NLP Task 1: Topic Modelling (750 words) 20%

***Literature Review and Rationale: (150 words)* 15% 311w**

As the internet affects how business is conducted, customer-generated reviews are becoming an increasingly important part of e-commerce. Collecting and presenting such information valuable to consumers is a core part of companies such as TripAdvisor.

TripAdvisor reviews are typically written by customers who have previously used a service or product. TripAdvisor makes the reviews widely accessible so potential customers can read them and use the information for their own purchasing decisions. A standard review format is a block of text and a numeric rating (1-5 bubbles) that summarises the customer's sentiment toward the accommodation in a single number. However, such a rating scheme is of limited use to the customer and the hotels themselves. Hotels are interested in the overall quality of the customer's experience and its quality in specific aspects. For example, a hotel interested in customer reviews may be interested in distinct aspects such as quality of the room, quality of the hotel facilities, location of the hotel, helpfulness of the staff and perceived value concerning the price. This suggests that one overall rating may be less meaningful than a set of ratings specific to each aspect of interest.

An area of Natural Language Processing (NLP) that can assist hotels better to understand the needs of their customers' needs and ultimately improve their customer experience is Topic Modelling (Shen, 2012). Topic Modelling, a type of unsupervised data mining technique, constitutes a popular tool for extracting important themes (topics) from unstructured data and is employed to reveal and annotate extensive documents collection with thematic information (Christodoulou, 2020). Latent Dirichlet Allocation (LDA) will be used in this study as a topic-modelling technique. In LDA, a topic is a probability distribution function over a set of words used as a type of text summarisation. Using Bayesian probabilities, LDA then expresses the relationships between words in terms of their affinity to certain latent variables (topics) (Alexander, Blank, & Hale, 2018).

***Data: (200)* 20% 122w**

Wrangling

To tokenise the words in the customer reviews, the **Gensim** package and **simple_preprocess()** function was used.

As discussed in part *2.3.3*, bigrams and trigram of the text was created to understand the frequency of words occurring together. The *min_count* and *threshold* arguments were employed to increase the difficulty of combining words to improve the quality of the bigrams and trigrams. The following code was used to achieve this:

```
# Create bigram and trigram

bigram = gensim.models.Phrases(data_words, min_count=5, threshold=100) # higher threshold fewer phrases.

trigram = gensim.models.Phrases(bigram[data_words], threshold=100)
```

Lemmatisation is the process of converting a word to its base form. Lemmatisation first considers the context and then converts the word to its meaningful base form whereas stemming removes the last few characters, potentially leading to incorrect meanings and spelling errors (Balakrishnan & Llyod-Yemoh, 2014). Therefore, lemmatisation was used over stemming and carried out via the following code:

```
# Define functions for stopwords, bigrams, trigrams and lemmatisation
def removeStopwords(texts):
    return [[word for word in simple_preprocess(str(doc)) if word not in stop_words] for doc in texts]

def makeBigrams(texts):
    return [bigram_mod[doc] for doc in texts]

def makeTrigrams(texts):
    return [trigram_mod[bigram_mod[doc]] for doc in texts]

def lemmatization(texts, allowed_postags=['NOUN', 'ADJ', 'VERB', 'ADV']):
    """https://spacy.io/api/annotation"""
    texts_out = []
    for sent in texts:
        doc = nlp(" ".join(sent))
        texts_out.append([token.lemma_ for token in doc if token.pos_ in allowed_postags])
    return texts_out
```

Summary of data for NLP task

Visualisation of data

NLP Task Implementation: (200) 35% 163w

Feature normalisation

To perform LDA Topic Modelling, a dictionary and a corpus was created. The dictionary was generated by implementing the **corpora.Dictionary()** function on the lemmatized text. The corpus was then created by generating a term document frequency matrix by employing the bag-of-words (BOW) method.

```
import gensim.corpora as corpora

# Create Dictionary

id2word = corpora.Dictionary(data_lemmatized)

# Create Corpus

texts = data_lemmatized

# Term Document Frequency

corpus = [id2word.doc2bow(text) for text in texts]
```

ML algorithm

The LDA model used for this study was trained using the **LdaModel()** function from the **Gensim** package. The study tested several different topics to represent the data including, 20, 10 and 5 topics. Given the coherence score discussed later, the study determined that seven topics were appropriate. This was implemented by using the *num_topics* argument when training the LDA model, as shown below:

```
# Build LDA model

lda_model = gensim.models.ldamodel.LdaModel(corpus=corpus,

                                            id2word=id2word,
                                            num_topics=7,

                                            random_state=100,

                                            update_every=1,

                                            chunksize=100,

                                            passes=10,

                                            alpha='auto',

                                            per_word_topics=True)
```

Hyper-parameters

The following hyper-parameters were optimised when training the LDA model:

- *Chunksize* refers to the number of documents used in each training chunk.
- *Passes* refer to the total number of training passes.
- *Alpha* determines the sparsity of the topics.
- *Update_every* specifies how often the model parameters should be updated.

NLP Task Output: (200) 30% 306w

Quality metrics

The model Perplexity and model Coherence Score was computed using the following code:

```
# Compute Perplexity

print('\nPerplexity: ', lda_model.log_perplexity(corpus))

# Compute Coherence Score

coherence_model_lda = CoherenceModel(model=lda_model, texts=data_lemmatized,
dictionary=id2word, coherence='c_v')

coherence_lda = coherence_model_lda.get_coherence()
```

The model Perplexity was -7.11. Perplexity is a measurement of how well a probability model predicts test data (Fig. 1). The lower the Perplexity, the more influential the model.

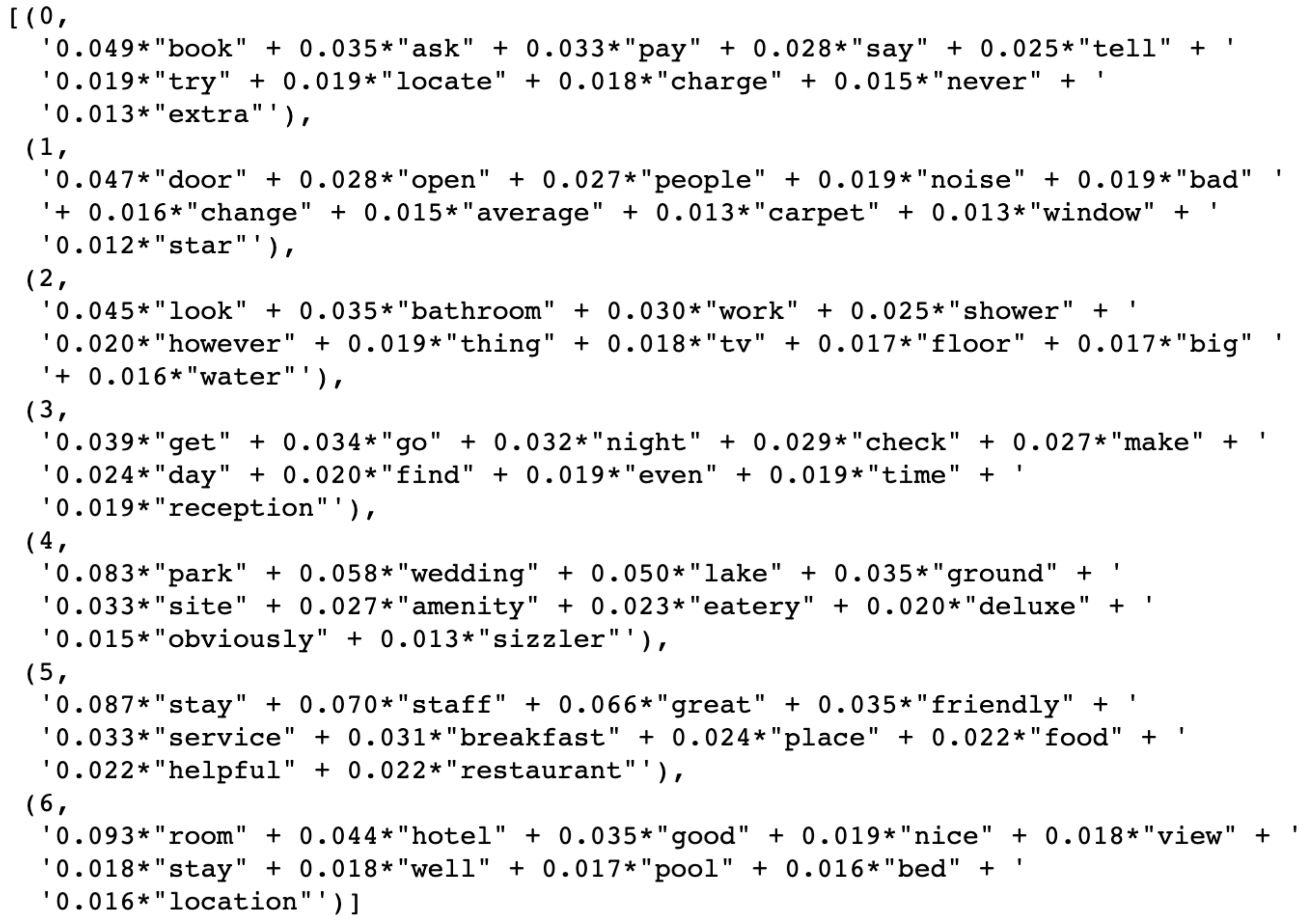
The model Coherence Score was 0.427. Topic Coherence measures score a single topic by measuring the degree of semantic similarity between high scoring words in the topic. The perplexity and coherence score give a sense of the relevance of the words categorised in each topic—the higher the Coherence Score, the better the model.

	Perplexity	**Coherence Score**
LDA model	-7.11	0.427

Fig 1. Model performance.

Summary of outputs

Below is a screenshot of the output representing the words in each topic and their relative weight (Fig. 2).



```
[(0,
  '0.049*"book" + 0.035*"ask" + 0.033*"pay" + 0.028*"say" + 0.025*"tell" + '
  '0.019*"try" + 0.019*"locate" + 0.018*"charge" + 0.015*"never" + '
  '0.013*"extra"'),
 (1,
  '0.047*"door" + 0.028*"open" + 0.027*"people" + 0.019*"noise" + 0.019*"bad" '
  '+ 0.016*"change" + 0.015*"average" + 0.013*"carpet" + 0.013*"window" + '
  '0.012*"star"'),
 (2,
  '0.045*"look" + 0.035*"bathroom" + 0.030*"work" + 0.025*"shower" + '
  '0.020*"however" + 0.019*"thing" + 0.018*"tv" + 0.017*"floor" + 0.017*"big" '
  '+ 0.016*"water"'),
 (3,
  '0.039*"get" + 0.034*"go" + 0.032*"night" + 0.029*"check" + 0.027*"make" + '
  '0.024*"day" + 0.020*"find" + 0.019*"even" + 0.019*"time" + '
  '0.019*"reception"'),
 (4,
  '0.083*"park" + 0.058*"wedding" + 0.050*"lake" + 0.035*"ground" + '
  '0.033*"site" + 0.027*"amenity" + 0.023*"eatery" + 0.020*"deluxe" + '
  '0.015*"obviously" + 0.013*"sizzler"'),
 (5,
  '0.087*"stay" + 0.070*"staff" + 0.066*"great" + 0.035*"friendly" + '
  '0.033*"service" + 0.031*"breakfast" + 0.024*"place" + 0.022*"food" + '
  '0.022*"helpful" + 0.022*"restaurant"'),
 (6,
  '0.093*"room" + 0.044*"hotel" + 0.035*"good" + 0.019*"nice" + 0.018*"view" + '
  '0.018*"stay" + 0.018*"well" + 0.017*"pool" + 0.016*"bed" + '
  '0.016*"location"')]
```

Fig 2. Screenshot of the ten keywords and their weighting for each topic.

The seven topics that were extracted via the LDA model are summarised in Table 1. The keywords were determined by those words with the highest beta value within the topic. The keywords with the highest relative probability of belonging to the given topic and were organised in order from the most important to least important keyword per topic. As LDA is an admixture model, the same words can belong to more than one topic.

Topic themes were derived from the keywords identified following the LDA Topic Modelling process. These themes are assumed are

Topic Assumption	**Relevant keywords**
Hotel booking process	Book, ask, pay, say, tell, try, locate, charge, never, extra.
Ambience	Door, open, people, noise, bad, change, average, carpet, window, star.
Room Features	Look, bathroom, work, shower, however, thing, tv, floor, big, water.
Reception	Get, go, night, check, make, day, find, even, time, reception.
Neighbourhood	Park, wedding, lake, ground, site, amenity, eatery, deluxe, obviously sizzler.
Staff Professionalism	Stay, staff, great, friendly, service, breakfast, place, food, helpful, restaurant.
Facilities	Room, hotel, good, nice, view, stay, well, pool, bed, location.

Table 1. Assumed topic themes and their keywords.

Service-related topics include the hotel booking process, reception, staff professionalism. The topic of hotel booking process includes the tasks associated with the booking process such as booking the accommodation (e.g., ‘book’, ‘ask’, ‘say’, ‘tell’), and payment (e.g., ‘pay’, ‘charge’, ‘extra’). The reception topic encompasses guest services such as luggage assistance (e.g., ‘get’, ‘go’, ‘check’, ‘day’, ‘night’) and booking services (e.g., ‘make’, ‘find’, ‘time’). The staff professionalism topic emphasizes the emotional warmth of the experience (e.g., ‘friendly’, ‘service’, ‘helpful’) and hospitality (e.g., ‘breakfast’, ‘food’, ‘restaurant’).

Topics related to the physical condition of the accommodation include the facilities and ambience. The topic dubbed facilities relate to the accommodation facilities (e.g., 'pool') and the view of the surrounding area (e.g., 'view', 'location'). The ambience topic mainly encompasses two distinct ambient qualities of the room, including sound (e.g., 'people', 'noise') and airflow (e.g., 'open', 'door', 'window', 'carpet').

Topics related to the physical condition of the room include the room features. The room features topic relates specifically to room amenities (e.g., ‘bathroom’, ‘shower’, ‘floor’, ‘water’) and dimensions/style (e.g., ‘look’, ‘floor’, ‘big’).

Location-related topics include the surrounding neighbourhood. The neighbourhood topic encompasses the local attractions (e.g., ‘park’, ‘wedding’, ‘lake’, ‘eatery’).

Visualise outputs

An Intertopic Distance Map (via multidimensional scaling) was used to interactively visualise the outputs of the Topic Model (Fig. 3). This was achieved using the pyLDAvis package and the following code:

```
# Visualise the topics

pyLDAvis.enable_notebook()

vis = pyLDAvis.gensim_models.prepare(lda_model, corpus, id2word)
```

From our topic model, we were able to obtain top keywords from each topic:

1. 39.7% of tokens about the hotel facilities such as view, pool, and location.
2. 19.8% of tokens about staff professionalism such as friendly, service, and helpful.
3. 15.6% of tokens about the hotel reception such as night, check-in, and reception.
4. 10.1% of tokens about room features such as bathroom, shower and tv.
5. 7% of tokens about hotel booking process such as paying and booking accommodation.
6. 5.6% of tokens about accommodation ambience such as noise and décor.
7. 2.2% of tokens about the neighbourhood such as parks, weddings, lakes, and eateries.

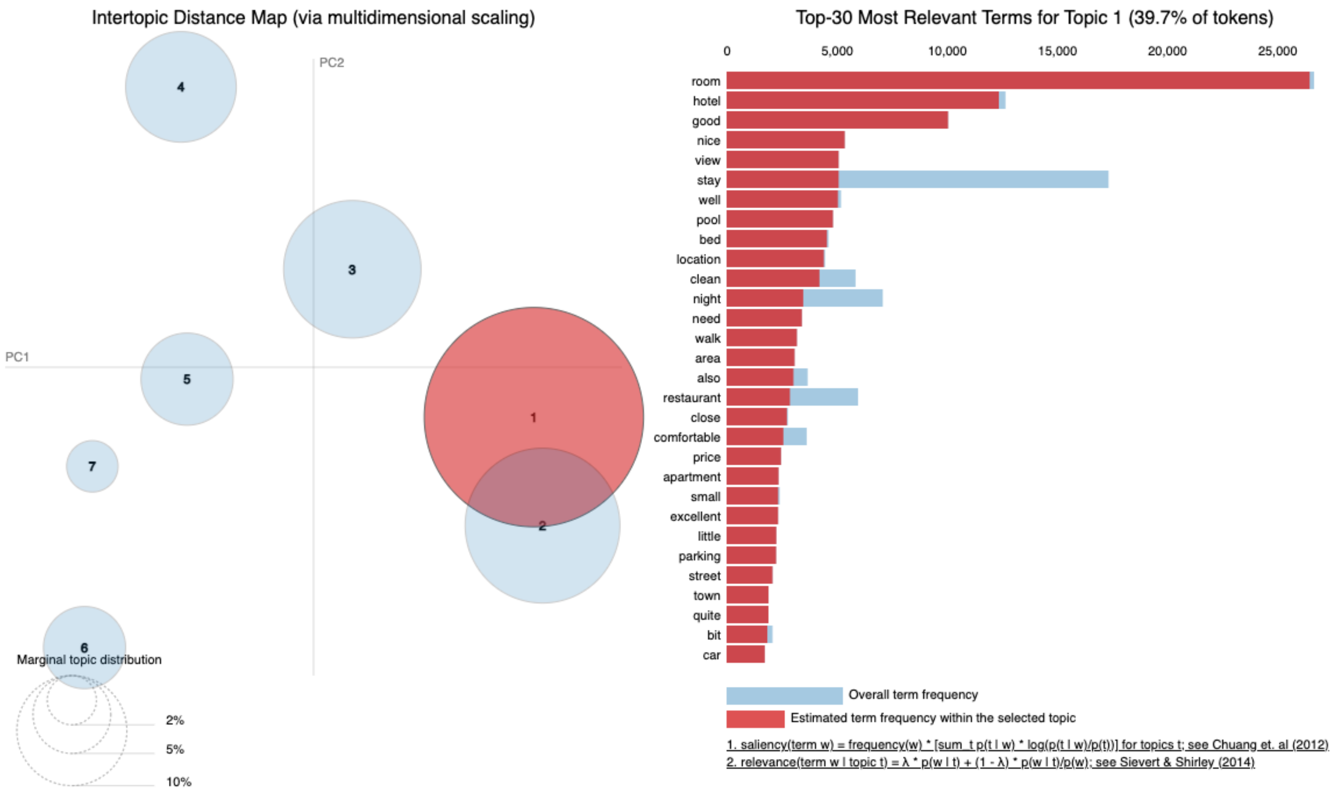


Fig 3. Screenshot of interactive visualisation representing the seven modelled topics and their associated words.

NLP task alignment with issue

Topic Modelling provides an inductive, data-driven approach that validates and extends current theory regarding the dimensions that affect customers in the tourism and hospitality industry (Sutherland, Sim, Lee, & Byun, 2020). This is done by utilising a large amount of text data from the customers. LDA extends the theory by offering more precise distinctions between the dimensions. This study extracted seven valuable topics in the customer reviews. The three topics outlining the most text include hotel facilities, staff professionalism and reception service.

By performing Topic Modelling, hotels can better understand the topic themes that customers discuss in their reviews left on TripAdvisor. For hotels to improve their customer experience, they could look at how they approach their facilities, how their staff interact with the customers and their reception processes.