Aspect-Based Sentiment Analysis Methods for Analyzing Airline Passenger Reviews

Student : Jade Arpaliangeas

Company : Airbus

Supervised by : Stéphane Malkawi

AIRBUS

Dates of thesis : 15/04/2024 – 02/09/2024

Acknowledgments

I thank Stéphane Malkawi for his involvment and supervision during the 5 months of this master thesis, and for his help in the testing data labeling process.

References

Aspect-Based Sentiment Analysis Methods for Analyzing Airline Passenger Reviews

April 2024 – September 2024

Abstract

This study aims to conduct Aspect-Based Sentiment Analysis (ABSA) on airline passengers' reviews to detect and assess the sentiment polarity associated with various service aspects, such as food and beverages quality, staff sympathy, inflight entertainment, and seat comfort. We scraped over 110,000 reviews from the Skytrax website, focusing on major airlines worldwide. Our methodology involves a two-step approach: Aspect Extraction and Aspect Sentiment Classification. Initially, we tested several models, including LDA, BERTopic, and Lemma Keywords Detection, for aspect extraction, ultimately selecting the Lemma Keywords Detection method as the basis for sentiment analysis. In the sentiment classification phase, both supervised (BERT-based models) and unsupervised models (e.g., VADER) were evaluated. While the baseline model performed well across the dataset, supervised models showed significant improvements, especially on reviews with mixed sentiment (bipolar data). Our final models were applied to assess customer satisfaction across five key service aspects for major US airlines, demonstrating the potential of our approach to provide actionable insights into airline service quality. Despite some limitations in the aspect extraction process, the study highlights the effectiveness of the developed pipeline in performing ABSA on airline reviews.

Introduction

The objective of this study is to compile airline passergers reviews about the quality of the provided service for major airlines worldwide and to build a machine learning model that would be able to detect the aspects of service that are mentioned in the review, such as quality of food and beverages, sympathy from both airport staff and flight attendants, diversity of inflight entertainment, comfort of seat and aircraft… Second, the polarity of the sentiment of the reviewer about the detected aspects in the review should be predicted, meaning that we aim to assess the satisfaction of the reviewer about all upcited aspects. The application of this study would be to be able to draw conclusions about the customer satisfaction concerning different airlines, and more specifically to be able to assess the pitfalls of the airlines in terms of service analysing reviews extracted from various source such as Tripadvisor, Reddit, or Social Medias such as Twitter for instance. This work implements Natural Language Processing methods and is part of a field called Aspect-Based Sentiment Analysis, a branch of Sentiment Analysis.

To do so, we code a scraper for the Skytrax review website, that is specialised in airlines critics. We extract and process over 110 000 reviews in english that serve as a basis for our study. We split the study in 2 sub-tasks. We first aim to build an Aspect Extraction model that predicts which aspects of a given list are cited in a review. In a second time, we seek to perform Aspect Sentiment Classification, that is to uncover the polarity (positive or negative) of the comment about each detected aspect of service in the review.

The detailed outline of our study is the following : After scraping, we clean, process, explore and describe the data to better understand how to best leverage it to achieve our goal. We then explore the scientific literature about the ABSA field in order to rely on previous findings to decide of the models we should test and implement in our use-case. The testing phase is separated between Aspect Extraction and Aspect Sentiment Classification, as stated before. Both supervised and unsupervised models are tested. The results are evaluated according to various relevant metrics such as F1-score to better depict the performances of the models. Finally, supervised and unsupervised methods are compared in an application case where we try to assess the global satisfaction about a few aspects among customers of major US airlines, while the results are also compared to the real overall satisfaction computed from the Skytrax stars ratings for each considered aspect.

Data presentation

1. Data Scraping and Variables Description

In order to train an ABSA model that is adapted to airline passenger reviews, we first need to collect relevant data. No available and up-to-date dataset was found, so we decided to scrap airline passenger reviews online. Three websites were considered: Trustpilot, Tripadvisor and Skytrax. Trustpilot provides a big quantity of reviews but users rarely specifically rate all aspects of service. At first, a scraper was developed for Tripadvisor as its data is the most complete, but as Tripadvisor makes scraping purposely difficult, the process was very long and tedious so we eventually decided to use the Skytrax data. Solely the reviews concerning the biggest airlines were scraped. In total, 179 airlines are represented by 110 156 observations. An observation consists of an ensemble of variables describing the flight and the passenger (Origin, Destination, Aircraft, Date of travel, Seat Type, Type of Traveller…) and its appreciation of the service provided. This appreciation is expressed by an overall rating (ranging from 1 to 10), and by a list of aspect ratings with grades varying between 1 and 5: Seat Comfort, Cabin Staff Service, Food & Beverages, Ground Service, Inflight Entertainment, Wifi & Connectivity, Value For Money. Besides, a textual review allows reviewers to express their exact feelings and personal experience about their flight.

However, we are not only interested in these fixed categories but would also like to assess the performance of the models we will build on chosen aspects (such as punctuality for instance). Conversely, we are not that much interested in the Wifi & Connectivity aspect as we consider it is already part of the more general Inflight Entertainment variable.

2. Data exploration

In table 1, we display the number of missing observations per variable. We remark that not all reviewers rate all aspects, and that some aspects of service are more often graded than others. For instance, Seat Comfort and Cabin Staff Service (=’crew flight’) have only 9 094 missing values against 80 871 for Wifi & Connectivity. Eventually, only 26 256 observations are fully rated - meaning all aspects are graded (23.9% of the total database).

Variable	Number of Missing Values
Airline Name	0
Overall_Rating	0
Review_Title	0
Review Date	0
Verified	0
Review	0
Aircraft	77265
Type Of Traveller	27069
Seat Type	1505
Route	27546
Date Flown	27185
Seat Comfort	9094
Cabin Staff Service	9418
Food & Beverages	26616
Ground Service	29939
Inflight Entertainment	39719
Wifi & Connectivity	80871
Value For Money	443
Recommended	0
origin	27546
destination	28750

Table 1 : number of missing observations per variable

If we have a look at the sentiment distribution for skytrax aspects in table 2, we remark that not all aspects have the same proportion of complaints (we exclude missing ratings). For instance, Cabin Staff/Crew Flight have the higher percentage of happy customers (44%), whereas Wifi & Connectivity has the lowest satisfaction rate (followed by Ground Services). No aspect has more positive than negative ratings.

Polarity	**Seat Comfort**	**Crew Flight**	**Food and Beverages**	**On Ground Services**	**Entertainment**
NEG (%)	64.27	56.30	63.84	69.15	64.28
POS (%)	35.73	43.70	36.16	30.85	35.72

Table 2 : sentiment distribution for skytrax aspects

Furthermore, we wondered if reviewers tend to rate all aspects with the same polarity. If this was the case, an Aspect-Based Sentiment Analysis (ABSA) model would be superfluous as predicting the overall sentiment with classical Sentiment Analysis would suffice to describe a review. To assess this, we compute the correlations between the 'Overall Rating' variable and all aspects (table 3). We remark that whereas the correlation is fortunately positive and quite high (~0.5), it is not close to 1. So we can conclude that the Overall Rating is not sufficient as the sentiment expressed towards an aspect can differ from the overall sentiment of the review. Therefore, ABSA should be used to differentiate each aspect's sentiment polarity.

Aspect	**Correlation with Overall Rating**
Seat Comfort	0,50
Cabin Staff Service	0,52
Food and Beverages	0,48
Ground Service	0,53
Inflight Entertainment	0,45
Wifi and Connectivity	0,39
Value For Money	0,60

Table 3 : correlations between the 'Overall Rating' variable and all aspects

In order to give us insights about the rating gap between aspects, we compute for each review and each aspect, a 'bipolarity' distance defined as the difference between the aspect rating and the mean of all available ratings.

$$bipolarity\ measure(aspect) = rating(aspect) - \frac{1}{\#K}\sum_{i \in K} ratings(i),$$

$$where\ K\ is\ the\ ensemble\ of\ rated\ aspects\ for\ a\ review$$

For each aspect, we then display the distribution of the distances. We obtain that the 90th percentiles are the following:

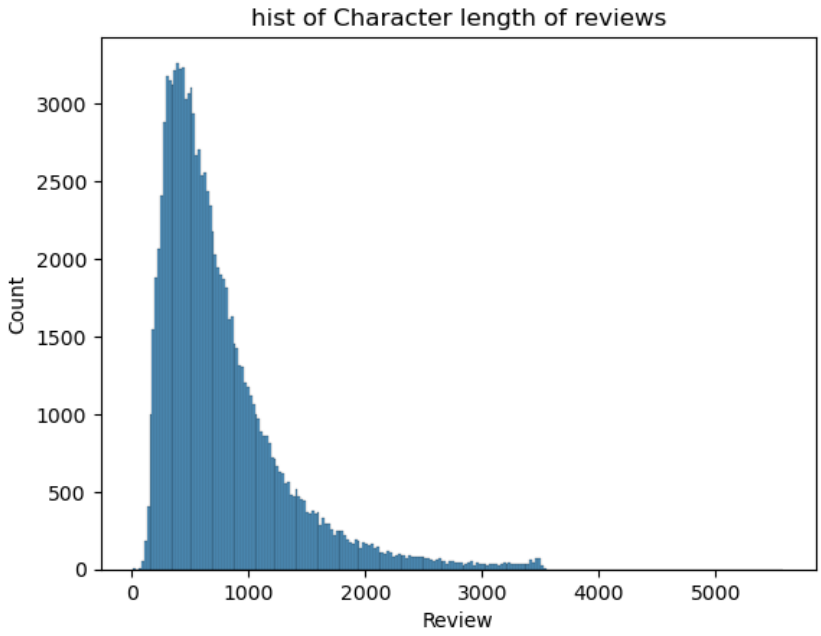
	seat comfort	crew flight	food and beverages	on ground services	entertainment
Distance	1.20	1.43	1.29	1.33	1.6

Table 3bis : threshold of distance for a review to be considered as bipolar for an aspect

We consider all reviews with a distance above these thresholds as 'bipolar' for the concerned aspect. This means that the polarity for the specific aspect is likely to be different from the rest of aspects' polarities, making the Aspect Sentiment Classification more tricky for the model. We will evaluate some models on both bipolar and random/classical review in which the bipolar reviews are not oversampled. Note: A review can be considered bipolar for one or some aspect(s) and not for others. We note that 63% of the reviews displays a unique polarity for all aspects (uniquely 1/2/3 or 4/5).

3. Data cleaning

The text of the reviews was cleaned to retain only alphanumeric and punctuation characters. The histogram 4 shows the distribution of the length of the reviews in numbers of characters. We remark that only a tiny minority of the reviews have a length over 3000 characters, which could induce truncated reviews during the modelisation process. We do not consider this as an issue and do not exclude those reviews as the information they contain in the first 2 or 3 thousand characters may suffice to convey the reviewer's opinion. The final dataset contains 109 936 observations after erasing duplicates. Further preprocessing such as lemmatization (see literature part) or stop words removing will be done later on. However, this data preparation is only relevant for a few models and will be further explained in the concerned parts of this study.



Histogram 4 : distribution of the length of the reviews in numbers of characters

4. Test set description

In order to evaluate the efficiency of the models on important aspects, we decide to label manually a few hundred reviews, by reading them and allotting the labels positive, negative, neutral or none/Nan for some interesting aspects. 304 reviews were labeled by hand, selected among fully-rated reviews. However, not all aspects are dealt with in a review, so less than 304 examples are present for each aspect alone. As a result, this test set will only be used to evaluate the Aspect Extraction part of this study (for which we do not have labeled data), but the skytrax Ratings will be preferred for the assessement of the Aspect Sentiment Classification models.

The table 5 displays the distributions of the labels per aspect in the test set.

Aspect	**NEG (%)**	**POS (%)**
Seat Comfort	57.4	42.6
Entertainment	59.8	40.2
Food and Beverages	60.2	39.8
Crew Flight	48.2	51.8
Delay	79.4	20.6
On Ground Services	76.4	23.6

Table 5 : distributions of the labels per aspect in the test set, percentages, excluding missing data

The labeled observations were not randomly drawn for the data, but were selected as to oversample positive and bipolar reviews. That is to be able to better assess the models on this

type of data. As we will see in the next section, the baseline model has good performance in the case where all aspects have the same sentiment polarity, but not in the case of a bipolar review. As this is something we wish to improve in our models, we chose to oversample bipolar reviews (outputs of Aspect Extraction are used as input to Sentiment Classification).

The probability distribution of the reviews within the labeled set is: 40% of reviews were sampled at random, 10% was specifically selected from the positive pool of reviews, and 50% are specifically bipolar.

The table 5 is therefore only representative of the distribution of sentiments in the labeled set, not in the overall data (table 2 above for this purpose).

In order to justify the use of skytrax ratings in the Sentiment Claissification model as a proxy for reviewer's sentiment, we need to ensure that the link is strong enough for all aspects. The percentage of agreement (=accuracy) between hand-made labels and the skytrax ratings for important aspects is displayed in table 6. It seems that the skytrax and hand-made ratings coincide, hence we can deduce that the text review comments are in line with the associated rating. Note that we considered that a 1-3 stars rating was negative and 4-5 stars was positive. Indeed, even if a 3 out 5 rather implies neutrality, we chose to define positivity in opposition to non-positivity (emcompassing both neutral and negatives) to simplify our problem and make it a binary classification task. However, we could challenge this classification choice.

Aspect	**Accuracy**
Seat Comfort	88.3
Entertainment	83.3
Food and Beverages	88.5
Crew Flight	96.9
On Ground Services	90.9

Table 6 : accuracy score between hand-made labels and the skytrax ratings

To avoid data leakage, this test set was drawn from a larger control set comprising 9936 observations. The 100 000 remaining observations were allotted to the training set.

Literature

In order to familiarize ourselves with Natural Language Processing, we describe and explain some important and recurrent concepts of this field in the following section.

1. General NLP concepts and reference papers

- Tokenization

Tokenization is the process of breaking down text into smaller units called tokens. These tokens can be words, subwords, or characters, depending on the granularity required. Tokenization is a crucial preprocessing step in NLP, as models typically operate on tokens rather than raw text.

- Lemmatization/stemmatization

Lemmatization reduces words to their base or root form (lemma), ensuring that all forms of a word are treated as one. For example, "running" becomes "run."

Stemming is similar to lemmatization but is more aggressive. It reduces words to their root form by stripping affixes. For example, "running," "runner," and "runs" might all be reduced to "run."

- Embeddings

Embeddings are dense vector representations of words or tokens in a continuous vector space, where similar words are positioned close to each other. Unlike traditional sparse representations like one-hot encoding, embeddings capture semantic relationships between words. Word2Vec [1] and GloVe are famous word embedders. BERT embeddings are also frequently used as they not only capture word meanings but also adjust their representation based on the context in which the word appears (see below).

- Transformers

The Transformer, a neural network architecture introduced by Vaswani et al. (2017) [2], allows models to process text in parallel rather than sequentially, which was a limitation of previous RNN-based models. Transformers use self-attention mechanisms to capture relationships between all words in a sentence simultaneously. Self-attention allows the model to weight the importance of each word in the sentence relative to every other word, effectively capturing long-range dependencies and contextual relationships within the text. This architecture not only improves computational efficiency but also enhances the ability to model complex linguistic structures.

- BERT (Bidirectional Encoder Representations from Transformers)

BERT is a transformer-based model introduced by Devlin et al. (2018) [3]. BERT is pre-trained on large text corpora (BooksCorpus dataset that contains approximately 800 million words and english Wikipedia ~2.5 billion words). This model regards tokens in a bidirectional manner, meaning it looks at both the left and right context of a token in the sentence simultaneously. 2 sub-tasks are implemented to trained the model : Masked Language Modeling (MLM), where random words are masked and the model learns to predict them, and Next Sentence Prediction (NSP), where the model predicts whether two sentences appear sequentially in the text. After pre-training, BERT can be fine-tuned on specific downstream tasks such as question answering, sentiment analysis, or named entity recognition, with minimal task-specific architecture modifications. BERT comes in two main versions : BERT Base (110 million parameters, 768 token size, 12 Transformers layers with 12 attention Heads) and BERT Large with a deeper and larger architecture.

2. Research papers on Aspect-Based Sentiment Analysis

Aspect-based sentiment analysis (ABSA) is a field of Natural Language Processing that focuses on determining the sentiment polarity associated with a specific aspect of the data. Conversely to the classical Sentiment Analysis - that aims at uncovering the overall polarity of a text, aspect-based sentiment analysis strives to classify the emotions displayed for a set of various features/aspects. Aspects could represent anything a person could have an opinion about (food, people, politics, sport team…). In the example of the airline passenger reviews, the different facets of the airlines' service are the target aspects. While sentiment analysis only concentrates on classifying the sentiment polarity of a text, ABSA models first need to pinpoint the aspects mentioned in the text, before trying to find its associated polarity. This first part can be called Aspect (Term) Extraction/ Aspect Identification. Second, a polarity has to be assigned to each detected aspect in the corpus. This step is called Aspect Sentiment Classification. Some ABSA models are the combination of two independent sub-models for the two tasks, and others have a two-in-one architecture that enables them to either perform Aspect Identification and Aspect Sentiment Classification consecutively, or both at the same time.

- Aspect Extraction/Identification (AE)

We will first describe some methods that permit Aspect Identification.

Let's use the following sentences as an example: "The food is so good and so popular that waiting can really be a nightmare."

Some models aim to extract the opinion target term (Opinion Target Extraction). In our example, the opinion terms to spot would be 'food' and 'waiting', as they are the terms the user expresses an opinion about. The opinion terms would later need to be linked to the aspects 'food' and 'service' of a restaurant. On the other hand, some methods directly go for the aspect

in the text, therefore the aspects ‘food’ and ‘service’ would be recognized straight away. This task is referred to as Aspect Category Detection (ACD).

In order to perform Aspect Extraction, the recent literature has been mostly focused on supervised models based on neural networks, as stated in the paper of Linan Zhu, Minhao Xu, Yinwei Bao, Yifei Xu and Xiangjie Kong [4]. Various types of neural networks are cited for Aspect Term Extraction, such as CNN (Convolutional Neural Network), (bidirectional) RNN (Recurrent NN), or GRU (Gated Recurrent Unit). Some other supervised Machine Learning algorithms include Conditional Random Field (CRF) [5, 6] and Support Vector Machine (SVM) [7] for aspect detection.

However, these architectures assume an available labeled training dataset. There are indeed plenty of training ABSA datasets designed for the most common domains interested in aspect-based sentiment analysis. The reference dataset is SemEval (*Pontiki, Maria, et al.)* [8], that has several versions and focuses on hotels, restaurants, and technology products (phones, laptop, camera…) reviews, each in various languages (French, English, Chinese, Arabic…). However, ABSA models are very domain-specific and a model trained on restaurant data will perform quite poorly on laptop reviews. Indeed, the link between word/terms and their underlying aspect has to be learned on specific data, and cannot be easily transferred from one domain to another. To provide a quick intuition for this, let’s consider the fact that some adjectives could be considered as positive or negative depending on the context. For instance, the adjectives ‘small’ and ‘light’ are praising when describing an electrical device as they highlight its portability, but are quite pejorative in the context of a restaurant: a ‘small’ or ‘light’ portion conveys a negative sentiment about the food portions.

Thus, some researchers developed unsupervised Aspect Term Extraction (ATE) models that could be applied to any document corpus, for the domains for which a dedicated labeled dataset does not exist. Most of those models are based on techniques such as topic modeling. A reference procedure is Latent Dirichlet Allocation and its variants (such as Hierarchical LDA). Other topic modeling methods include Top2Vec [9], Biterm Topic Model (BTM) [10] and BERTopic [11]. The details of the functionning of those algorithms will be described in the part concerning their implementation.

- Aspect Sentiment Classification (ASC)

ASC models strive to uncover the link between aspect and context to find the sentiment polarity associated to the aspect. Many previous works exploit the Long-Short Term Memory (LSTM) neural networks architecture, with Attention Mechanisms, as in [12]. Bilateral LSTM is also implemented by Ruder S, Ghaffari P and Breslin JG in [13]. Moreover, most studies cited in the review paper of Zhu, L et al. About Deep Learning Methods for ABSA [14] found that binary classification accuracy is higher than ternary classification accuracy, because it seems difficult for some neural networks to distinguish neutral sentiment. Some papers on ASC also

focus on Attention mechanisms, especially using BERT-based models. We rather deal with this category of models in the Multi-task ABSA section.

- Multi-task ABSA (Aspect Extraction and Aspect Sentiment Classification at the same time)

In [15], Yung-Chun Chang, Chih-Hao Ku and, Duy-Duc Le Nguyen attempted to apply Aspect-Based Sentiment Anaylis to the Airline Industry.

They used passengers' flight reviews posted on TripAdvisor from January 2016 to August 2020, in which reviewers expressed their sentiment about Legroom, seat comfort, in-flight entertainment, client service, value for money, cleanliness, check-in and boarding, and food and beverages. The ratings (1 for lowest appreciation, 5 for best) are exploited in association with the textual review to train a supervised deep learning model to perform ABSA. The input of the model is a preprocessed textual review, and the labels (traget output) are the aggregated ratings for each aspect (1-3 are aggregated as 'Negative', and 4-5 as 'Positive', and a missing value is interpreted as an uncited aspect in the review). Their model is designed to perform both AE and ASC at the same time (all-in-one model), therefore the output is of size 24 (8 aspects * 3 classes). To build their model, they rely on embeddings usually used as input to BERT (token and position embeddings), and on discriminative linguistic embeddings. These embeddings are then passed to a block of 12 Transformers Layers (hiden size = 768) with 12 Attention heads each, corresponding to the size of the BERT-base model, whose weights are used to initialize the model. Additionally, fully connected layers gradually reduce the layers size to 24 (output size). A softmax function is applied to generate probabilities.

However, we believe some points of their work could be improved :

- They considered the absence of rating for an aspect as an absence of mention of this specific aspect in the review. While we aknowledge a missing value could be a proxy of an omission in the review, we doubt that the link is strong enough for it to be used as a substitute for Aspect Extraction.
- Is seems that the researchers chose to remove stop words before applying the WordPiece Tokenization for generating bert embeddings. However, as BERT is based on self-attention mechanisms that leverage the context of sentences to produce meaningful contextual embeddings, stop words should be kept as they convey contextual and attention information.

The perfomance of the model is measured using recall, precision and F1-score. F1-score is the harmonic mean between recall and precision. It is not precised which class is concerned by those metrics. The displayed F1-score lies between 55% and 65% for all aspects (56.5% for food and beverages). This paper also mentions some baseline models, among which a variant of KNN, that calculates document similarity in the bag-of-words feature space with TF–IDF term weighting, that achieve worse results, as well as Random Forest method also relying on TF-IDF also yielding disapointing results. Then, we will not attempt those alternative methods.

The dataset was imbalance between positive and negative class so the cross-entropy loss function they implemented was weighted in order to compensate the over-representativity of negatives. The weights were inversely proportional to the classes frequency.

In [16], Xin Li, Lidong Bing, Wenxuan Zhang and Wai Lam also performed 2-in-1 ABSA as their model does AE and ASC at the same time. The researchers expoited the pre-trained "bert-base-uncased" model fine-tuned on their data, and tested a myriad of architectures between BERT and the classifier layer, among which linear (dense) layers, Gated Recurrent Unit, Self Attention layers, and Conditionnal Random Field (CRF). They found that their BERT-based models were exceptionally robust to overfitting.

In [17], Zhuang Liu, Wayne Lin, Ya Shi, Jun Zhao advise to post-train BERT on domain-specific data before fine-tuning it for downstream task such as ABSA. This paper, along with, [18] state that BERT variants such as ROBERTA and DEBERTA may sometimes be better suited for ABSA.

- Interest of our work

Unfortunately, no labeled dataset is available for airline passengers reviews, and we were unable to find a model that was specifically trained on this kind of data (one or two were found on Hugging face but with no detail about how to use them nor about their architecture/training process). As a result, our goal is to construct a dataset with airline passengers reviews in order to train an adapted ABSA model.

ABSA Models

1. Aspect Extraction Models

In order to perform Aspect-Based Sentiment Analysis on our data, we decide to use a pipeline of 2 models. The first model will need to be an Aspect Extraction model, to enable the detection of mentioned aspects in the review. We cannot consider that a rated aspect implies a mention in the review (as done in [15]), so the skytrax ratings cannot serve the Aspect Extraction step. To confirm this, we can have a look at table 7 showing the number of mentioned and not mentioned aspects in the fully-rated reviews of a test set (hand-labeled reviews + a few reviews labeled using the gpt4 model). We conclude that in our hand-made test set, a rating for an aspect does not always cause a related comment in the review.

	Seat Comfort	**Crew Flight**	**Food and Beverages**	**On Ground Services**	**Entertainment**
Mentioned aspect	235	421	269	348	169
Not mentioned aspect	441	255	407	328	507

Table 7 : number of mentioned and not mentioned aspects in the fully-rated reviews of the test set.

As we do not have a labeled training dataset, the Aspect Extraction procedure should be unsupervised. We decide to do topic modeling to perform Aspect Extraction.

- Latent Dirichlet Allocation

The first algorithm we implement is Latent Dirichlet Allocation (LDA) [19]. LDA assumes that each document is a mixture of a small given number of topics, and each topic is characterized by a distribution of words. Given the documents, the task is to uncover the topics (a set of words) and their distribution within the documents. This process involves statistical methods like Gibbs sampling or variational inference to estimate the distributions. LDA relies on the Dirichlet distribution described below :

The Dirichlet distribution is defined for a vector of non-negative random variables $\theta = (\theta_1, \theta_2, \dots, \theta_K)$ that sum to 1 (i.e., they represent probabilities). The distribution is parameterized by a vector of positive real numbers $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_K)$

The probability density function of a dirichlet distribution is defined as :

$$p(\theta \mid \alpha) = \frac{1}{B(\alpha)} \prod_{i=1}^{K} \theta_i^{\alpha_i - 1} \quad \text{where} \quad B(\alpha) = \frac{\prod_{i=1}^{K} \Gamma(\alpha_i)}{\Gamma\left(\sum_{i=1}^{K} \alpha_i\right)} \quad ,$$

where $\Gamma(n) = \int_0^\infty t^{n-1} e^{-t}\, dt$ for a positive real number n.

For a document *d*, let θ_d represent the vector of topic proportions (i.e., how much each topic is represented in the document). The distribution of θ_d is modeled as: $\theta_d \sim$Dir(**α**). dim(α) = nb of topics. For a topic *t*, let Φ_t represent the word distribution (i.e., the probability of each word occurring in that topic). The distribution of Φ_t is modeled as: $\Phi_t \sim$Dir(**β**) where dim(β) = nb of words.

Parameter **α** controls the mixture of topics in any given document. Higher values suggest that documents are likely to contain a mixture of most of the topics, and not just one or two topics. Parameter **β** controls the distribution of words in a topic. Higher values mean that each topic is likely to contain a mixture of most words, making the topics less distinct.

The underneath example representation (Figure 8) of topic representative keywords generated by LDA with 10 topics shows that even when given enough data as input (4377 sentences=documents) and allowing 5 to 20 topics to be formed, some topics are irrelavant or still mix disjoint aspects of the service. For instance, the topic 9 displays the keywords 'seats' and 'staff' that point to different aspects. Plus, topic 1 has both 'luggage' and 'food' as keywords. Varying the α and β parameters did not improve the resulting topics. We chose to abandon this approach to implement a more advanced topic modeling method : BERTopic.

```
Topic 1:
['luggage', 'flight', 'hours', 'did', 'food', 'good', 'told', 'plane', 'airport', 'people']
Topic 2:
['staff', 'flight', 'told', 'service', 'airline', 'customer', 'just', 'plane', 'london', 'terrible']
Topic 3:
['flight', 'time', 'fly', 'hours', 'plane', 'delayed', 'flights', 'pay', 'booked', 'day']
Topic 4:
['service', 'food', 'good', 'customer', 'experience', 'flight', 'airline', 'worst', 'cabin', 'poor']
Topic 5:
['flight', 'airlines', 'crew', 'plane', 'better', 'cabin', 'airline', 'passengers', 'service', 'fly']
Topic 6:
['check', 'airport', 'crew', 'cabin', 'got', 'nice', 'boarding', 'process', 'old', 'staff']
Topic 7:
['flight', 'seat', 'seats', 'comfortable', 'aircraft', 'good', 'time', 'minutes', 'hour', 'flights']
Topic 8:
['flight', 'drinks', 'ticket', 'time', 'great', 'offered', 'food', 'pm', 'flew', 'buy']
Topic 9:
['class', 'business', 'airline', 'airlines', 'recommend', 'travel', 'staff', 'seat', 'cost', 'low']
Topic 10:
['flight', 'airline', 'crew', 'staff', 'time', 'did', 'singapore', 'airport', 'return', 'rude']
```

Figure 8 : Topic representations of LDA with 10 topics

- BERTopic

- Functioning of the algorithm

We chose to leverage the potential of the BERTopic framework designed by Maarten Grootendorst in order to perform the Aspect Extraction task on our data.

BERTopic was initially created for topic modeling within a group of documents and it is a concatenation of several independent components described below. Those steps can be interpreted as a training strategy. Indeed, the trained topic model is then able to assign new documents to an already existing cluster. We note that inferring the belonging of a document to a cluster/topic does not induce the redefinition of the clusters, which are not modified in order to integrate the new instance. The prediction only consists of an allotment to an existing cluster.

More about the distance used in [21].

- Embedding: that part relies on sentence-transformers/all-MiniLM-L6-v2 to produce sentence embeddings, which is a trained version of a MiniLM model (Self-Attention based model like BERT but with fewer parameters that retain most of the initial model capabilities). More details in this paper [22].

  The document is embedded as a vector to be passed to the next step of the algorithm. (see embeddings in literature part).
- Dimension reduction: depending on the embedding model used in step 1, the dimension of the embedding vectors can be between 500 and 1000, a reduction of dimensionality is performed thanks to UMAP (Uniform Manifold Approximation and Projection) [23]. It enables to keep the same distance between initial embeddings in a lower-dimensional space, in order to facilitate the upcoming clustering step.
- Clustering: As embeddings convey the semantics of a document, clustering the embeddings thanks to the cosine distance allows to gather semantically close documents into clusters. We note that some documents are not associated with a cluster and are considered as outliers (topic = -1). HDBSCAN by default (Hierarchical clustering that identifies clusters based on the density of data points. Points in high-density areas form clusters, while points in low-density areas are usually treated as noise), or k-means are used to create clusters of embedding. $\text{Cosine Distance}(A, B) = 1 - \frac{(A \cdot B)}{\|A\|\|B\|}$ where $\|.\|$ is the euclidian norm.

Now that the clusters (or topics) are defined, we need to find a description for each of them in the form of representative keywords (bag of words):

- Tokenization (CountVectorizer): This part sets the list of the potential keywords/representative expressions for a cluster based on criteria such as minimum occurrence and number of tokens allowed per ‘keyword’ (ex : ngram_range=(1,3) enables a sequence of 1 to 3 words to be considered as a keyword, as some recurrent expressions are a combination of several words)

It also allows the user to remove stopwords from the potential keywords. Indeed, those were kept during the embedding process because MiniLM relies on attention mechanisms and transformers that need context to be efficient. However, those recurring stopwords become irrelevant when looking for representative keywords for a cluster. Words such as 'the', 'as', 'my' decrease the quality of the topic representations.

- Weighting: Then, a TF-IDF (Term frequency - Inverse document frequency) is applied in order to extract the best keyword illustrations of a cluster. The potential keywords that appear the most in a cluster relative to their occurrence among other clusters are selected.
- Topic representations fine-tuning (optional): zero-shot classification. Explained later.

See [20] for more details about each step.

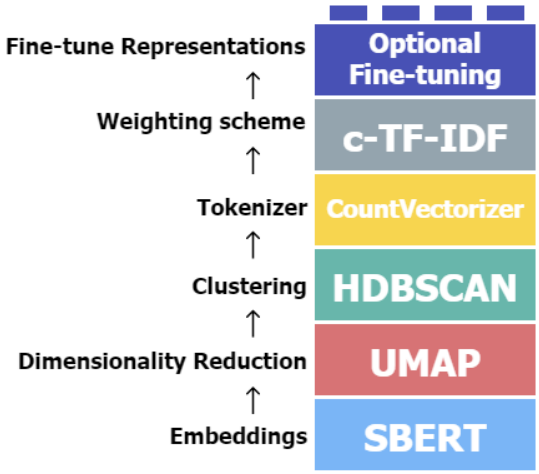


Figure 9 : BERTopic steps schema

- Topics keywords representations

To start implementing this method, we train the model on 500 documents (=reviews). The topic representations that we obtain are in figure 10.

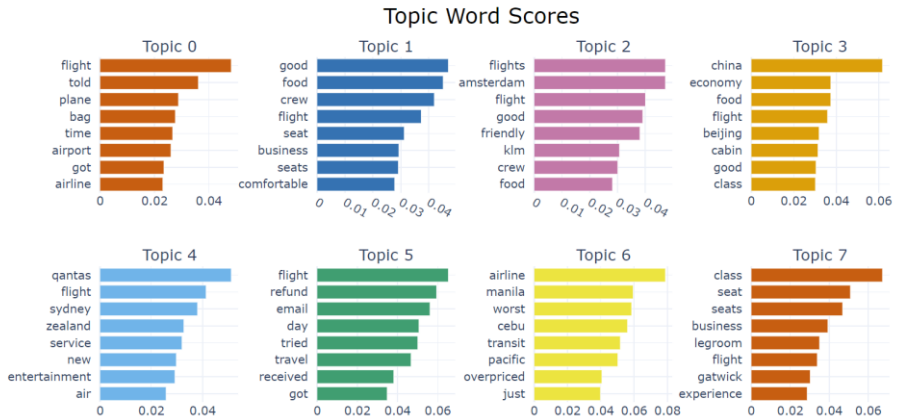


Figure 10: full reviews (not separated by sentences), keywords per topic (stopwords were removed from tokens)

We observe that the topic representations are not very informative: for example, in topic 0, the keywords ‘flight’, ‘plane’ and ‘airport’ are reported. Actually, we conjecture that a review that can be long of several hundred words is too semantically diverse to be analyzed by bertopic. We decided to split the reviews into sentences and to set the rule 1 document = 1 sentence. We use the same 500 reviews as inputs but split them by sentences. As we can see in figure 11, the topic representations are way more relevant. Indeed, topic 0 is undoubtedly associated with the ‘Seat Comfort’ aspect, topic 1 with the ‘Food and beverages’ aspect, topic 2 with ‘Entertainment’... From now on, and unless explicitly stated otherwise, we split reviews by sentences.

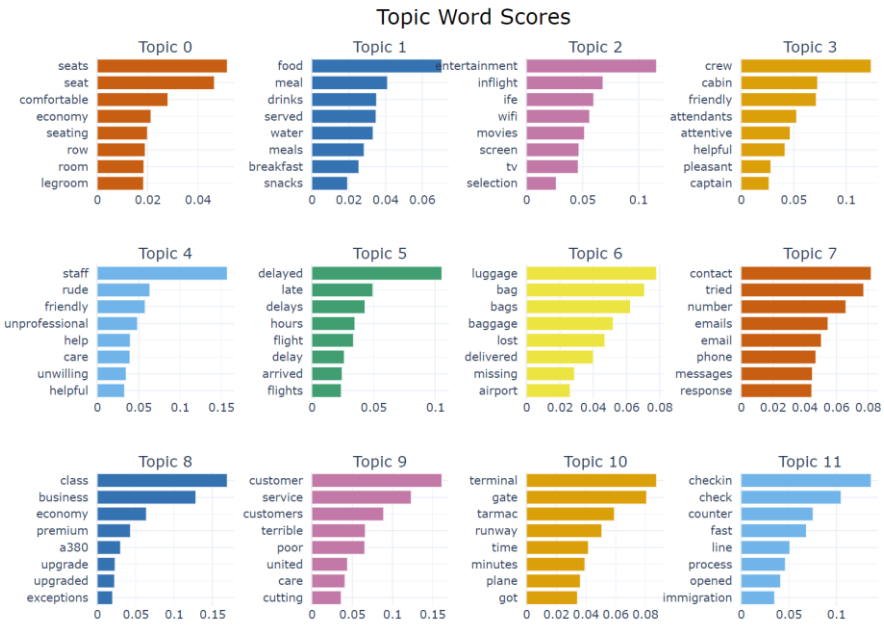


Figure 11: reviews split by sentence, keywords per topic, for the 12 biggest topics (stopwords were removed from tokens)

- Topics to aspects linkage

However, this clustering yields 62 topics (plus outliers) compared to 8 topics for the full-review version. We certainly do not have 62 specific aspects. Some aspects are therefore shared between several topics. Another challenge is then to link the topics to the corresponding aspects when relevant - some topics will not be used as they may describe uninteresting service aspects from our point of view. To tackle this issue, we introduce the zero-shot fine-tuning of topic representations (6th optional step of bertopic). This last step is used to fine-tune the topic representations. We propose aspects as the titles of the topics and a Zero Shot Text Classification model is applied to allot topics to given aspects, if the match is strong enough. The model used is facebook/bart-large-mnli from HuggingFace [24], that predicts the probability that 2 expressions/sentences could follow each other, that is have similar contexts. Therefore, we feed the model with the keywords as generated through c-TF-IDF (1st entry) and the set of candidate aspects (2nd entry). If, for a certain topic, we find a similar enough aspects, then it is assigned to it.

- How to make predictions

To make predictions about cited or uncited aspects in the reviews, we split reviews into sentences and assign each sentence to its closer topic according to the distance described in [21] (or consider it as an outlier = no relevant aspect cited). If the topics keywords representations are close enough (controled by min_prob hyperparameter) to one of the target aspects according to the facebook/bart-large-mnli model, then the sentence is assigned to this aspect. An aspect is assumed to be present is a review if at least on of its sentence was linked to this aspect. We introduce two class : 'IN' and 'NOT IN' for every pair review/aspect.

- Evaluation metrics

In order to evaluate the performance of the zero-shot classification, we concatenate all the aspects recognized in all sentences of a review and compare them to the ground truths (real labels). Our goal is firstly to avoid too many false positives, that is to avoid that an aspect is being wrongly detected in a sentence if it is not actually cited, as this could bias the upcoming sentiment classification model and diminish its performances. We believe recall of cited aspect is also important as we want to recognize as much as possible of the reviews mentioning an aspect. However, we tend to prioritize precision over recall for the 'IN' class.

In order to assess the performance of the BERTopic model, we mainly focus on accuracy, and F1-score for the 'IN' class (aspects actually present in the review), for each aspect. This F1 score is the harmonic mean between precision and recall for true 'IN' instances :

$$F1\ score = \frac{2 \times precision \times recall}{precision + recall}$$

- Topics to aspects 'fine-tuning'

Actually, not only the literal aspect names such as 'food and beverages' or 'on ground services' were passed as entry to the BART model. We tested several configurations and keywords. In the first setting, entries are words that are in the semantic field of the target aspects. For instance 'food' and 'beverages' for the aspect 'food and beverages', or 'flight attendant' and 'steward' for the onboard crew ('crew flight'). If a topic is linked to any of the entry keywords (that is BART issues a probability of at least min_prob (0.8 by default) for the pair entry 'keywords'/target aspect), then it is considered linked to the aspect. In the second setting, for comparision, all these keywords are joined together and separated by a comma in entries of the form 'food, beverages' and 'flight attendant, steward'. To see the list of used keywords, refer to table 15. Results are averaged over 6 trainings on 1000 reviews (as UMAP introduces randomness). The results between separated and joined keywords options are compared in the tables 12 and 13. We see that joining keywords with comas in a single entry link to a topic yields overall better results. Indeed, all F1 score are better (except for entertainment where is decreases from 57.7 to 41%). Combining all keywords together likely helps BART seize context more easily.

	Accuracy Score	F1 Score	Recall	Precision
Crew Flight	59.0	22.7	14.8	52.0
Entertainment	87.1	57.7	42.9	95.1
Food and Beverages	64.8	0.0	0.0	0.0
Seat Comfort	75.0	23.8	14.2	87.7

Table 12 : Bertopic results for separated keywords

	Accuracy Score	F1 Score	Recall	Precision
Crew Flight	65.2	39.2	29.8	89.2
Entertainment	83.8	41.0	26.8	94.8
Food and Beverages	89.5	85.0	85.0	85.0
Seat Comfort	82.3	56.5	40.6	95.3

Table 13 : Bertopic results for keywords joined with coma in a signle entry

- Outliers reduction techniques

BERTopic allows to implement several outliers reduction techniques, that can force the assignation of more sentences to topics. This could help enhance the recalls that are pretty low. We test the ‘c-TF-IDF’ alternative that calculates the TF-IDF representation for each outlier document and finds the best matching TF-IDF topic representation using cosine similarity. We compare this method to no outliers reduction. The metrics are displayed in table 14. When comparing to table 13, we observe that (pos) F1-scores increase (or level out) for all aspects. For example, the entertainment recall reached 47.5 against 26.8% without outliers reduction, without damaging the precision.

	Accuracy Score	F1 Score	Recall	Precision
Crew Flight	66.0	45.6	36.0	77.7
Entertainment	88.2	63.2	47.5	96.0
Food and Beverages	88.4	84.1	87.2	81.2
Seat Comfort	84.4	68.2	58.2	82.4

Table 14 : Bertopic results for keywords joined with coma, c-tf-idf outliers reduction

- Unsuccessful attempts (did not improve results) :

- Further Mini-LM training : As we had seen in the literature, Attention-Based models such as BERT may necessitate further training to be better adapted to domain specific data. As Mini-LM used for sentence embeddings is also based on transformers, we came up with the idea of further training it in order to build more relevant embeddings for airline reviews. We used PyTorch to further train the base 'all-MiniLM-L6-v2' for 5 epochs of 12 000 reviews (batch_size = 64). The result we obtain are not clearly enhanced by the training so we finally keep the initial model.
- Attempt to use BERT instead of Mini-LM
- Concatenation of sentences : As seen before, cutting the reviews into sentences highly helps with aspect detection. However, we could still improve this technique. For instance, some aspects are evoked over more than 1 sentence, and considering the sentence as an independent entity is restrictive. Ex: 'The seat was narrow. I had no place to put my water bottle'. Here, the 2nd sentence will be recognized as 'food and beverage' although both refer to 'seat comfort' in reality. We then decide that all combinations of 1 to N consecutive sentences (N=2 or 3 reasonably) would be considered as an entity and fed to bertopic for training. This approach did not clearly improve the results.

Overall, BERTopic can produce good performances if configured well but the UMAP step always introduces a lot of randomness in the algorithm and the performance is very unstable across trainings. Implementing PCA (Principal Component Analysis) in place of UMAP for the dimension reduction brick did not bring satisfactory results. We eventually look for a more deterministic procedure to do Aspect Extraction.

- Lemma Keywords Detection

As we could not obtain reliable, stable and reproducible results with BERTopic, we chose to use a lemma keywords recognition method. First, all sentences are lemmatized in order to remove conjugation and plural forms in particular, and we use a list of lemma keywords for each aspect that belong to its lexical field, and are therefore expected to be present in most sentences talking about the considered aspect. If an aspect's keyword appears in the lemmatized sentence, we consider that the associated review mentions the linked aspect. Several lists of keywords are attempted but the retained one is the following :

Aspect	**Keywords**
Food and Beverages	food, beverage, meal, snack
Entertainment	entertainment, movie, screen, wifi, headphone, TV, music, IFE, video, internet, wi-fi, connectivity
On Ground Services	check-in, lounge, counter, luggage, ground staff, ground crew, check in, gate
Crew Flight	attendant, steward, FA, crew, hostess, staff
Seat Comfort	legroom, armrest, recline, width, seat

Table 15 : list of keywords for Bertopic and Lemma Keywords Detection

The results of Aspect Extraction on this method using the test set are shown in table 16. We observe that this method is rather satisfactory and more stable than BERTopic (as it is deterministic), although it excludes all reviews that mention an aspect via words that do not belong to the list, or that mention it implicitly. Results are particularly good for 'Seat Comfort', 'Food and Beverages' and 'Entertainment', and globally better that with BERTopic (except for 'Food and Beverage' that decreases slightly, but was already very good in Bertopic). Consequently, we select Lemma Keywords Detection to perform AE on the sreview in order to extract the sentences that will be used to train/assess the Sentiment Classification models of each aspect.

	Seat Comfort	**Crew Flight**	**Food and Beverages**	**Entertainment**
Accuracy	87.5	69.0	87.5	90.1
Pos F1 score	80.8	67.5	81.7	79.2

Table 16 : Performance of Lemma Keywords Detection

2. Aspect Sentiment Classification Models

For the second part of the model, we aim at finding the polarity of opinion of the reviewer for each aspect of the review that was previously detected by the first part of the model. Our data is only labeled based on a fixed set of aspects (skytrax classification). As a result, we do not have labeled data for a self-chosen list of aspects. For instance, if we wanted to differentiate between satisfaction on 'food' and on 'beverages', or predict sentiment for new aspects (punctuality, customer service, cleanliness...), the fixed skytrax classification would not allow it. We will then build 2 separate models for this second part: one unsupervised, and the other supervised, that we will evaluate based on the skytrax ratings of the reviews, alloting the 0 label (negative) if the reviewer gave 1 to 3 stars, and positive otherwise. The unlabeled reviews are removed. Indeed, as we have seen in table 6, the ratings usually are in agreement with the expressed sentiment, justifying the use of the skytrax ratings as proper labels. As in AE, we restrict our tests to the aspects 'food and beverages', 'crew flight', 'seat comfort', 'entertainment', and 'on ground services' to demonstatre the potential interest of the methods. The sentences extracted by Lemma Keywords Detection are passed as input for ASC (ratings are outputs to predict).

A. Baseline model

In order to set a baseline for this study, we aim to find an already existing model that would serve as a benchmark to compare the models we will train. As stated in the literature, we could not find a model specifically designed for Aspect-Based Sentiment Analysis on airline passenger reviews.

As a result, we chose to focus on a general model that was trained on a large corpus comprising data from various fields. The "yangheng/deberta-v3-base-absa-v1.1" model available on Hugging Face corresponds to this need as it was trained on laptops, restaurants and yelp (review website for shopping, Food, Entertainment, Services…) reviews among others. This model is trained based on the FAST-LCF-BERT, that leverages Local Context Focus. The LCF mechanism allows the model to balance between focusing on the local context (words immediately surrounding the aspect) and the global context (the overall sentence). This model relies on microsoft/deberta-v3-base [25].

The performances of this model on the 5 aspects are displayed in table 17. First of all, we remark that the metrics are pretty high for all aspects for the overall data (randomly sampled from test set), with an accuracy almost always between 0.8 and 0.9. Macro F1 scores (non-weighted average of Pos and Neg F1 Scores) are also very satisfying, with very little variation between F1 scores for 'Positive' and 'Negative' classes. If the model performs slightly better for 'seat comfort', 'food and beverages' and 'on ground services', this could be due to an intrisic

difference in the performance of the model, but this may also be due to the fact that the F1 score of the AE task was lower on 'crew flight' and 'entertainment', suggesting more false positive than for other aspects. For those reviews, the model could struggle to detect the polarity of the specified aspect, as it is in reality not mentioned in the review, even if it was rated by the reviewer. However, this case is rare, so we make the assumption that this does not largely and significatively impact our results.

Besides, we notice that the results (pos F1 score particularly) are largely better on overall reviews than on reviews that are specifically bipolar for the considered aspect. For instance, the positives' F1 score for 'food and beverages' is 30.8% for bipolar reviews versus around 84% more generally in all data. This can be explained by the fact that the baseline model tends to predict the overall polarity and to allot it to all aspects. When the polarity associated with an aspect contrasts a lot with those of other aspects, the model struggles to discriminate the constrating aspect's sentiment. This is an area of improvement for the models to be built. We aim to improve the baseline overall, and especially on bipolar data, as this would prove the capacity of our models to differentiate between the polarities of several aspects, which is the ultimate goal of ASBA.

Baseline model		bipolar	overall
	Accuracy	0.687	0.866
	F1pos	0.578	0.887
	F1neg	0.751	0.836
	F1 macro	0.664	0.861
Seat Comfort	test size	182	500
	Accuracy	0.518	0.814
	F1pos	0.473	0.796
	F1neg	0.555	0.829
	F1 macro	0.514	0.812
Crew flight	test size	309	500
	Accuracy	0.670	0.860
	F1pos	0.308	0.839
	F1neg	0.783	0.876
	F1 macro	0.545	0.858
Food and beverages	test size	218	500
	Accuracy	0.757	0.886
	F1pos	0.316	0.807
On ground services	F1neg	0.852	0.919

	F1 macro	0.584	0.863
	test size	267	500
	Accuracy	0.529	0.792
	F1pos	0.319	0.758
	F1neg	0.640	0.818
	F1 macro	0.480	0.788
Entertainment	test size	136	500

Table 17 : Performance of baseline model

B. Unsupervised Models

We test several unsupervised models to improve our baseline.

In particular, we test VADER and SentiWordNet, two lexicon-based and rule-based models that do not necessitate further training and are already available.

- SentiWordNet

The first lexicon-based model we assess is SentiWordNet, an extension of WordNet, a lexical database where words are grouped into sets of cognitive synonyms (synsets), that share the same meaning. SentiWordNet provides 3 sentiment scores for each of those synset : one for positivity, negativity, and neutrality, summing up to 1. We can then add the score of each word in a sentence to extract its overall polarity. We implemented this method, but it did not yield good results, as it prdicted too many false Negative (unable to detect positive instances). We aknowledge that this method does not account for negations, context or amplificative adverbs for instance. As a result, we introduce another procedure that corrects the upcited drawbacks in the next section : VADER.

- VADER

VADER [26] stands for Valence Aware Dictionary and sEntiment Reasoner. In a sentence, each word has an intrinsic sentiment intensity score between +4 and -4 that is lexicon-based and context-independent. This sentiment intensity can be amplified, mitigated or reversed by:

- Punctuation: (“Good!!!” would be considered more positive than “Good”)
- Capitalization (“GOOD” is more positive than “good”)
- Degree Modifiers ("very", "quite", "extremely")
- Conjunctions (like ‘but’)
- Negations

VADER issues positive, negative, and neutral scores that indicate the magnitude/intensity of each sentiment, as well as a 'compound' score computed as compound = tanh(vsum/α), where: vsum is the sum of scores of each word in the text, α is a normalization constant (usually set to 15). The compound score lies between -1 and +1.

To apply this model to our use-case, for a specific aspect, we apply the Aspect Extraction model to predict the sentences dealing with the aspect and pass them to VADER. Initially, the prediction was set to 0 if the mean of the compound scores of all sentences associated to the aspect for a review was negative, and to 1 if it was positive. However, as the probabilities given by VADER may not have a linear relationship with the intensity of the sentiment, and as we sum up these probabilities, the 0 threshold to differentiate between positives and negatives might not be optimal. Consequently, we assessed the best threshold to maximize the Macro F1-score on control data for each aspect. The results are given in table 18.

We observe that the threshold is always positive, indicating that for example, if 2 sentences are extracted in a review for an aspect and have opposite polarities but same absolute score value (mean=0), then the aspect's polarity should still be considered as negative.

However, as VADER is supposed to be used as an unsupervised method for unrated aspects/categories potentially different from the skytrax ones, we are not supposed to be able to find the best threshold for each specific aspect (as for unrated/new no labeled data is available). Nonetheless, we assume that the upcited conclusion is general and that we can infer an approximative 'best' threshold for VADER that applies to all aspects. We select it as the mean of the 5 found thresholds for the tested aspects : 0.24.
This selected threshold improves the results for all aspects compared to threshold = 0.

	Best thresholds VADER
Seat Comfort	0.191
Crew flight	0.207
Food and beverages	0.262
On ground services	0.147
Entertainment	0.389
Mean	**0.239**

Table 18 : best vader threshold for each aspect

The table 19 describes the results of VADER, for both bipolar and random/overall reviews, with the 'optimal' threshold of 0.24. The Macro F1 score is compared to baseline (best in bold). We can clearly state that the baseline model is better overall, but that for bipolar reviews, our basic aspect extraction method followed by VADER outperforms the baseline. We remark that there are large discrepancies of performance between aspects in the bipolar case : 'Seat Comfort', 'Food and beverages' and 'On ground services' have fair associated results (F1 = 0.676, 0.681 and 0.665) but 'Crew'has a Macro F1 score of 0.552 and a mediocre accuracy of 0.553.

VADER Results vs Baseline		bipolar	overall
	Accuracy	0.705	0.764
	F1pos	0.579	0.699
	F1neg	0.773	0.806
	F1 macro	**0.676**	0.752
	test size	555	3486
Seat Comfort	F1 macro baseline	0.664	**0.861**
	Accuracy	0.553	0.763
	F1pos	0.567	0.735
	F1neg	0.538	0.786
	F1 macro	**0.552**	0.760
	test size	1279	6313
Crew flight	F1 macro baseline	0.514	**0.812**
	Accuracy	0.777	0.767
	F1pos	0.506	0.729
	F1neg	0.856	0.797
	F1 macro	**0.681**	0.763
	test size	435	3124
Food and beverages	F1 macro baseline	0.545	**0.858**
	Accuracy	0.806	0.778
	F1pos	0.448	0.585
	F1neg	0.883	0.848
	F1 macro	**0.665**	0.717
	test size	629	3288
On ground services	F1 macro baseline	0.584	**0.863**
	Accuracy	0.635	0.726
	F1pos	0.391	0.698
	F1neg	0.739	0.712
	F1 macro	**0.565**	0.723
	test size	469	2174
Entertainment	F1 macro baseline	0.480	**0.788**

Table 19 : Performance of VADER model

The achievements of VADER are pretty fair for an unsupervised method, but we would like to be able to better our metrics by leveraging the skytrax ratings as proxy for sentiment polarity for each aspect in order to train a supervised model, implementing deep learning methods.

C. Supervised Models

We decide to build specific models for different aspects (one model per aspect). We mostly design models focused on 'food and beverages' and 'crew flight' aspects.

- Input data

As we do not have a labeled dataset for aspect extraction, the input data for supervised ASC directly stems from the Aspect Extraction model. The sentences that were identified as positive for an aspect will be the fed as input to the models, and the labels will be the associated skytrax ratings for the concerned review, providing that the aspect was actually rated. Otherwise, it is dropped and not used for training.

Remark : Several sentences can be given to the model for a same review and a same aspect.

Unfortunately, we cannot make sure that we do not have false positives for an aspect, that is if the AE model wrongly stated that a review was mentioning an aspect. On the other hand, false negatives are not problematic since they just do not belong to the ASC traning data. However, if the AE model consistenty failed to detect some keywords or certain context associated with the aspect, the model could lack training on those examples and could have a lessened performance on them. This is a bias we acknowledge.

- Models Architectures

For the supervised variant of the ASC model, in view of the literature, we rely on deep learning methods.

We introduce 3 types of neural network architectures all based on the pretrained 'bert-base-uncased' model that we post-train, and from which we extract the embeddings corresponding to the ['CLS'] token, followed by a dropout layer and :

1) 1 dense hidden layer (Model1)
2) 3 dense hidden layers + 3 dropouts (p=0.3) (Model2)
3) 4 successive attention heads + 3 dense layers with dropouts (Model3). This was not a standard implementation and should be changed for at least a multi-head attention layer (heads functionning in parallel) in future work.

For the ASC model, we consider that False negatives are as costly as False postives so the Macro F1-score and accuracy score will be used to assess the performances. As a result, each model ends with a classification layer using cross-entropy loss and soft-max function to output probabilities, as the cross-entropy loss is well suited to maximize accuracy. Finally, the output does not directly take the form of a Positive/Negative prediction : it is the softmax probability of the positive class.

Indeed, as stated before, several sentences can describe an aspect in a review. So when we apply the model, we finally have to compute the mean of the probabilities of all concerned sentences, to output a final score. Probabilities are assumed to account for the strength of the displayed sentiment and should nuance sentence predictions, leading to an increased performance of the models compared to a basic majority vote among class predictions (0/1) for the sentences of a review.

We do not simply allot the Positive class to reviews with a mean above 0.5 and the Negative class in the inverse case. Instead, we leverage the ROC curve to find the best threshold, that allows to maximize the Macro F1-score on our test set (skytrax ratings). The Macro F1-score is used as we value equally performances on both classes. An example of ROC curve for Model2 trained for 'food and beverages' in given in figure 20.

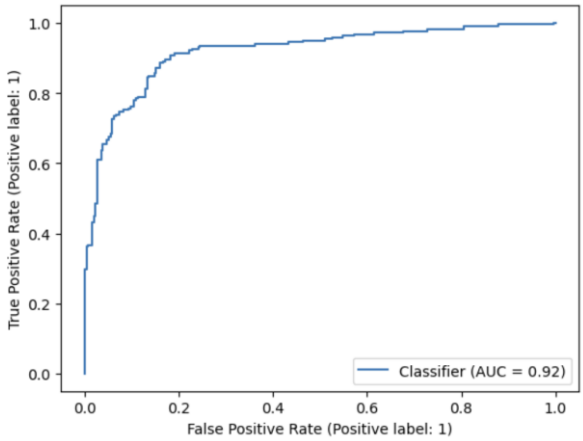


Figure 20 : ROC curve of Model2, for food and bevevarges.

- Training Settings, loss and metrics evolution

The models are trained on Pytorch using the AdamW optimizer, with a batch_size of 32 and 5 epochs on 10 000 observations. The evaluation of the models is performed every 10 steps (that is every 10 batches) on a separate validation set containing 500 observations.

At each validation step, the current model accuracy was compared to the one of the last model and the current model replaces the former model if it outperforms it. Hence, only the best model was saved during each training. Training details can be found it table 21.

Food and beverages

	Best step	**Best validation accuracy**	**weigth_decay (=λ in L2 penalty)**
BERT + 1 dense layer	580	0.828	1,00E-03
BERT + 3 dense layer and dropouts	360	0.832	1,00E-02
BERT + 4 attention heads + 3 dense layers	1390	0.758	1,00E-03

Crew flight

	Best step	**Best validation accuracy**	**weigth_decay**
BERT + 1 dense layer	320	0.814	1,00E-03
BERT + 3 dense layer and dropouts	540	0.79	1,00E-02
BERT + 4 attention heads + 3 dense layers	1180	0.80	1,00E-03
BERT + 4 attention heads + 3 dense layers, 8 epochs	ads	0.784	1,00E-03

Table 21 : Hyperparameters settings and training details

In Figures 22 and 23, we plot the evolution of the training and validation losses along training of Model1 for ‘food and beverages’, as well as the evaluation metrics on the validation data.

Figure 22 and 23 : evolution of the training and validation losses along training of Model1 for ‘food and beverages’

We see that the validation metrics fluctuate a lot with a slight increasing trend in the beginning and tend to quickly level out. On the other hand, we observe overfitting while looking at the loss evolution of the validation set. This is due to an overconfidence of the model in its prediction. Indeed, while the nature a final output (0 or 1) does not change (metrics stagnate), the logits increase too much for the predicted class and decrease for the other class. An overfitted model would be a problem in our case as we sum the probabilities of positive class for several sentences in a review before issuing predictions.

For Model2, in order to reduce overfitting, we introduce regularization in the loss computation. To do so, we increase the weight_decay hyperparameter of AdamW to $\lambda = 1e-2$ instead of $\lambda = 1e-3$ as in Model1. Indeed, this hyperparameter introduces L2 regularization into the model, penalizing large weights that could lead to overfitting. The cross-entropy loss function is then summed with a regularization term :

$$L_{\text{reg}}(\theta) = L(\theta) + \lambda \sum_{i=1}^{n} \theta_i^2$$

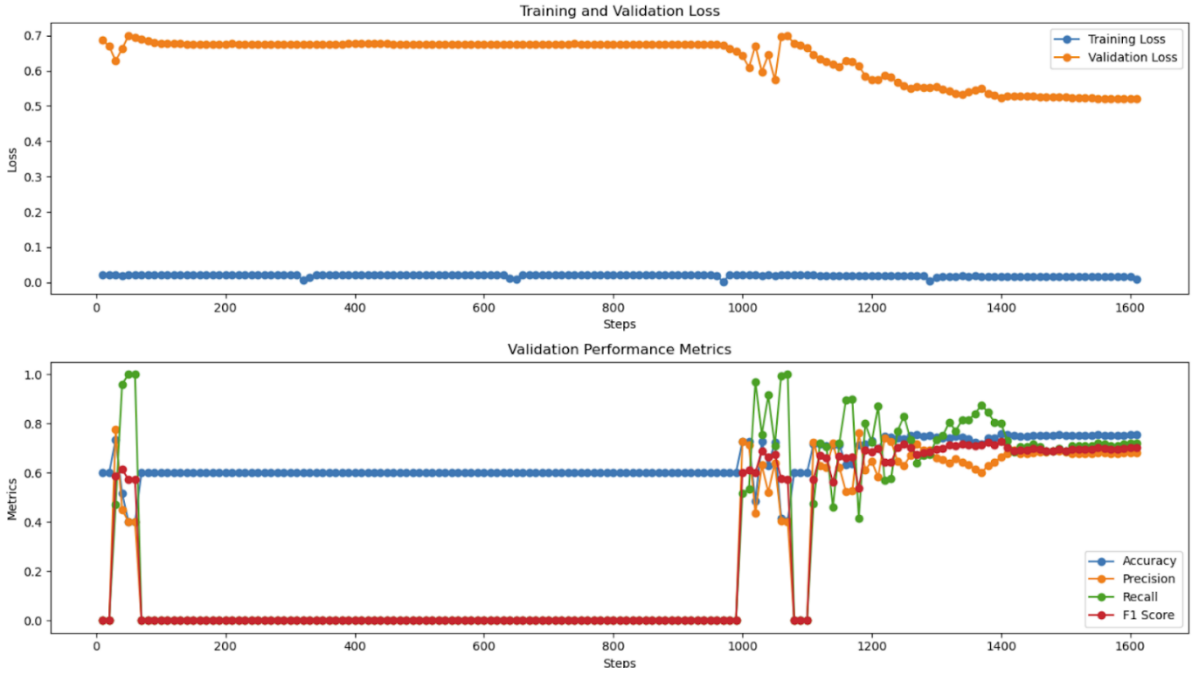
Where θ represent the model weights and L(θ) is the cross-entropy loss :

$$L(\theta) = -\frac{1}{N} \sum_{j=1}^{N} [y_j \log(\hat{y}_j(\theta)) + (1 - y_j) \log(1 - \hat{y}_j(\theta))]$$

where $\hat{y}_j(\theta)$ is the predicted probability for class 1 and y_j is the true label. N = number of datapoints

The Model2 also ended up overfitting the data, but less harshly. In order to avoid this issue, an idea could be trying to introduce more data in our model, by scraping new data or proceeding to data augmentation on the one we have.

For Model3 (with attention heads), in figures 24 and 25 we see that the validation loss started declining after 3 epochs. At the same time, the validation metrics started fluctuating a lot before stabilizing. The model was trained for 3 more epochs inducing a further decrease in loss and metrics improvement. However, the evolution of the loss and metrics along training indicate that the initial learning rate configuration of the model might not be adapted. Indeed, the learning rate was set to 2e-5 decaying linearly to 0. The learning rate was probably too high in the beginning for the model to start training properly. We retrain the model setting a warmup for the 250 first steps. The model fitted quicker but the results did not improve at the end.



Figures 24 and 25 : evolution of the training and validation losses along training of Model3 for 'crew flight'

- Evaluation of the models on the control set

We evaluate each model on 2 test sets : bipolar (for the concerned aspect), as defined in the data exploration part, and random/overall (normal dataset, representing the natural classes distribution in the data).

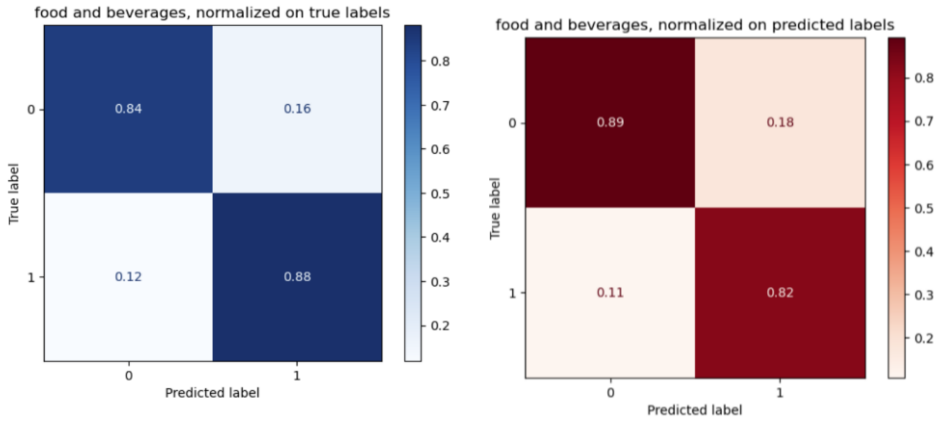
If we first have a look at the table 26, we notice that the best models for aspect 'food and beverages' are Model1 and Model2 on random reviews, as they achieve very close performances according to all metrics. The accuracy is 0.872 for Model1 and 0.870 for Model2. We notice the efficiency of these models is quite balanced between Positve and Negative

classes, with a light advantage to Negative. When looking at the detail of the confusion matrices 32 and 33 for Model2, we observe that the recall for positives is higher than for negatives, and the inverse for precision. We conclude that the model tends to be less conservative in its positive predictions, and produce false positives more easily than false negatives. The same pattern is found for Model1. We note that this phenomenon is likely a consequence of the choice of the threshold, selected to maximize the macro F1 score.

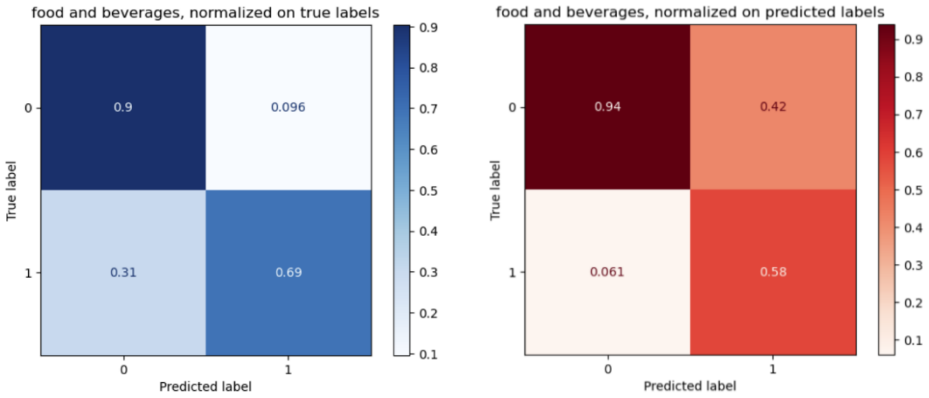
Table 28 presents the results on bipolar reviews for 'food and beverages', the results are based on the best found threshold based on bipolar reviews. In this case, the accuracy is maintained nay increases in all models, but the Macro F1 score decreases, due to the drop in Positives' F1 Score. Indeed, as shown in matrices 34 and 35, the recall and especially the precision (only 58%) for the Positive class are mediocre. For 'crew flight', both accuracy and Macro F1 score decrease sharply compared to random reviews. Overall, all model architectures seem to be less efficient on bipolar data.

In addition, we notice that the best thresholds are not the same for bipolar and random reviews, even for an exact same model architecture. However, when using a model to output predictions on new data, we cannot determine if the new unlabeled reviews are bipolar or not for an aspect. We need a unique threshold. Therefore, we would like to assess the performance of the best overall threshold on bipolar observation. The results are compiled in tables 30 and 31. We see that the Model2 is the best for both 'food and beverages' and 'crew flight', with respective Macro F1 Scores of 79.9% and 63.0%. Moreover, if we compare the outcomes on bipolar reviews between best threshold for bipolars and best overall threshold, the latter only induces a loss of around 2 to 3 points of percentages in Macro F1 scores and accuracies of all models. Consequently, we conclude that we can use the overall threshold and keep a correct efficacy on bipolar reviews.

Eventually, we conclude that our supervised models slightly beat the baseline in the case of randomly selected reviews (as aspects polarities are usually correlated together and to the overall sentiment), but our neural networks totally outperform the baseline in the bipolar case. This is the sign that our models are better tailored to differentiate polarities between aspects when ambivalant sentiments are expressed about various aspects of the airline service.



Matrices 32 and 33 : Model2 food and beverages, random reviews testing



Matrices 34 and 35 : Model1 food and beverages, bipolar reviews for this aspect

Evaluation on random/overall reviews	**food and beverage**					
	Accuracy	F1pos/F1neg	F1 macro	Best threshold	AUC score	F1 macro baseline
BERT + 1 dense layer	0.872	0.856 / 0.886	**0.871**	0.435	0.93	0.858
BERT + 3 dense layer and dropouts	0.870	0.851 / 0.885	**0.868**	0.49	0.93	0.858
BERT + 4 attention heads + 3 dense layers	0.803	0.783 / 0.820	0.802	0.375	0.87	**0.858**

Evaluation on random/overall reviews	**crew flight**					
	Accuracy	F1pos/F1neg	F1 macro	Best threshold	AUC score	F1 macro baseline
BERT + 1 dense layer	0.837	0.818 / 0.852	**0.835**	0.601013	0.88	0.812
BERT + 3 dense layer and dropouts	0.845	0.843 / 0.848	**0.845**	0.492806	0.90	0.812
BERT + 4 attention heads + 3 dense layers	0.825	0.806 / 0.841	**0.824**	0.619636	0.89	0.812

Evaluation on bipolar reviews	**food and beverages**					
	Accuracy	F1pos/F1neg	F1 macro	Best threshold	AUC score	F1 macro baseline
BERT + 1 dense layer	0.873	0.632 / 0.923	**0.777**	0.54	0.84	0.545
BERT + 3 dense layer and dropouts	0.909	0.706 / 0.946	**0.826**	0.66	0.86	0.545
BERT + 4 attention heads + 3 dense layers	0.870	0.519 / 0.925	**0.722**	0.59	0.77	0.545

Evaluation on bipolar reviews	**crew flight**					
	Accuracy	F1pos/F1neg	F1 macro	Best threshold	AUC score	F1 macro baseline
BERT + 1 dense layer	0.706	0.777 / 0.568	**0.673**	0.392462	0.74	0.514
BERT + 3 dense layer and dropouts	0.674	0.726 / 0.598	**0.662**	0.416358	0.74	0.514
BERT + 4 attention heads + 3 dense layers	0.666	0.734 / 0.552	**0.643**	0.373397	0.72	0.514

Best overall threshold on bipolar instances	**food and beverages**				
	Accuracy	F1pos/F1neg	F1 macro	best threshold all	F1 macro baseline
BERT + 1 dense layer	0.844	0.615 / 0.902	**0.759**	0.435	0.545
BERT + 3 dense layer and dropouts	0.881	0.671 / 0.927	**0.799**	0.49	0.545
BERT + 4 attention heads + 3 dense layers	0.779	0.497 / 0.859	**0.678**	0.375	0.545

Best overall threshold on bipolar instances	**crew flight**				
	Accuracy	F1pos/F1neg	F1 macro	Best threshold bipolar	F1 macro baseline
BERT + 1 dense layer	0.599	0.620 / 0.575	**0.598**	0.601013	0.514
BERT + 3 dense layer and dropouts	0.636	0.677 / 0.582	**0.630**	0.492806	0.514
BERT + 4 attention heads + 3 dense layers	0.569	0.573 / 0.565	**0.569**	0.619636	0.514

Tables 26 to 31 : Performance of supervised models (neural networks)

D. Reflexion on other models and improvements

In this section, we discuss the possible alternatives and untested models that could complement our work. If we wanted to push this study to enhance its results, we should consider the underneath points :

Aspect Extraction Part :

- We should focus on improving the Aspect Extraction model to recognize a larger part of sentences mentionning an aspect. As we could not rely on topic modeling to train the supervised model and had to implement a basic procedure based on Lemma Keywords Detection, we can assume that the ASC models we obtained were slightly biased towards those keywords. This observation is mitigated by the fact that even if sentences were assigned to an aspect on the basis of a few keywords, other relevant words related to the same aspect were likely present is the same sentences, participating to the expression of the sentiment, and being taken into account in the supervised models.
- Other Topic modeling algorithms should be tested. For instance, we wanted to try Top2Vec and Biterm Topic Model but were unable to do so due to package installation issues.

Aspect Sentiment Classification :

- Obtaining more data or perform data augmentation to be able to train more efficient and general models.
- Introducing a neutral class to nuance the prediction, instead of forcing a binary classification that defines ‘Negative’ in opposition to ‘Positive’, the former then encompassing both neutral and negative sentiments.
- Architecture of model 3 should be modified. We trained 4 successive attention heads whereas they should be trained in parallel in a multi-head attention layer. This mistake should be corrected.
- Large Language Models like ‘gpt-4’, also based on transformers are usually very well suited for ABSA tasks, supposing a good data preprocessing and a relevant prompt sending. We iteratively sent a few hundred requests using the OpenAI API is order to ask gpt-4 for both Aspect Extraction and Sentiment Classification at the same time. The results were not strictly and massively evaluated, however this method is promising. Nonetheless, the slowness of gpt-4 and its API is a drawback compared to a specifically trained neural network such as ours if thousand of reviews need to be processed. In addition, the use of the GPT-3.5 or GPT-4 is paying.

  The prompt we sent was of the following form : *prompt = f"Analyze the following review and provide sentiment ('Positive', 'Neutral', 'Negative', 'not mentioned') for each aspect: {', '.join(aspects)}. Review: {review}. Be careful before stating 'not mentioned'."*

We note that gpt-4 does not always strictly respect the instruction of returning *'Positive'*, *'Neutral'*, *'Negative'* or *'not mentioned'*, the answers consequently necessitate cleaning and post-processing.

Application Case

In order to prove the efficacy of the models to detect trends in customer satisfaction from the data, and to demonstrate its ability to draw conclusions by aggregating its predictions, we apply the model to the 1405 reviews dealing with 3 major US airlines : American Airines, Spirit Airlines, and United Airlines. The control data was used for this test. On the barplot 36, we see that for both 'food and beverages' and 'crew flight' aspects, the satistaction rate (rate of 4-5 stars ratings) and insatisfaction rates predicted by Model2 are close to the real rates computed from the rated data for these aspect, for which the aspect had been detected in the reviews.

On figure 37, we notice that for all 5 aspects the satisfaction rate is overestimated. The overestimation for 'Food and beverages' and 'Crew flight' is higher with VADER than Model2, suggesting that Model2 might be more suitable to draw global conclusions on customer satisfaction.

However, beware that the satisfaction rate among reviews that cited the aspect in the text and rated it (17.1% of satisfaction) is lower that the satisfaction rate of those who rated it but did not mention it (34.8%). We conclude that unhappy reviewers tend to elaborate about their sentiment in the textual review more than the happy ones.

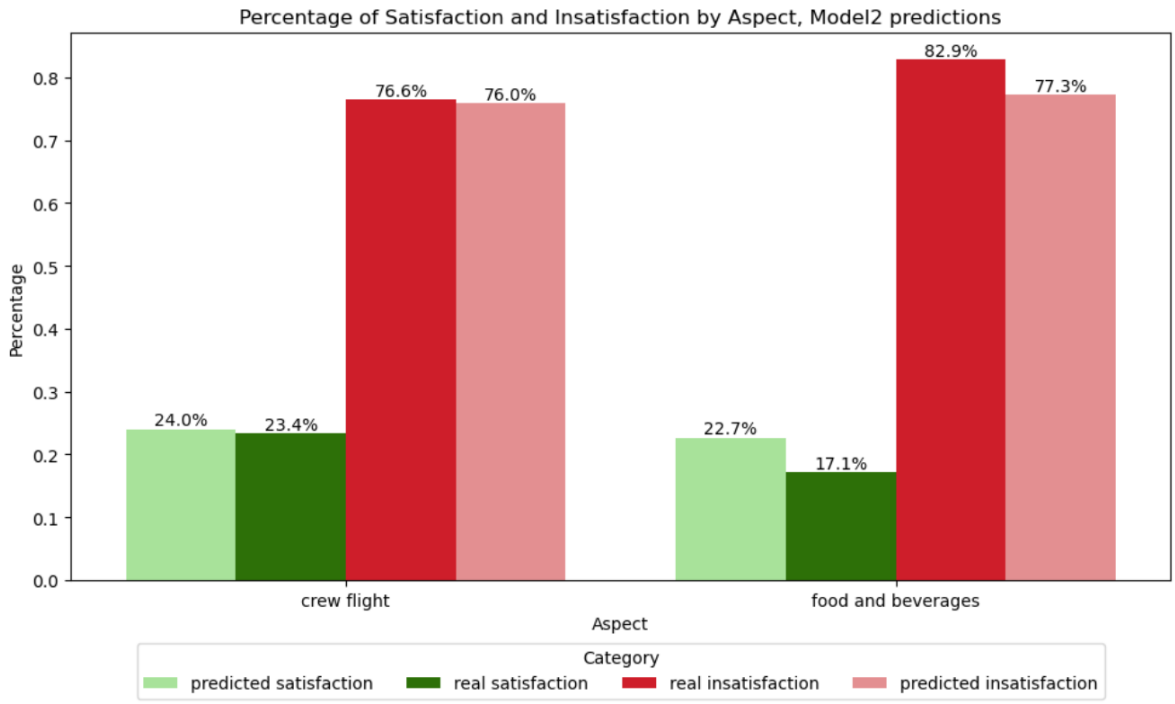


Table 36 : Percentage of Satisfaction and insatisfaction by aspect, Model2 (for each aspect) predictions

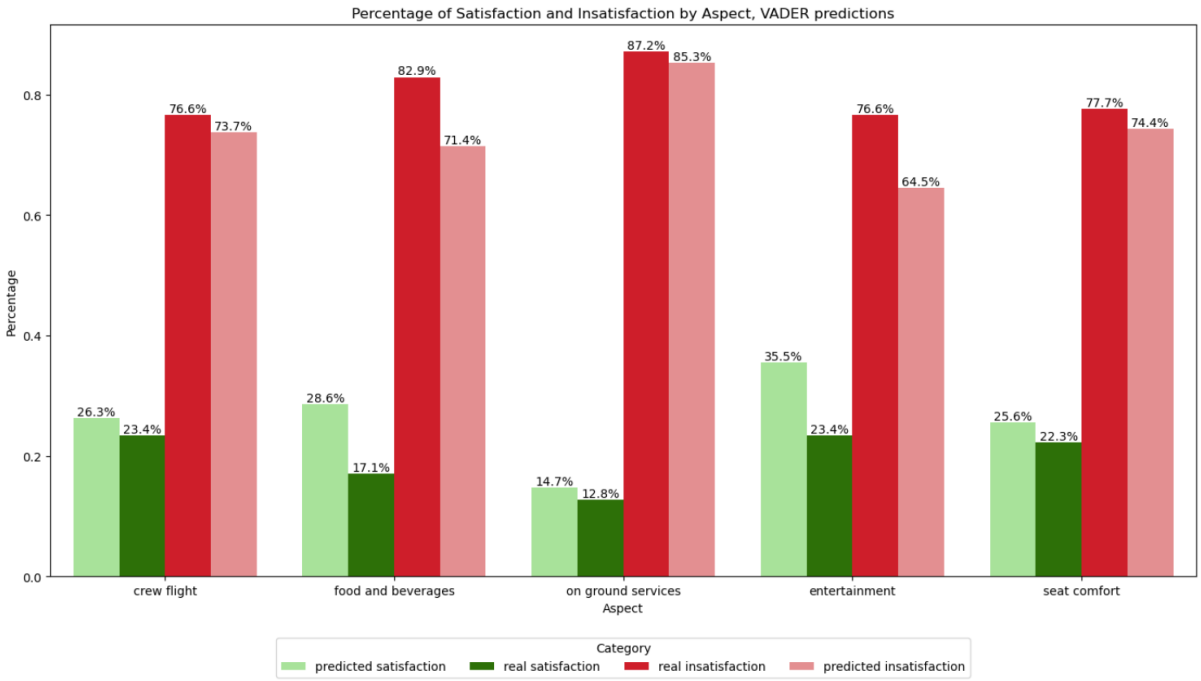


Table 37 : Percentage of Satisfaction and insatisfaction by aspect, VADER predictions

Conclusion

The objective of this study was to build models to perform Aspect-Based Sentiment Analysis on airline passenger reviews, with the aim of detecting and differentiating various aspects of the service and predicting their associated sentiment polarity. To do so, we exploited 110 000 reviews from the website Skytrax about major operating airline. After exploring the literature about ABSA and how it had previously been applied to our use-case, we decided to implement a pipeline of two independent models for Aspect Extraction and Aspect Sentiment Classification. We used a hand-made dataset to assess the Aspect Extraction models. A few procedures were tested (LDA, BERTopic, Lemma Keywords Detection). While we still believe BERTopic is the most promising algorithm, we could not fully exploit its potential in the end and selected the Lemma Keywords Detection method to serve as a Basis for the Sentiment Classification Models. Concerning this second task, both supervised and unsupervised models were tested and compared to a baseline model for 5 main aspects of the service. The baseline model already performed pretty well on the overall dataset – but this result is due to the correlation of aspects between them and with the overall sentiment : 63% of the reviews display a unique polarity for all aspects. The baseline is way less efficient on bipolar data, that express diverging polarities of sentiment for at least one aspect. It turns out that VADER (supervised model) beats the baseline benchmark on bipolar reviews, for all 5 assessed aspects, although is does not outperform it overall (non-oversampled bipolar reviews). However, our supersived models based on BERT + Fully Connected Layers almost all surpass the baseline for both classical and bipolar reviews. The improvement is particularly large for bipolar data. We finally apply the models to predict satisfaction rates on the 5 aspects for 3 major US airlines, in order to demonstrate the capacity of the models to generate useful statistical summaries. Although the Aspect Extraction part could be ameliorated, we have demonstrated the capabilities of our pipeline and the interest of our work and models to perform Aspect-Based Sentiment Analysis on Airline Passenger reviews.

Bibliography

[1] [Mikolov, T. (2013). Efficient estimation of word representations in vector space. arXiv preprint arXiv:1301.3781.]

[2] Vaswani, A. (2017). Attention is all you need. Advances in Neural Information Processing Systems.

[3] Devlin, J. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805

[4] Zhu, L., Xu, M., Bao, Y., Xu, Y., & Kong, X. (2022). Deep learning for aspect-based sentiment analysis: a review. PeerJ Computer Science, 8, e1044

[5] Toh, Z., & Su, J. (2016, June). Nlangp at semeval-2016 task 5: Improving aspect based sentiment analysis using neural network features. In Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016) (pp. 282-288)

[6] Brun, C., Perez, J., & Roux, C. (2016, June). XRCE at SemEval-2016 task 5: Feedbacked ensemble modeling on syntactico-semantic knowledge for aspect based sentiment analysis. In Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016) (pp. 277-281)

[7] Vicente, I. S., Saralegi, X., & Agerri, R. (2017). Elixa: A modular and flexible absa platform. arXiv preprint arXiv:1702.01944.

[8] Pontiki, M., Galanis, D., Papageorgiou, H., Androutsopoulos, I., Manandhar, S., Al-Smadi, M., ... & Eryiğit, G. (2016, January). Semeval-2016 task 5: Aspect based sentiment analysis. In International workshop on semantic evaluation (pp. 19-30).

[9] Angelov, D. (2020). Top2vec: Distributed representations of topics. arXiv preprint arXiv:2008.09470.

[10] Yan, X., Guo, J., Lan, Y., & Cheng, X. (2013, May). A biterm topic model for short texts. In Proceedings of the 22nd international conference on World Wide Web (pp. 1445-1456).

[11] Grootendorst, M. (2022). BERTopic: Neural topic modeling with a class-based TF-IDF procedure. arXiv preprint arXiv:2203.05794.

[12] Wang Y, Huang M, Zhu X, Zhao L. 2016b. Attention-based LSTM for aspect-level sentiment classification. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Austin, Texas: Association for Computational Linguistics, 606–615.].

[13] A hierarchical model of reviews for aspect-based sentiment analysis. In: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing. Austin, Texas: Association for Computational Linguistics, 999–1005.

[14] Zhu, L., Xu, M., Bao, Y., Xu, Y., & Kong, X. (2022). Deep learning for aspect-based sentiment analysis: a review. PeerJ Computer Science, 8, e1044.

[15] Chang, Y. C., Ku, C. H., & Le Nguyen, D. D. (2022). Predicting aspect-based sentiment using deep learning and information visualization: The impact of COVID-19 on the airline industry. Information & Management, 59(2), 103587

[16] Li, X., Bing, L., Zhang, W., & Lam, W. (2019). Exploiting BERT for end-to-end aspect-based sentiment analysis. arXiv preprint arXiv:1910.00883.

[17] Liu, Z., Lin, W., Shi, Y., & Zhao, J. (2021, August). A robustly optimized BERT pre-training approach with post-training. In China National Conference on Chinese Computational Linguistics (pp. 471-484). Cham: Springer International Publishing.

[18] Li, Z., Yang, C., & Huang, C. (2023). A Comparative Sentiment Analysis of Airline Customer Reviews Using Bidirectional Encoder Representations from Transformers (BERT) and Its Variants. Mathematics, 12(1), 53.

[19] Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. Journal of machine Learning research, 3(Jan), 993-1022.]

[20] https://maartengr.github.io/BERTopic/algorithm/algorithm.html

[21] : https://hdbscan.readthedocs.io/en/latest/soft_clustering.html

[22] Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., & Zhou, M. (2020). Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. Advances in Neural Information Processing Systems, 33, 5776-5788.

[23] McInnes, L., Healy, J., & Melville, J. (2018). Umap: Uniform manifold approximation and projection for dimension reduction. arXiv preprint arXiv:1802.03426.

[24] https://huggingface.co/facebook/bart-large-mnli

[25] https://huggingface.co/microsoft/deberta-v3-base

[26] Hutto, C., & Gilbert, E. (2014, May). Vader: A parsimonious rule-based model for sentiment analysis of social media text. In Proceedings of the international AAAI conference on web and social media (Vol. 8, No. 1, pp. 216-225).]