones Market to Reach Values of $36 Billion During the Period 2018–2024,” PRNewswire, 14 Feb 2019. [Online]. Available: https://www.prnewswire.com/news-releases/global-earphones-and-headphones-market-to-reach-values-of-36-billion-during-the-period-20182024--market-research-by-arizton-300795642.html.

[10] T. O. Team, “The State of Play 2019: What’s next for audio tech,” Qualcomm, 5 Sept 2019. [Online]. Available: https://www.qualcomm.com/news/onq/2019/09/05/state-play-2019-whats-next-audio-tech.

[11] N. Smith, “Does Your Brand Have Multiple Personality Disorder? A Look at Brand Archetypes,” nvision designs, 10 November 2015. [Online]. Available: http://www.nvision-that.com/design-from-all-angles/what-is-your-brand-personality-a-look-at-brand-archetypes.

[12] T. Manifest, “How to Give Your Chatbot a Personality,” 11 Dec 2018. [Online]. Available: https://chatbotslife.com/how-to-give-your-chatbot-a-personality-5e0fb239b28c.

[13] Leah, “What Do Your Customers Actually Think About Chatbots?,” Userlike, 30 Jan 2019. [Online]. Available: https://www.userlike.com/en/blog/consumer-chatbot-perceptions.

[14] K. Waddell, “Chatbots Have Entered the Uncanny Valley,” The Atlantic, 21 Apr 2017. [Online]. Available: https://www.theatlantic.com/technology/archive/2017/04/uncanny-valley-digital-assistants/523806/.

[15] S.-2. T. 4, “Task Description: Aspect Based Sentiment Analysis,” 12 May 2014. [Online]. Available: http://alt.qcri.org/semeval2014/task4/. [Accessed 30 March 2020].

[16] W. L. H. N. D. D. R. He, “An unsupervised neural attention model for aspect extraction.,” *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics,* vol. 1: Long Papers, 2017.

[17] madrugado, “Attention-Based-Aspect-Extraction,” 24 March 2020. [Online]. Available: https://github.com/madrugado/Attention-Based-Aspect-Extraction. [Accessed 30 March 2020].

[18] V. MacMillan, “How to Develop a Chatbot Persona That Fits Your Brand,” Medium, 26 April 2018. [Online]. Available: https://chatbotslife.com/how-to-develop-a-chatbot-persona-that-fits-your-brand-c48055970372.

[19] T. F. Tay, “Tourist arrivals, spending in Singapore at record highs,” Straits Times, 14 February 2019. [Online]. Available: https://www.straitstimes.com/singapore/tourist-arrivals-spending-at-record-highs. [Accessed 28 August 2019].

[20] N. S. M. M. A. A. A.S Shafie, “Aspect extraction performance with pos tag pattern of dependency relation in aspect-based sentiment analysis.,” in *2018 Fourth International Conference on Information Retrieval and Knowledge Management (CAMP). IEEE*, 2018.