Practicum Report

Project Title	Implicit aspect-based opinion mining and analysis for airline industry reviews
Student ID	19210344; 19210149
Student Name	Kanishk Verma; Vaibhav Oberoi
Student Email	Kanishk.verma2@mail.dcu.ie; vaibhav.oberoi2@mail.dcu.ie
Chosen major:	Computing (Data Analytics)
Supervisor	Dr. Brian Davis
Date of submission	16/08/2020

DISCLAIMER

A report submitted to Dublin City University, School of Computing for module CA640 Professional and Research Practice, 2019/2020.

I understand that the University regards breaches of academic integrity and plagiarism as grave and serious. I have read and understood the DCU Academic Integrity and Plagiarism Policy.

I accept the penalties that may be imposed should I engage in practice or practices that breach this policy. I have identified and included the source of all facts, ideas, opinions, viewpoints of others in the assignment references. Direct quotations, paraphrasing, discussion of ideas from books, journal articles, internet sources, module text, or any other source whatsoever are acknowledged and the sources cited are identified in the assignment references.

I declare that this material, which I now submit for assessment, is entirely my own work and has not been taken from the work of others save and to the extent that such work has been cited and acknowledged within the text of my work.

By signing this form or by submitting this material online I confirm that this assignment, or any part of it, has not been previously submitted by me or any other person for assessment on this or any other course of study. By signing this form or by submitting

material for assessment online I confirm that I have read and understood DCU Academic Integrity and Plagiarism Policy

(available at: http://www.dcu.ie/registry/examinations/index.shtml)

Name(s): Kanishk Verma; Vaibhav Oberoi

Date: 16th August 2020.

Implicit aspect-based opinion mining and analysis of airline industry based on user generated reviews

Kanishk Verma
School of Computing
Dublin City University
Dublin, Ireland
kanishk.verma2@mail.dcu.ie
0000-0001-7172-4098

Vaibhav Oberoi
School of Computing
Dublin City University
Dublin, Ireland
vaibhav.oberoi2@mail.dcu.ie

Abstract – Mining opinions from reviews has been a field of ever-growing research. These include mining opinions on document level, sentence-level, and even aspect level of a review. While explicitly mentioned aspects in a review have been widely researched, very little work has been done in gathering opinions on aspects that are implied and not explicitly mentioned. In this paper, we present a novel study for extracting and analysing opinions from airline reviews. We were able to develop an airline domain-specific aspect-based *corpus*, and a technique which augments pre-trained word embeddings for sequential labelling with stochastic gradient descent optimized *conditional random fields*. The results of this method were used to extract and classify implied aspects from opinionated texts using state-of-the-art *machine* and *ensemble learning* approaches.

Index Terms – Conditional random field, machine learning, ensemble learning, implicit aspects, airline, classification, corpus, sequential labelling**.**

I. Introduction

Travel and tourism are well-liked terms amongst all generations of people. The airline industry is a key facilitator in this domain. For this industry, serving its customers with not only cost-effective but also satisfactory service options is paramount.[1] Gone are the days when passengers were required to fill feedback forms during their journey. In this 21st information age, with constant development in social and web media, a multitude of platforms are available like *Trip Advisor*, *Airline Ratings*, etc, for consumers to express their views on air travel. This also serves in favour of the airline companies, as it becomes their one-stop to access rich customer feedback information. Also, opinions are very important to businesses and organizations because they always want to find consumer or public opinions about their products and features. [2]

Many times, due to a variety of reasons like paid promotions, fraudulent, and even unstructured nature of these reviews, insightful information cannot be extracted. So, a need is felt to have a mechanism that can gather cognizance in terms of the perception of customers on airline-specific aspects. [3] The present study provides a mechanism to gather opinions and aspects from such reviews which are not explicitly mentioned

Liu and Zhang et. al. defined the term opinion as a concept covering sentiment, evaluation, appraisal, or attitude held by a person. [2] Aspects and entities are more like topics in a text document. *Hu and Liu et. al*. coined this type of analysis as feature-based sentiment analysis. [4]. Aspect or entity-based analysis identifies the target of the opinion. It is a fine-grained approach for text analysis.

In this paper, an entity is the feature of the airline and an implicit aspect or sub-aspect is its attribute. Example for entities are *food, cabin, seat, staff*, etc. Since these entities in themselves can have various attributes associated with them. It becomes important to divide them further into sub-aspects or implicit aspects. For example, *the cabin* is not always referred independently, it has its attributes like *space, condition*, and *temperature* mentioned along with it. For example, a review for cabin looks like "*the cabin was cold and a bit weary"*. Here opinion terms like "*cold"* and "*weary"* become sub-aspects for temperature and condition respectively. This approach is used in this present study to make a fine-grained analysis of opinions and map them accurately to the respective entity sub-aspect pair.

The motivation behind this study, is to understand which passenger airline industry specific aspects can be leveraged for implicit aspect-base opinion mining. Also, how will these implicit aspects be annotated[1] to build a sentiment corpus, and what specific lexicon[2] generation techniques can influence this type of opinion mining.

Trip Advisor and *Airline Ratings* are online microblogging platforms primarily used for viewing reviews and experiences of travellers either travelling to the same destination or other, all over the globe. Usually, people before making airline ticket purchases do read reviews. [4]

Using the *selenium framework*, a python-based script was developed to collect these reviews meticulously without extracting any personal information of review authors to comply and adhere to GDPR[3] laws.[5]

In this study, 3000 reviews were collected within a time period of 1 month (from 1st November to 30th November 2019) with an aim to study public's opinion with respect to 16 Airlines. (See appendix A). From these 3000 reviews, after curating only a 1803 reviews were determined to be relevant for this study.

In summary, the goal of this study is to extract implied aspects and opinions from airline reviews. To achieve this goal, we created a new dataset, which to our knowledge, is the first time a dataset specifically for implicit aspects of airline reviews is created. Using a supervised lexicon-based technique, we ran a few experiments to gather insightful information about airline-based implied aspects and opinions. The results of which were favourable for the study. Further in this paper, we will discuss our methodology, experimental setup, and evaluations/results of our approach.

II. Methodology

The methodology of our approach consists of multiple modules. Each module was developed keeping in mind that the dataset is fresh, new and one of kind. So, the methodology pipeline includes data gathering, corpus statistics, annotation, feature engineering, sequence labelling and classification. The details can be seen in figure 1.

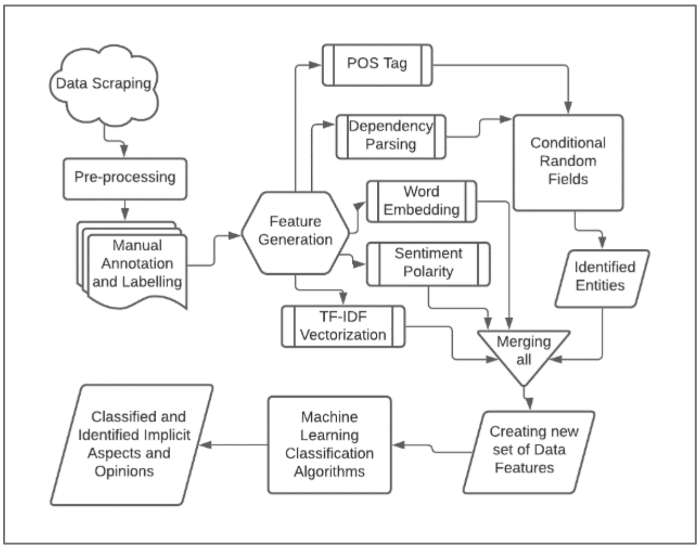

Fig.1. Methodology Framework

Dataset

About 1803 manually curated reviews for 16 airlines were gathered from the blogging websites. Detailed statistical analysis was carried on the dataset, to understand the quality of it. Techniques used are available in the table 1.

Table 1. Dataset Statistics

Number of reviews	1803
Number of sentences	9591
Average number of sentences in a review	3
Type Token Ratio[6]	0.27
Most common word	Flight
Average word length	6
Labelled unique word corpus size	3280

Entity and Aspect Selection

About 500 reviews were read carefully, the two annotators made a list of features of passenger aircrafts, services offered by airlines both in and off. After curating the list, a decision was made to enlist entities into 8 categories. A representation of the entities can be found in table 2.

Every entity had its own attributes i.e. for example, *food* has features that are *service* based including *temperature* and *taste*. So, all the 8 entities were categorized further into implicit-aspects or sub-aspects. This decision was taken

[1] Annotation: It is like a metadata tag to markup specific elements in a dataset.

[2] Lexicon: It is a component of natural language processing that contains grammatical information about individual words or strings

[3] GDPR: European Union's General Data Protection Regulation that lays down rules relating to protection of natural persons.

after performing exploratory data analysis and visualising them. (See appendix A)

Table 2. Entity-wise implicit aspect list

Entity	Implicit Aspect(s)
Food	Service Temperature Taste
Entertainment	Visual Audio General
Cabin	Condition Fragrance Size Temperature
In-flight service	Operations Facility
Off-flight service	Ticketing General Facility
Staff	Behaviour General
Seat	Operations Comfort
Possession	Handling General

Data Annotation

Manual annotation and labelling of all the reviews using Doccano [7] annotation tool was conducted. An inter-annotator agreement guideline [8] was also set up. (See appendix A) Annotation was done on two levels i.e. entity level and implicit aspect level. So, using Cohen's Kappa coefficient [9], annotators agreement level was determined. The results of Kappa coefficient are available in Evaluation section. (See Appendix A)

Feature Engineering

The feature engineering task was divided in two methods, one to capture word features and the other to gather numeric representations of the word features. List available in table 2. (See Appendix B)

Table.3 Feature Engineering Tasks

Word Features	Parts-of-speech tags; Dependency parsing
Numeric word representation	Count Vectorizer; Term Frequency – Inverse Document Frequency; Augmenting Word Embeddings

Augmenting Word Embeddings: The numeric representations like count vectorizer and TF-IDF are more frequency based and lack contextual information. [10]The dataset for this study being small and limited, a need was felt to augment a pre-trained word embedding model with word vector representation of the dataset. So, pre-trained Glove [11] vectors trained on user-generated were used. These pre-trained vectors were augmented with Word2Vec[12] for corpus embeddings. Also, the parameters augmented are one's that considered maximum distance between focus word and its contextual neighbour. (See appendix D)

Sequence Labelling with Conditional Random Fields

Sequence Labelling is a supervised learning [4] task where a label is assigned to each element of a sequence. For our study, to extract words and classify them into respective entities, a conditional random fields algorithm was selected. Conditional random fields [13] adjust to a variety of statistically correlated features as input just like a sequential classifier. Also, like a generative probabilistic model it trades-off decisions at different sequence to obtain a global optimal labelling. (See appendix E)

The CRF model was optimized using stochastic gradient descent[5] with L2 regularization[6]. This is done to maximize the likelihood of the CRF and be represented as follows,

$$lo\,g\,P(Y|X) = w\,.\varphi(Y,X) - \left(log \sum\nolimits_{Y^T} e^{w\varphi(Y,X).}\right)$$ [26]Error! Bookmark not defined.

After taking derivates on the above equation, we get below,

$$\frac{d}{dw}\log P(Y|X) = \varphi(Y,X) - L^2 \sum\nolimits_{Y^T} P(Y^T|X)\varphi(Y^T,X)$$

Where it means $\varphi(Y,X)$ to add correct feature and subtract $P(Y^T|X)$ which is expectation of features and L^2 is a regularization penalty term.

[4] It means learning a mapping between a set of input variables X and output variables Y and applying these mappings, predictions can be made for unseen data

[5] Gradient Descent: An optimization algorithm used to minimize some function iteratively.

[6] L2 regularization: It is a penalty regularization technique which does not let the algorithm over-fit.

Classification for implicit aspect extraction

The aspect extraction task needed classifier models that could accurately predict the aspect. Different algorithms were used to classify and compare how accurate each model was to classify these sub-aspects. Algorithms like Support Vector Machine, Decision Trees, Random Forest, a bagging ensemble learning algorithm Voting Classifier and a boosting ensemble learning algorithm XGBOOST were used.

Decision Tree: It is a recursive classification procedure that partitions dataset into smaller groups based on a set of tests defined at each branch.[14] (See Appendix F)

Random Forest: It is essentially an ensemble classifier that uses several decision trees and outputs the class that is predicted by the maximum number of trees. It is not dependent on any decision tree, instead it is dependent on a bunch of them making it robust. It is a way to decrease variance of the prediction by generating supplementary data from training set using several combinations with repetitions. This implements Brieman's bagging technique. [15] (See Appendix F)

Voting Classifier: It is an ensemble learning method. It is a wrapper of a set of different algorithms which are trained and evaluated parallelly to make use of features of each algorithm.[16] In our study, three classification algorithms are wrapped within the voting classifier. They include decision trees, random forest and Extra Trees Classifier. (See Appendix F)

XGBOOST: Boosting means training a sequence of classifiers one after another so that the last classifier is trained in a better and efficient manner to predict class labels for examples the previous classifiers performed poorly on. The boosting algorithm has been optimized using Algorithm 1.

Algorithm 1 Boosting Algorithm
Input *Full training set of N examples; maximum ensemble size T; sample size L ¡ N.*
Approach to Boosting
Assign an equal weight of 1/N to all training examples

for i=1 to T **do**
a) based on current weights, randomly sample L examples from training set without replacements
b)train classifier on this sample
c) identify misclassifed examples
d) increase weights for misclassified examples
Output: ***Final Model based on all classifiers***

Algorithm 1. Boosting Algorithm

Using this approach, the present study identified XGBOOST as the boosting algorithm to classify sub-aspects.[17] (See Appendix F)

III. Experimental setup and Discussions

Data Scraping: After getting permission from Trip-Advisor and Airline Ratings website, making use of Selenium web-drivers, bots were created.[18] Any form of personal data or personal identity identifying data such as review author's name, review author's id, timestamp of review, review number and even flight ticket details was not extracted. This was done to adhere and abide by the GDPR laws.[5]

Data Pre-processing: Using standard pre-processing techniques like removing domain-specific stop words, removal of unnecessary punctuations, spell correction, converting numbers to words, and word standardization. The motivation for doing so was to avoid misleading the training model. Also, since the data was user generated, there were many contractions of words, for example, couldn't, can't, aren't, I'm etc were seen quite often in the texts. So, fixing these contraction words was also a part of the study. Words like couldn't were replaced by could not. (See Appendix G)

Corpus Statistics: The data being user generated was raw and unstructured. It is the first time this group of reviews were considered text mining and analysis. So, two statistical strategies, viz, type token ratio [6] and Zipf's distribution[19] were used to determine variability in dataset.

Type Token ratio (TTR) is represented as follows, (See Appendix H)

$$TTR = \frac{(number\ of\ types)}{(number\ of\ tokens)} [6]$$

Table 4 Type Token Ratio Scores

Source	TTR Score
Trip Advisor	0.35
Airline Ratings	0.37

TTR Scores are low for both data sources, this means that there are many repeated terms in the corpus. (See appendix H)

Zipf's law states that a relationship between frequency of word (f) and its position in the list i.e. its rank (r) is inversely proportional to one another. [19] (Appendix H)

$$f \propto \frac{1}{\mathrm{r}} [19]$$

Manual Annotation: As explained in the methodology, annotation was done on two levels using Doccano software[7]. There are detailed examples and explanation of this manual annotation strategy.

Table 5. Detailed example of Level 1 annotation

INPUT: "Overall the experience was comfortable and spacious with delicious meals"
Output: [("experience was comfortable", "Inflight"), ("spacious", cabin), ("delicious meals", "food)]

Once, entity-level tuples[7] were tagged containing word or word phrases with entity-name, as seen in Table 5. After completing entity level annotation, another fine-grained approach to classify entity-wise word or word phrases to their respective implied aspects was conducted.

Table 6. Detailed example of Level 2 annotation

INPUT: [("experience was comfortable", Inflight), ("spacious", cabin), ("delicious meals", "food)]
OUTPUT: [[("experience", inflight-operations), ("comfortable", inflight-operations)], [("spacious", cabin-size)], [("delicious", food-taste), ("meals", food-service)]]

Cohen's Kappa Co-efficient and Inter Annotator Agreement: As explained in methodology of this experiment study, after adhering with the guidelines in the inter-annotator agreement, and using Sklearn Kappa score library, the Cohen's Kappa [9]score for level of agreement was calculated.

Training Data Preparation: The experiment study used techniques described in the methodology section for preparing the training data. Taking an example sentence, this process will be explained in detail.

Example sentence: "*Overall, the experience was comfortable and spacious with delicious meals."*

Table 7. Annotated and labelled list of example sentence

Entity Level	
Entity	Word/Phrases
In-flight service	Experience was comfortable
Cabin	Spacious
Food	Delicious meals
Implicit Aspect Level	
Aspect	Word
Inflight Operations	Experience
Inflight Operations	Comfortable
Cabin Size	Spacious
Food Taste	Delicious
Food Service	Meals

From this review, words like *experience*, *comfortable*, *spacious*, *delicious*, and *meals* were identified as aspect terms and their semantic and syntactic information was extracted by parsing them through off-the shelf state of the art models like Stanford Core NLP API [20] to extract part-of-speech tags and dependency tags, and Vader for sentiment score

Part-of-speech and Dependency Tags:

Table 8. POS-tags using Stanford Core NLP

Input: *"Overall the experience was comfortable and spacious with delicious meals"*
Pre-processing: *"overall experience comfortable spacious delicious meals"*
POS-Tags: [(*'overall'*, 'JJ'), (*'experience'*, 'NN'), (*'comfortable'*, 'JJ'), (*'spacious'*, 'JJ'), (*'delicious'*, 'JJ'), (*'meals'*, 'NNS')]

Here the tags "JJ", "NN" and "NNS" mean adjective, noun and singular noun respectively. (See Appendix B)

The information from the dependency parsing is captured in a list of tuples and stored in a data frame in python, (see Appendix B)

[7] Tuples are a data type that is similar but also distinct to the list data type. The instances are characterized by having fixed attributes and the elements of a tuple instance can differ in data type amongst one another.

Vader Sentiment Score: Being a simple rule-based model which generates sentiment analysis for social media texts, sentiment score for label words and their dependent words was added to the tuple. [21] (See appendix B)

Using these techniques, following information for labelled words was extracted in the form, (Main-word, Main-word POS Tag, Dependent word, Dependent word POS Tag, Main-word sentiment score, Dependent Word sentiment score, Dependency Tag, Previous, Next Word)

List of tuple features: [(“experience”, “NN”, “comfortable”, “JJ”, 0.0, 0.4, “amod”, “overall”, “comfortable”), ..., (“delicious”, “JJ”, “meals”, “NNS”, 0.6, 0.0, “advmod”, “spacious”, “meals)]

For the task of sequence labelling in order to identify the entity a word or word phrase belongs to, the tuples were added with their respective labels i.e. the label added to a tuple was the label that word belonged to.

For example, Tuple, (“delicious”, “JJ”, “meals”, “NNS”, 0.6, 0.0, “advmod”, “spacious”, “meals) has main word *food*, so a new entry to this was made as *“f”*, which became the *Y* or dependent variable.

After getting results from the CRF model, the entity-id i.e. if it was classified as “*food*” then *id* was “*f*”.

Table 9. Entity ID List

Entity	Entity-ID
Food	f
Cabin	c
Entertainment	e
Staff	st
Seat	s
Off-flight	o
In-flight	i
Possession	p

Once the correct entity is identified, the next step is to classify which aspect is mentioned in the sentence. Then to the word feature tuple ENTITY-ID is added to the training data and it is then vectorized.

Vectorization: The main word, dependent word, previous word and next word are replaced by their numeral values.

Count Vectorization: For this experiment study, since the methodology does try to keep certain punctuations and special characters, a need is felt to create own tokenizer. The results for an example sentence

Sentence: *“so overall I highly recommend this airline”*

Vectors: {“*so*”:5, *“overall”*:3, *“I”*:2, *“highly”*:1, *“recommend”*:4, *“this”* :6, *“airline”* :0}

TF-IDF Vectorization: For this experiment study, TF-IDF score for the words in the feature sets was calculated using sci-kit learn tf-idf vectorizer. Table 10 has the result of tf-idf scores for all corpus words.

Table 10. TF-IDF Vectorization

Word	TF-IDF score
Basic	0.965545
Redemption	0.965545
Rescue	0.958253

Word Embeddings: As mentioned in the methodology, the corpus of this experiment study was small. So, a word embedding model using Word2Vec for the corpus was trained. And a pre-trained Twitter Glove Embeddings consisting of a vocabulary size of 1.2 million words and 27 billion tokenized twitter words with a 100-dimensional vector was selected.

Using the algorithm 2, a set of new vector embeddings were merged using pre-trained Glove and corpus Word2Vec embeddings.

Algorithm 1 *Word embedding vector generation using pre-trained glove vectors*

Inputs
$S = [W_1, W_2, W_3, ..., W_n]$, *Input sentence S contains n words*
path = path where downloaded embeddings are stored
GloveVec = Pretrained Glove Vectors
Output
Word2Vec Embedding Model

```
// Load the Glove Pre-trained Vectors
with open(path):
  gloveVec = embedding vectors // Create Word2Vec Embedding for Airline Corpus
word2vec = Word2Vec Create Model

for word, vector in zip(word2vec.index2word, word2vec.vectors) do
 | w2v = dict(word: vector)
end
// Vectors for airline Corpus are loaded
for each W_i in S do
  if W_i exists in gloveVec then
  | extract vecW_i  MV_i = vecW_i end                                   end
  else if W_i exists in w2v then
  | extract vecW_i  MV_i = vecW_i end
  else
  |  extract vecW_i  MV_i = generateNewvecW_i end
```

Algorithm 2. Custom word embedding algorithm

With this algorithm 2, a new set of word embeddings were generated to vectorize main, dependent, previous and next words.

Cosine Similarity Index: Along with the word embeddings, cosine similarity between main and dependent word was added as a new feature. (See Appendix D)

To identify words with high cosine similarity index and the relations between the word vectors of these words, using T-Distributed Stochastic Nearest Neighbour technique[22], visual representation can be understood in figure 2. (See Appendix D)

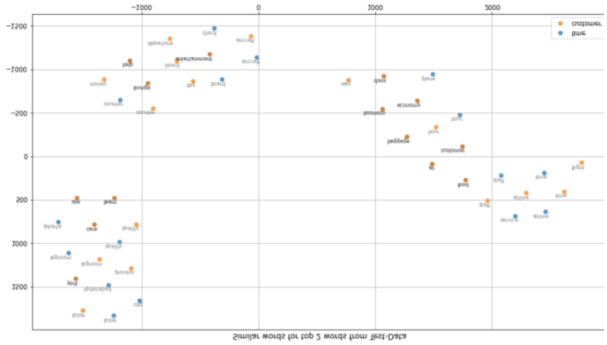

Fig 2. T-SNE Visualization

These new features were then used to classify opinionated texts into their respective implicit-aspect classes.

Handling Class Imbalance: After annotation, there was high imbalance amongst implicit aspect classes of almost all entities. The imbalance for entity cabin, can be seen below.

Class: {"*Condition*":182, *"Size"* :61, *"Temperature"* :39, *"Fragrance"* :20}

This imbalance was handled using an oversampling technique called Synthetic Minority Oversampling Technique. (SMOTE) [23] (See appendix F). Results of SMOTE imbalance handling is as follows,

Class: {"*Condition*":182, *"Size"*:182, *"Temperature"*: 117, *"Fragrance"*: 102}

This could be visualized as a scatter distribution shown in figure 3.

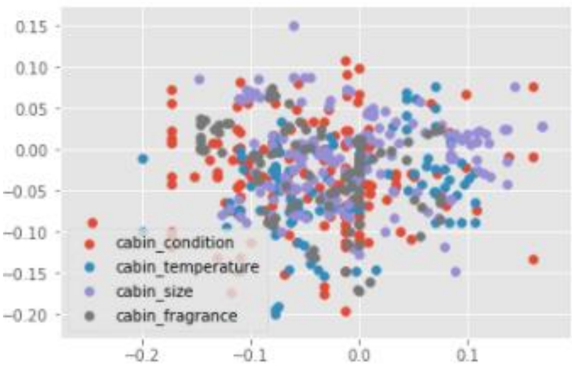

Fig 3 Scattered class distribution after handling imbalance using SMOTE

Implicit Aspect Classification: A total of 8 models were created for each entity i.e. there are independent classification models for training each entity.

This experiment study makes use of state-of-the-art classification algorithms. Three of which were ensemble learning techniques. These include, Gradient boosting algorithm – XGBOOST, a Voting Bagging algorithm using three tree-based classification techniques like Decision Trees, Random Forest and Extra Trees Classifier. And other machine learning techniques like SVM, Decision Tree.

The reason for using these different algorithms was to gather insightful information on the performance of classification which was evaluated based on ROC-AUC[24] score and F1 [25] scores. (See Appendix I)

IV. Evaluation and Results

This experiment study using state-of-the-art techniques and algorithms is a new approach to mine and extract implicit aspects from opinionated texts.

First evaluation was for the annotation of the dataset using Cohen's Kappa Co-efficient.

The two annotators agreement score ranged from **80.48% to 82.13%** for entity level and implicit aspect level annotation. (See Appendix B)

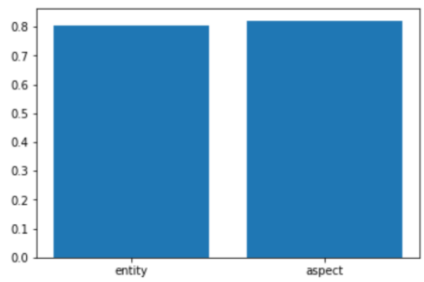

Fig 4. Kappa Coefficient scores

Second evaluation was for the sequence labelling task using a stochastic gradient descent with L2 regularization Conditional Random Field. This was done to classify texts in 8 different entities.

The ROC-AUC score achieved for this task is **96.5%** and a F1 score of **94.56%**. (Appendix I)

Third evaluation was for the classification task using five different classification algorithms. (See Appendix I)

A detailed ROC-AUC score evaluation metric is available in Table 8

Table 11
ROC-AUC Scores for classification of entities

Entity	Algorithms				
	S	D	R	V	X
Food	84%	92%	94%	94.8 %	94.7 %
Cabin	75 %	75 %	85%	85.6 %	77 %
Entertainment	73.6 %	79.9 %	83.1 %	84.3 %	85.9 %
In-flight	60.3 %	70.3 %	72.2 %	74.9 %	71.2 %
Off-flight	66.4 %	86.2 %	84.9 %	84.8 %	89.8 %
Possession	66.9 %	66.9 %	70.5 %	73.3 %	73.4 %
Seat	66 %	73.7 %	75 %	75.7 %	78%
Staff	75.6 %	76.9 %	80.9 %	82.1 %	81.4 %

In the table 8, S stands for Support Vector Machines, D for Decision Trees, R for Random Forest, V for Voting Classifier and X for XGBoost algorithms. In all these machine learning and ensemble learning classification algorithms, the bagging technique of ensemble using tree-based classifiers has out-performed all other classification algorithms. (Appendix I)

V. Related work and improvements

Our research concentrates on implicit aspect extraction, opinion lexicon generation, and engineering an annotated implicit aspect-based sentiment corpus that can influence implicit opinion mining from consumer reviews in the airline industry. Few studies that are done in this realm of implicit aspect-based opinion mining and extraction but very few on implicit aspect-based opinion mining.

In a research study proposed by *Chinsha T C et al.* the methodology proposes a syntactic based approach using dependency parsing[8].[26] In another research for comparing word representations for implicit classification.[27] Both these studies use *SentiWord Net* and have dataset restrictions. The present study intends to extend the results of these two papers. By using a syntactic approach to group implicit aspect synonyms for a larger dataset. As the two studies were restricted to 170 and SemEval dataset respectively.

Research dealing with the double-implicit problem[9] in opinion mining and sentiment analysis proposes a protocol to derive a labelled corpus for implicit polarity and aspect analysis. [28] The work in this paper is limited to only Chinese restaurant reviews. The present study addresses not only the dataset limitation but also the labelling of corpus technique by using Type/token Ratio and other corpus statistic techniques which are explained in the experimental setup section III.

Another study using two corpora proposed a hybrid model to support Naïve Bayes training to identify implicit aspects. [29] This corpus and dictionary-based approach is limited to only adjective type words of a sentence. The present study extends this work by taking considering a combination of adjectives, adverbs, nouns, and other part-of-speech indicators and uses ensemble learning for classification.

A study conducted on implicit aspect indicator extraction, models relations between the polarity of a document and its opinion target using Conditional Random Field (CRF). [30] This method is limited however to only cellular device data and the entities are picked from a pre-trained Stanford CRF model. Our work extends Conditional Random Field and extends it to the airline domain.

VI. Conclusion and future work

The present research study using a supervised machine learning approach provides a novel technique to overcome the implicit opinion and aspect mining problem. It does so by, identifying eight different airline industry specific aspects that can be leveraged for the task of opinion mining. They include fine-grained entities like cabin, entertainment, food, in-flight service, off-flight service, seat, staff and possessions. The annotation being done on two levels, one on the entity level and the other is on the sub-aspect level, allows for a more detailed label construction. The two annotators in this experiment study have a very good agreement on annotated terms. This can be reflected by a Cohen's Kappa score ranging from 0.77 to 0.80. So, it can be said that the corpus derived from this study, can be used as a gold standard for implicit

[8] Dependency parsing: A methodology that is used to extract grammatical structure from sentences.

[9] Double-implicit: Word or word-phrases that not only describe an entity but also the opinion of the entity.

aspect-based mining tasks for airline reviews. This experiment study presents a novel approach of dividing the implicit aspect-based opinion mining task in two levels, one using stochastic gradient descent with L2 regularization for improving conditional random fields to identify entities. This is done with a ROC-AUC Score of **96.58%**, a F statistic score of **94.56%** and with **0.01** degrees of mean absolute error on testing data. The second level is to classify each entity into implicit aspect sub-groups. For this state-of-the-art machine and ensemble learning algorithms are used. From the experiments, it is found that ensemble learning outperformed the machine learning approaches. The ROC-AUC scores for ensemble learning algorithms like Voting Classifier ranges from **73% to 94.8%** and the boosting algorithm like XGBOOST ranges from **71% to 94.7%** for all eight entities. Synthetic Minority Oversampling technique proved to be an effective performance improver for the classification and extraction of implicit aspects. task

The scope of this experiment study is limited to a few reviews, as a possible future work, another study can carry forward the methods proposed in this paper to a larger dataset. Also, another possible future work can be implementing a neural architecture of these proposed methods.

VII. References

[1] F. Misopoulos, M. Mitic, A. Kapoulas, and C. Karapiperis, “Uncovering customer service experiences with Twitter: the case of airline industry,” *Management Decision*, vol. 52, no. 4, pp. 705–723, Jan. 2014, doi: 10.1108/MD-03-2012-0235.

[2] B. Liu and L. Zhang, “A Survey of Opinion Mining and Sentiment Analysis,” in *Mining Text Data*, C. C. Aggarwal and C. Zhai, Eds. Boston, MA: Springer US, 2012, pp. 415–463.

[3] K. Wongleedee, “Customer satisfaction in the airlines industry: comparison between low-cost and full service airlines,” *Актуальні проблеми економіки*, no. 1, pp. 218–222, 2017.

[4] M. Hu and B. Liu, “Mining and Summarizing Customer Reviews,” p. 10.

[5] “General Data Protection Regulation (GDPR) – Official Legal Text,” *General Data Protection Regulation (GDPR)*. https://gdpr-info.eu/ (accessed Aug. 17, 2020).

[6] M. C. TEMPLIN, *Certain Language Skills in Children: Their Development and Interrelationships*, NED-New edition., vol. 26. University of Minnesota Press, 1957.

[7] *doccano/doccano*. doccano, 2020.

[8] “Natural Language Annotation for Machine Learning [Book].” https://www.oreilly.com/library/view/natural-language-annotation/9781449332693/ (accessed Aug. 16, 2020).

[9] A. Rosenberg and E. Binkowski, “Augmenting the kappa statistic to determine interannotator reliability for multiply labeled data points,” in *Proceedings of HLT-NAACL 2004: Short Papers*, Boston, Massachusetts, USA, May 2004, pp. 77–80, Accessed: Aug. 15, 2020. [Online]. Available: https://www.aclweb.org/anthology/N04-4020.

[10] O. Levy, Y. Goldberg, and I. Dagan, “Improving Distributional Similarity with Lessons Learned from Word Embeddings,” *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 211–225, 2015, doi: 10.1162/tacl_a_00134.

[11] J. Pennington, R. Socher, and C. Manning, “GloVe: Global Vectors for Word Representation,” in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Doha, Qatar, Oct. 2014, pp. 1532–1543, doi: 10.3115/v1/D14-1162.

[12] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, “Advances in Pre-Training Distributed Word Representations,” *arXiv:1712.09405 [cs]*, Dec. 2017, Accessed: Apr. 21, 2020. [Online]. Available: http://arxiv.org/abs/1712.09405.

[13] J. Lafferty, A. McCallum, and F. C. N. Pereira, “Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data,” p. 10.

[14] S. R. Safavian and D. Landgrebe, “A survey of decision tree classifier methodology,” *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, pp. 660–674, May 1991, doi: 10.1109/21.97458.

[15] A. Cutler, D. R. Cutler, and J. R. Stevens, “Random Forests,” in *Ensemble Machine Learning: Methods and Applications*, C. Zhang and Y. Ma, Eds. Boston, MA: Springer US, 2012, pp. 157–175.

[16] S. Saha and A. Ekbal, “Combining multiple classifiers using vote based classifier ensemble technique for named entity recognition,” *Data & Knowledge Engineering*, vol. 85, pp. 15–39, May 2013, doi: 10.1016/j.datak.2012.06.003.

[17] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” in

Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco California USA, Aug. 2016, pp. 785–794, doi: 10.1145/2939672.2939785.

[18] R. Mitchell, *Web Scraping with Python: Collecting More Data from the Modern Web*. O'Reilly Media, Inc., 2018.

[19] D. M. W. Powers, "Applications and Explanations of Zipf's Law," 1998, Accessed: Aug. 16, 2020. [Online]. Available: https://www.aclweb.org/anthology/W98-1218.

[20] "Document (Stanford CoreNLP API)." https://nlp.stanford.edu/nlp/javadoc/javanlp-3.5.0/edu/stanford/nlp/dcoref/Document.html (accessed Aug. 17, 2020).

[21] C. J. Hutto and E. Gilbert, "VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text," May 2014, doi: null.

[22] "Visualizing Data using t-SNE | BibSonomy." https://www.bibsonomy.org/bibtex/28b9aebb404ad4a4c6a436ea413550b30/lopusz_kdd (accessed Apr. 23, 2020).

[23] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *jair*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[24] S. Wu and P. Flach, *A scored AUC Metric for Classifier Evaluation and Selection*. 2005.

[25] Z. C. Lipton, C. Elkan, and B. Narayanaswamy, "Thresholding Classifiers to Maximize F1 Score," *arXiv:1402.1892 [cs, stat]*, May 2014, Accessed: Aug. 17, 2020. [Online]. Available: http://arxiv.org/abs/1402.1892.

[26] C. T. C and S. Joseph, "A syntactic approach for aspect based opinion mining," in *Proceedings of the 2015 IEEE 9th International Conference on Semantic Computing (IEEE ICSC 2015)*, Feb. 2015, pp. 24–31, doi: 10.1109/ICOSC.2015.7050774.

[27] C. Braud and P. Denis, "Comparing Word Representations for Implicit Discourse Relation Classification," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Lisbon, Portugal, Sep. 2015, pp. 2201–2211, doi: 10.18653/v1/D15-1262.

[28] H.-Y. Chen and H.-H. Chen, "Implicit Polarity and Implicit Aspect Recognition in Opinion Mining," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, Berlin, Germany, Aug. 2016, pp. 20–25, doi: 10.18653/v1/P16-2004.

[29] E. H. Hajar and B. Mohammed, "Hybrid approach to extract adjectives for implicit aspect identification in opinion mining," in *2016 11th International Conference on Intelligent Systems: Theories and Applications (SITA)*, Oct. 2016, pp. 1–5, doi: 10.1109/SITA.2016.7772284.

[30] I. Cruz, A. Gelbukh, and G. Sidorov, "Implicit Aspect Indicator Extraction for Aspect-based Opinion Mining," p. 18.