1-2022

Analysing Natural Language Processing Techniques: A Comparative Study of NLTK, spaCy, BERT, and DistilBERT on Customer Query Datasets

Patrizia De Camillis
CCT College Dublin

Analysing Natural Language Processing Techniques: A Comparative Study of NLTK, spaCy, BERT, and DistilBERT on Customer Query Datasets

Patrizia De Camillis

A Thesis Submitted in Partial Fulfilment

of the requirements for the

Degree of

Master of Science in Data Analytics

January 2024

Supervisor: Dr Muhammad Iqbal

1.1 Table of Contents

Acronymns

A Lexicalized Domain Ontology and a Regularised Neural Attention model (ALDONAr)

Artificial Neural Network (ANN)

Aspect-based sentiment analysis (ABSA)

Bidirectional Encoder Representations from Transformers (BERT)

Credit Default Swaps (CDS)

Customer Relationship Management (CRM)

DistilBERT (distilled BERT)

Feedback Aspect-based sentiment analysis (FABSA)

Generative Pre-trained Transformer (GPT)

Harris Hawk (HH)

Hierarchical Attention Network (HAN)

Latent Semantic Analysis (LSA)

LatentDirichel Allocation (LDA)

Long Short -Term Memory (LSTM)

Natural Language Toolkit (NLTK)

Multinomial Naïve Bayes (MNB)

Natural Language Processing (NLP)

Ordinary Least Squares (OLS)

Parts-of-Speech (PoS)

Recurrent Neural Network (RNN)

Support Vector Machine (SVM)

Term Frequency Inverse Document Frequency (TF-IDF)

2 Abstract

Sentiment analysis within customer queries stems from its critical role in shaping the perception of a company's brand. Poor handling of customer queries may lead to adverse consequences. This paper explored and compared the performances of NLP models, including NLTK, spaCy, BERT and DistilBERT on a dataset comprising of customer queries and feedback. The study aimed to evaluate the accuracy and effectiveness of these diverse NLP models in analysing sentiment within customer communications.

The findings reveal distinct patterns among the models. BERT and DistilBERT exhibit greater similarity in their results, as do NLTK and spaCy. Notably, BERT and DistilBERT demonstrate a tendency to categorize queries as predominantly neutral, suggesting potential strengths in handling diverse customer sentiments. This analysis contributes valuable insights into the strengths and weaknesses of various NLP models.

3 Introduction

In today's customer focused environment, it is crucial to efficiently manage customers queries, feedback and expectations. A focus on NLP models offers unmatched opportunities for automating and improving sentiment analysis in customer data. This research compares four important NLP models: NLTK, spaCy, BERT, and DistilBERT, focusing on sentiment analysis in customer queries.

Accurate sentiment analysis in customer queries is crucial since it can immediately impact a company's reputation and brand image. Inadequately addressing consumer problems can lead to decreased customer satisfaction and brand loyalty. Understanding the intricacies of sentiment analysis models is essential for organisations looking to optimise their client interactions.

This paper evaluates and compares the outcomes of each NLP model. The findings show that BERT and DistilBERT produce similar results, while NLTK and spaCy follow a comparable pattern, prompting a closer look at the strengths and weaknesses of these models. Particularly, the prevalence of neutral sentiments in BERT and DistilBERT's analysed queries suggests potential advantages in capturing diverse customer sentiment. The subsequent sections delve into the methodologies, dataset, and a comprehensive discussion of the comparative results, offering valuable insights for businesses seeking effective sentiment analysis strategies in customer queries.

4 Research problem and objective

4.1 Research Overview

Sentiment analysis has become a crucial aspect of understanding customer feedback and opinions, providing valuable insights for businesses. This research aims to conduct a comprehensive comparison of sentiment analysis results on a dataset comprising customer queries. The study focuses on leveraging state-of-the-art NLP tools, including spaCy, NLTK, BERT, and DistilBERT to evaluate the polarity of each customer query.

The selected dataset consists of diverse customer queries, capturing a range of sentiments expressed by users. The goal is to assess the effectiveness and distinctions of sentiment analysis across these different NLP frameworks and models, shedding light on their respective strengths and limitations.

4.2 Research Objectives

4.2.1 Evaluate Performance

Assess the accuracy and efficiency of sentiment analysis using spaCy, NLTK, BERT, and DistilBERT on the customer query dataset.

Analyse the strengths and weaknesses of each tool in identifying sentiment polarity, considering factors such as context, complexity, and language nuances.

4.2.2 Compare Algorithmic Approaches

Compare the underlying algorithms and methodologies employed by spaCy, NLTK, BERT, and DistilBERT) in sentiment analysis.

Investigate how each approach handles sentiment classification tasks and understand their respective mechanisms.

4.2.3 Provide Comparative Insights

Offer a comprehensive comparison and contrast of sentiment analysis results obtained from each tool, highlighting their respective performances.

Draw conclusions on the most effective tool for sentiment analysis in the context of customer queries and identify potential areas for improvement.

Through this research, we aim to contribute valuable insights into the performance and suitability of various NLP tools for sentiment analysis in customer feedback scenarios, assisting businesses in making informed decisions based on the unique requirements of their datasets and applications.

5 Contribution

The contributions of this paper are the systematic evaluation and comparison of two NLP tools, spaCy and NLTK, and two deep learning models, BERT and DistilBERT, to determine the most suitable for sentiment analysis within the context of customer feedback and queries.

6 Sampling Strategy

6.1 Populations of Interest

6.1.1 Customer Queries Dataset

The primary population of interest is a dataset of customer queries sourced from a cloud-based customer service platform and saved in a csv format. This population is representative of written customer interactions, providing a representative sample of the various queries and concerns raised by customers. This population directly aligns with the research objectives, allowing for a focused analysis of the impact of NLP on query resolution and customer interactions.

The sampling method for the dataset will be simple random sampling – utilising a random sampling method to extract a representative subset of written customer queries from different categories or segments within the overall dataset. This is to ensure that each query in the dataset has an equal chance of being selected which will enhance the generalisability of findings to the entire population of customer queries. This method also minimises selection bias and ensures that a fair representation of queries is selected.

7 Primary Research, Methodology and Ethics

The chosen research methodology involved conducting interviews with five candidates that had been carefully selected. Two of the candidates could not be reached or communicated with and the remaining candidates were interviewed. These included one individual with expertise in deep learning and two end users of the sentiment analysis tools or services.

Among the reasons for choosing interviews as the insightful perspectives that could be gleaned. The interviews provided an opportunity to gather insights from experts who possess in-depth knowledge, experience and nuanced perspectives in the relevant fields. This allowed for a holistic understanding of the applications, challenges and benefits of sentiment analysis. Having engaged with experts, the research gained a better

understanding of specific challenges and requirements necessary for the implementation of sentiment analysis.

For the interview process, an email was dispatched to the interviewee, articulately detailing the interview's purpose and the intended use of the gathered information. In cases where recording was contemplated, explicit permission was sought in advance through email correspondence as a verifiable record. Participants were also be briefed on the dissemination of findings and informed of their potential access to the results.

The importance of training a deep learning model on multiple epochs and visualising the results was discussed by (Individual with expertise in deep learning, 2024). Both end users (End User 1, 2024; End User 2, 2024) emphasized the importance of sentiment on the customer relationship as well as brand perception. The sentiment within the customer queries that are received on their platform can be used to determine resourcing. If the queries are not dealt with adequately this can lead to customers using alternate public platforms where negative sentiment can directly influence the brand. So while the sentiment of the customer queries may not directly affect brand perception, if not dealt with satisfactorily the implications are have increased effects. They also spoke to the importance of sentiment analysis during sales periods. Because these are the busiest periods there is a lean towards more negative sentiment and there is a need to determine where these periods are and if the solutions are effective in reducing negative sentiment.

8 Timeframe and Supervisor Meetings

The project timeline was from 29 November 2023 – 23 February 2024.

The meetings with the supervisor were conducted on the dates of 06 December, 09 January, 24 January, 07 February, 13 February and 20 February. Additional communication was done via email.

Each meeting reviewed the status of the thesis and discussed the minimum necessary steps that needed to be taken for the following meeting. The meetings focused on the development of the code and specifically any machine learning. The continual building of the literature review included focus on using papers with high impact factor. The methodology for writing the thesis and the natural progression and evolution of the research objectives as the thesis progressed, and the architecture of the thesis to tie all these factors together.

9 Literature Review

Article	Ye	Models Used	Data Source	Ref
A 2-tuple fuzzy linguistic model for recommending health care services grounded on aspect-based sentiment analysis	2024	Fuzzy linguistic model	Careopinion	(Serrano-Guerrero et al., 2024)
A novel fusion-based deep learning model for sentiment analysis of COVID-19 tweets	2021	NBSVM CNN BiGRU fastText DistilBERT Meta Learner	Twitter Google Trends	(Basiri et al., 2021)
ALDONAr: A hybrid solution for sentence-level aspect-based sentiment analysis using a lexicalized domain ontology and a regularized neural attention model	2020	ALDONAr	Subtask 1 Restaurant Domain English Training Data Subtask 1 Restaurant Domain English Gold Annotations Data	(Meškelė and Frasincar, 2020)
Automated sentiment analysis in social media using Harris Hawks optimisation and deep learning techniques	2023	ABiLSTM	Sentiment140 Tweets Airline Tweets SemEval	(Halawani et al., 2023)
Deep Learning-based Sentiment Analysis: Establishing Customer Dimension as the Lifeblood of Business Management	2019	Sentiment Analysis	Various Social Media	(Agarwal, 2019)
Effect of Negation in Sentences on Sentiment Analysis and Polarity Detection	2021	Sentiment Analysis	Amazon reviews	(Mukherjee et al., 2021)
Enriching semantic knowledge bases for opinion mining in big data applications	2014		Amazon Datsets IMDB Datasets	(Weichselbraun, Gindl and Scharl, 2014)
Extending latent semantic analysis to manage its syntactic blindness	2020	Naïve Bayes	SNLI corpus Flickr corpus	(Suleman and Korkontzelos, 2020)
FABSA: An aspect-based sentiment analysis dataset of user reviews	2023	ABSA CNN GRU-GNN	Trustpilot Google Play Apple App Store	(Kontonatsios et al., 2023)
Impact of demography on linguistic aspects and readability of reviews and performances of sentiment classifiers	2022	Lexicon Based Approaches Classifiers	Yelp Dataset BanglaRestaurant	(Sazzed, 2022)
Machine learning-based proactive social-sensor service for mental health monitoring using twitter data	2022	Support Vector Machines (SVM) Long Short-Term Memory (LSTM)	User Summary dataset User Weekly Negativity dataset	(Hinduja et al., 2022)
Sentiment analysis for customer relationship management: an incremental learning approach	2020	Text Categorization	Company (Analist Group) data	(Capuano et al., 2020)
Sentiment analysis for customer review: Case study of Traveloka	2023	Classification Models	Twitter	(Diekson et al., 2023)
Sentiment analysis of COVID-19 cases in Greece using Twitter data	2023	Vader Sentiment Nkryst	Twitter ECDC	(Samaras, García-Barriocanal and Sicilia, 2023)
Sentiment analysis techniques, challenges, and opportunities: Urdu language-based analytical study	2022			
Sentiment Spin: Attacking Financial Sentiment with GPT-3	2023	FinBERT	Financial Phrase Bank	(Leippold, 2023)
Twitter-derived measures of sentiment towards minorities (2015–2016) and associations with low birth weight and preterm birth in the United States	2018	Regression Modelling	Twitter 2015 restricted Natality File	(Nguyen et al., 2018)
Understanding public engagement on twitter using topic modeling: The 2019 Ridgecrest earthquake case	2021	Linear Regression Latent Dirichlet Allocation(LDA)	Twitter	(Ahn, Son and Chung, 2021)

Table 1 Literature Review Overview - Papers, Publication Year, Models Employed, Dataset Source

9.1 Language

9.1.1 Sentiment analysis techniques, challenges, and opportunities: Urdu language-based analytical study

In the realm of sentiment analysis research, Liaqat et al. (2022) noted that while high-resource languages had seen effective applications, the scope for resource-poor languages, such as Urdu, remained limited due to insufficient resources such as lack of data tools and specialised language models. This discrepancy hindered non-high-resource languages from fully capitalising on the advancements in NLP and sentiment analysis, which played pivotal roles in organisational decision-making processes. The study aimed to bridge this gap by synthesising existing research on machine learning and deep learning approaches specifically tailored for Urdu-based sentiment analysis. Focusing on dimensions overlooked

in prior studies, the research systematically reviewed 40 selected articles using the methodology proposed by Brereton et al. (2007). This systematic literature review encompassed planning, conducting, and reporting the findings, with a meticulous query strategy involving specific keywords and predefined filters. The study assessed article quality, assigned a '1' to those discussing Urdu-based sentiment analysis approaches, and employed a snowballing approach for article selection.

9.2 Machine Learning

9.2.1 Machine learning-based proactive social-sensor service for mental health monitoring using twitter data

The study addressed the limitations of traditional health administration systems in the prevention and treatment of mental illness. To address these challenges the research study attempted to predict mental illness based on posts to social media, specifically using social sensor cloud services based on Twitter data. The motivation for using Twitter data for mental health monitoring is rooted in the platform's role as an ecosystem of social sensors, facilitating open discussions about mental illness and providing a wealth of personal health-related information.

Using the Twitter API the authors pulled a series of tweets. Within these tweets they looked for phrases relating to mental illness. Health experts were used to classify these tweets as truly genuine and related to the user's health. A crawler was run using Scrapy (Dimitrios Kouzis-Loukas, 2018) to obtain the history of tweets for the users whose posts related to their personal health issue.

The information was pre-processed. This was necessary as tweets oven contain acronyms, URL's, usernames, and repeating characters. Textblob as used to determine the polarity of the tweets which was translated into sentiment (positive, negative, or neutral). The data was classified using LSTM and they summarised into two datasets. 'User Summary' was created for supervised binary classification using traditional machine learning classification algorithms. 'User Weekly Negativity' was created based on the same sentiment analyser results for the purpose of doing sequence or time-series-based analysis.

The study explored traditional machine learning algorithms, such as SVM, and deep learning techniques, including LSTM to classify users as likely to be suffering from mental illness based on their Twitter activity. The results of the study indicated the accuracy and efficiency of the proposed LSTM-based proactive mental health detection framework, with an 84.31% test accuracy, outperformed traditional machine learning techniques.

The research study contributed by proposing a proactive approach to mental health monitoring using social media data and machine learning techniques. The implications for practice included the potential use of this framework for real-time mental health monitoring and surveillance, enabling early detection of mental illness and timely intervention. The study also highlighted the need for reliable data collection and addressed the challenges of using sentiment analysis and machine learning for mental health monitoring on social media platforms.

9.2.2 Sentiment analysis for customer review: Case study of Traveloka

The research conducted by Diekson et al. (2023) leveraged Twitter as a data source, focusing on tweets related to the company Traveloka. The Scikit-Learn library was employed to analyse the collected data using three classification models. Initial data acquisition used the Twitter API, with subsequent transformation into numerical form facilitated by a vectorizer. The dataset was then divided into training and test subsets and subjected to the three classification models to ascertain their respective accuracies. Following the acquisition of accuracy scores, the model exhibiting the highest accuracy was further scrutinised. The study revealed that the Support Vector Machine model demonstrated the highest accuracy. Additionally, a word cloud, excluding stopwords, was generated to highlight frequently used terms. The study's conclusion, drawn from the analysis of 1200 tweets, pointed to a positive sentiment towards Traveloka, particularly centred around positive perceptions related to promotions, campaigns, and discounts extended to users.

9.2.3 Deep Learning-based Sentiment Analysis: Establishing Customer Dimension as the Lifeblood of Business Management

Agarwal (2019) delved into the pivotal role of the 'customer dimension' in business scaling, emphasising the significance of assessing customer reviews for ensuring profitability. The research advocated for leveraging sentiment analysis as a paramount method to glean valuable insights from customer feedback, underscoring the importance of discerning discussed topics, customer intentions, and accurate classification. Sentiment analytics, powered by artificial intelligence and machine learning, emerged as a potent tool to extract meaningful understandings of customer emotions and intents.

The article critiqued the prevalent use of the bag of words model for sentiment analysis, highlighting its limitations. Proposing a remedy, the research advocated for the adoption of recurrent neural networks integrated with a deep learning model. The proposed methodology involved algorithm design and Python coding applied to two popular social media topics across multiple platforms within a specific period.

The study's findings illuminated how customers expressed their views and identified influential authors shaping public opinion. Distinct timelines for the initiation and subsequent decline of positive and negative sentiments in both topics were revealed, indicating sentiment evolution over time. The research theorised those ongoing discussions significantly impacted the footfall to corresponding events. While acknowledging the influential role of social media sentiment on event outcomes, the study posited that relying on only two classifications may be limiting. To achieve a deeper understanding of consumer sentiments, the research suggested incorporating multiple classifications for a more comprehensive analysis.

9.2.4 Sentiment Analysis for Customer Relationship Management: An Incremental Learning Approach

(Capuano et al., 2020) explored the integration of sentiment analysis into CRM systems. The authors introduced an Incremental Learning Approach based on a HAN architecture to address the limitations of traditional sentiment analysis methods in adapting to evolving customer sentiment patterns. This review provided a comprehensive overview, focusing on the proposed methodology's strengths, limitations, and contributions to the field of sentiment analysis in CRM.

Capuano et al. addressed challenges in traditional sentiment analysis methods within CRM, proposing an Incremental Learning Approach based on HAN to capture contextual and semantic information from customer communications. The model's adaptability over time, incorporating new data and feedback, was crucial for rapidly changing customer sentiment patterns. Experiments on a dataset of over 30,000 annotated customer communications in Italian and English demonstrated the effectiveness of the HAN model, outperforming traditional methods in sentiment analysis accuracy.

The proposed approach offered several advantages for CRM systems, including improved accuracy, adaptability to evolving customer sentiment, reduced reliance on manual labelling, and real-time analysis capabilities. Despite its contributions, the study suggested future research areas, such as exploring alternative HAN architectures, expanding the approach to other languages, integrating additional data modalities, and assessing the impact on specific business outcomes.

In summary, Capuano et al. presented a compelling approach to sentiment analysis in CRM, emphasizing the importance of adapting to evolving customer sentiment and leveraging

machine learning techniques to gain deeper insights from customer interactions. Regression Modelling

9.2.5 Twitter-derived measures of sentiment towards minorities (2015–2016) and associations with low birth weight and preterm birth in the United States

In this study the authors investigated the association between Twitter-derived measures of sentiment towards racial and ethnic minorities and birth outcomes in the United States. The study utilised Twitter's Streaming API to collect 1,249,653 tweets containing relevant keywords pertaining to minority groups and merged this data with information on all 2015 U.S. births, totalling 3.99 million. The results showed that mothers living in states with lower levels of positive sentiment towards racial/ethnic minorities had higher prevalences of low birth weight, very low birth weight, and preterm birth, compared to those living in states with higher levels of positive sentiment. This pattern was consistent for specific racial/ethnic groups such as Black people and Middle Eastern groups. The study also found that the associations between sentiment towards racial/ethnic minorities and birth outcomes were similar for both the full population and minority subgroups, indicating that negative sentiment may have adverse effects for all. The study highlights the potential impact of social media sentiment on birth outcomes and emphasises the need for a more inclusive and accepting social environment to reduce adverse birth outcomes, especially for minority groups.

The researchers collected Twitter data from March 2015 to April 2016, focusing on tweets containing keywords related to racial and ethnic minority groups. The study used various statistical analyses, including log Poisson regression models, to estimate associations between state-level sentiment and birth outcomes, controlling for individual-level maternal characteristics and state demographic characteristics. The findings revealed that lower state-level positive sentiment towards racial/ethnic minority groups was associated with higher rates of adverse birth outcomes, such as low birth weight and preterm birth. The study also included subgroup analyses to examine the associations among specific racial/ethnic groups, finding similar patterns across minority subgroups and the full population.

Overall, the research demonstrates the potential of social media data, specifically Twitter, in capturing sentiment towards racial and ethnic minorities and its impact on birth outcomes. The study provides valuable insights into the broader social environment's influence on health outcomes and emphasises the need for a more accepting and inclusive social climate to mitigate adverse birth outcomes, especially for minority populations. The study's rigorous methodology, including the use of Twitter data and statistical analyses, contributes to

understanding the complex relationship between social sentiment and health outcomes, calling for further research and action to address disparities in birth outcomes.

9.2.6 Understanding public engagement on twitter using topic modelling: The 2019 Ridgecrest earthquake case

The study focused on the use of social media for disseminating information during disasters and the interaction between organisations and the public during crisis situations. It examined the diverse communication strategies of organisations, governments, and scientists, aiming to verify the effectiveness of these strategies and whether they aligned with the public's needs. The study also used an automated machine learning technique for data collection and topic modelling to overcome the limitations of previous methodologies, contributing further to research on topic modelling.

The document also discussed the theoretical framework of uses and gratifications, examining the different patterns of media that individuals used based on their social and psychological needs and expectations during disasters. It also addressed the core concept of uses and gratifications, exploring the individual motivations for using social media during disasters (e.g., Orchard et al., 2014; Park et al., 2009; Urista et al., 2009).

Based on the previous research of different information sources communication strategies (e.g., (Boulianne, Minaker and Haney, 2018), (Takahashi, Tandoc and Carmichael, 2015)), the study classified tweets into government, media, nonprofit, research organisations, and "others." The text data was standardised and lemmatized using spaCy (https://spacy.io). Topic modelling was implemented to classify conversations based on word frequencies in each document and group those documents into smaller sets based on similarity (Blei, Ng and Jordan, 2003). The CountVectorizer function was used to transform documents into a matrix of words and their frequency. The optimal number of topics was determined using LDA function. The study aimed to understand the effects of topics generated by different organisations on public engagement during a natural disaster using the OLS approach.

The study used regression analysis to examine the impact of different topics discussed during natural disaster situations on public engagement, finding that topics from media outlets and "others" (Twitter users not included in the selected organisations) positively related to the number of retweets and favourites. The research questions focused on identifying typical tweet topics for several types of organisations and examining the public responses to each organisation's tweet topics, shedding light on the public's preferences and engagement patterns with several types of content during disaster situations.

In conclusion, the study's findings had theoretical and practical implications for the methodological development of Twitter research and the practice of communication during disasters. It underscored the importance of tailoring social media content to meet the public's needs and preferences during crisis situations, providing valuable insights for social media managers and organisations engaging with the public during disasters. The document also discussed the limitations of the study and suggested future research directions to further explore the relationship between organisation type, tweet topics, and public engagement.

9.3 Time Series

9.3.1 An alternative approach to predicting bank credit risk in Europe with Google data

This paper presents an alternative approach to predicting bank credit risk in Europe using Google data. (González-Fernández and González-Velasco, 2020) construct a sentiment index based on Google data to measure bank credit risk in European countries. The index captures investor sentiment related to bank credit risk, showing a great similarity to traditional indexes based on bank CDS. The out-of-sample analysis demonstrates that the sentiment index is helpful for predicting bank credit risk during periods of financial distress, as it enhances the accuracy of estimations. The paper fills a gap in literature by addressing the relationships among investor sentiment, Google data, and bank credit risk.

This paper aims to construct a Google-based sentiment index to measure bank credit risk. The index is based on the Europe banks sector 5Y CDS Index, which summarises European bank CDS data. The Google trends tool, Google search volume index, is used to approximate investor sentiment. The index is derived from a set of keywords related to bank credit risk, which are then refined and expanded. The index is then run backward rolling regressions to identify specific keywords that better reflect investor sentiment to bank credit risk. The results show that the sentiment index is related to CDS spreads as a measure of bank credit risk. The correlations between the sentiment indexes and bank credit risk measured through the Europe banks 5Y CDS Index range between 0.63 and 0.69, with all correlations being highly significant.

The study evaluates the effectiveness of sentiment indexes for predicting bank credit risk using regression models. The research uses the Europe banks sector 5Y CDS as a proxy for credit risk. The results show positive and significant coefficients for sentiment indexes in all models, except for model 3. The constructed Google-based sentiment indexes have some ability to predict bank credit risk. An out-of-sample evaluation analysis confirms this

observation. The correlations between the Europe banks sector 5Y CDS Index and the sentiment indexes constructed based on Google data are significant.

This paper investigates the use of a Google-based sentiment index to predict bank credit risk in Europe using European banks' CDS data. The researchers construct a sentiment index using Google searches that better reflect investor sentiment and evaluate its ability to predict bank credit risk through regression analysis and out-of-sample procedures. The results show that the sentiment index exhibits a high correlation to bank CDS and its inclusion in the regression analysis slightly enhances the estimations. The out-of-sample process indicates that the presence of the sentiment index does not significantly worsen the benchmark results, but it significantly enhances the forecast accuracy in periods of financial instability. These findings may have important implications for banks to assess credit risk, as Google data is easily available and more transparent than other alternatives for measuring investor sentiment.

This paper discusses the impact of news on European sovereign bond markets during the global financial crisis and the impact of news on bank credit risk. The study uses a benchmark model to evaluate the predictive accuracy of various models, including the sentiment index, sentiment index PCA, and sentiment index PCA. The results show that news sentiment and Google trends significantly influence bank credit risk, with the sentiment index being the most influential factor. The study also explores the role of media in the stock market and the role of credit default swaps in determining commodity futures prices. The research is supported by the Ministerio de Economía, Industria y Competitividad, Gobierno de España. The paper concludes that news sentiment and Google trends can help predict bank credit risk and provide insights into the impact of news on financial markets.

9.4 Sentiment Analysis

9.4.1 Impact of demography on linguistic aspects and readability of reviews and performances of sentiment classifiers

The study investigated how user demographics influenced the linguistic characteristics and readability of the reviews and the performance of sentiment classification methods. It considered that with the rising popularity of social media, there is an increase in user-generated content about multiple entities. Sentiment analysis, or opinion mining, is one of the most one of the most explored research problems in text mining (Mahdikhani, (2022); Neogi, Garg, Mishra, & Dwivedi, (2021); Obembe, Kolade, Obembe, Owoseni, & Mafimisebi,

(2021); Ridhwan & Hargreaves, (2021); Sazzed, (2020); 2022a; Sazzed & Jayarathna, (2019); (2021)).

Two datasets were used – the BanglaRestaurant dataset (Sazzed, (2021)) which was written by non-native English speakers residing in Bangladesh and the Yelp review dataset written by English native speakers residing in the USA. Differences were seen in linguistic attributes, sentence structure and vocabulary usage between the two datasets as revealed by the Mann-Whitney U test. Another test that was performed for readability, the Flesch Reading Ease, found no noticeable difference in the results. The different results with lexicon-based and machine learning-based sentiment classification methods indicate the role of demography on the performance of the sentiment classifiers.

A linguistic analysis was performed comparing word and sentence length, presence of negation words, articles, adjectives, verbs, prepositions, and the coverage of sentiment lexicons. The performance of several lexicon-based and machine-learning based classification methods were evaluated. TextBlob, VADER and LRSentiA exhibited performance like the machine learning based classifiers in the BanglaRestaurant dataset, and poor performances in the Yelp dataset compared to the BanglaRestaurant dataset.

The paper emphasised the importance of comprehending the linkage between expressed sentiment and linguistic facets of the reviews for better insights. Because sentiments and attitudes may vary across the demography, it is crucial to understand how the attributes of the reviews are affected by the diversity and demographics of the users.

9.4.2 Effect of Negation in Sentences on Sentiment Analysis and Polarity Detection

The paper applied sentiment analysis to understanding customer opinions in the realm of e-commerce and social media, where customer feedback is abundant and there is an overwhelming need to understand human sentiments. Analysing brand sentiments in social networks have provided a means for companies to evaluate their prospect in the market competition, which makes sentiment analysis a useful decision-making tool for companies. An essential aspect of the sentiment analysis is the identification of negation in written text. Amazon has a diverse product line and is one of the most comprehensive repositories of online reviews. This makes Amazon product reviews a favoured choice among the research community for developing NLP techniques.

The from the dataset was pre-processed through the steps of tokenization, stopwords removal, part of speech tagging and lemmatization. Four distinct types of negations were identified, including morphological, syntactical, double, and implicit negations. The

definitions of the different negation types and the process to identify the negation was used to design a pre-processing algorithm that aimed to identify words and phrases with explicit negation and replace the words with negation equivalent for their lemmatized form.

Feature extraction methodology was applied to convert the textual data to numerical data using TF-IDF. Four separate machine learning models were developed to investigate which model performs best on classifying the reviews sentiments – MNB, SVM, ANN and RNN. The best performing classifier was the combination of ANN classifier along with negation identification at 95.67% accuracy, marginally ahead of RNN with negation combination at 95.30%. Deep learning algorithms generalised well over the new dataset, and their performance remains the same while the performance of the traditional machine learning algorithms SVM and Naïve Bayes degrades. This means that the deep learning models were more stable in comparison with the traditional machine learning models.

The paper presented a robust framework for sentiment analysis, emphasising the significance of handling negations in written text for accurate sentiment classification. The findings underscored the potential to enhance sentiment analysis accuracy, particularly using deep learning models with negation identification.

9.4.3 Extending latent semantic analysis to manage its syntactic blindness

LSA is a widely used corpus-based approach that evaluates similarity of text based on the semantic relations among words – intuitively the model considers that words with similar meaning will occur in similar contexts, and it has been used successfully in a diverse range of NLP applications (Landauer, 2002; Vrana et al., 2018; Wegba et al., 2018; Jirasatjanukul et al., 2019) . The paper discussed the limitations of LSA in understanding the meaning of text because it ignores the structure of sentences - a syntactic blindness problem.

LSA suffered from inherent problems. Because it is based on the semantic relations between words it ignored syntactic composition of sentences, it did not consider the positions of subject and object of a verb as distinct, it considered lists of words as complete structures even with a lack of structure and it did not consider negation.

An extended version of LSA, xLSA, was proposed to address this issue, focusing on the syntactic structure of sentences. The xLSA algorithm leveraged Sentence Dependency Structures, and PoS tags to identify proper sentence structure, analyse semantic similarity, and handle negation. For tokenisation and PoS tagging the system uses the spaCy Tokenizer and the spaCy PoS tagger respectively (Honnibal and Johnson, 2015). This extension aimed

to overcome LSA's inability to differentiate between sentences with similar words but different meanings, as well as its neglect of sentence structure.

The study conducted four experiments comparing xLSA with LSA and deep learning-based techniques such as Google's Universal Sentence Encoder, BERT, and XLNET. The paper used sentences from two publicly available datasets, the SNLI corpus (Bowman et al., 2015) and the Flickr corpus (Young et al., 2014).

The results showed that xLSA outperformed LSA by providing more accurate semantic similarity scores, particularly in cases where sentences had similar words but different meanings, inverse sentence structures, and negation. In contrast, LSA tended to overlook these nuances and produce misleading similarity scores. Furthermore, xLSA was evaluated against deep learning-based models and demonstrated its effectiveness in handling syntactic and semantic aspects of sentences, outperforming these models in cases involving these complexities.

9.4.4 Enriching semantic knowledge bases for opinion mining in big data applications

The document presented a method for enhancing semantic knowledge bases to facilitate opinion mining in big data applications, focusing on contextualizing sentiment analysis. The approach involved identifying ambiguous sentiment terms, extracting context information, and grounding this contextual information to structured background knowledge sources such as ConceptNet and WordNet. The method was evaluated quantitatively and qualitatively, showing significant improvements in sentiment analysis accuracy and the successful disambiguation of ambiguous sentiment terms. The enrichment process extended the semantic knowledge bases with contextual information and concept knowledge, enhancing their coverage and adaptability to specific domains.

The paper introduced a novel method for contextualizing and enriching semantic knowledge bases for opinion mining in big data applications. It outlined the steps involved in this approach, including identifying ambiguous sentiment terms, extracting context information, and grounding this contextual information to structured background knowledge sources such as ConceptNet and WordNet. The evaluation of the method was conducted quantitatively and qualitatively, demonstrating significant improvements in sentiment analysis accuracy and successful disambiguation of ambiguous sentiment terms. The enrichment process extended the semantic knowledge bases with contextual information and concept knowledge, enhancing their coverage and adaptability to specific domains.

The document presented a method for contextualizing and enriching semantic knowledge bases for opinion mining in big data applications. The approach involved several steps, such as identifying ambiguous sentiment terms, extracting context information, and grounding this contextual information to structured background knowledge sources. The evaluation of the method demonstrated significant improvements in sentiment analysis accuracy and successful disambiguation of ambiguous sentiment terms. The enrichment process extended the semantic knowledge bases with contextual information and concept knowledge, enhancing their coverage and adaptability to specific domains.

9.5 Sentiment Spin: Attacking Financial Sentiment with GPT-3

This study investigated the vulnerability of financial sentiment analysis to adversarial attacks that manipulated financial texts. With the rise of AI readership in the financial sector, companies adapted their language and disclosures to fit AI processing better, leading to concerns about the potential for manipulation. The finance literature used keyword-based methods, such as dictionaries, for sentiment analysis. However, recent research suggested that these approaches might not always provide dependable or accurate results, particularly when applied to more complex or nuanced texts. The study highlighted the importance of addressing the potential vulnerabilities of algorithmic sentiment classification approaches to adversarial attacks, particularly as the algorithmic readership of financial texts increased.

The Financial Dictionary (Henry, 2008) and the LM dictionary (Loughran and McDonald, 2011) were used to analyse sentiment words in the finance domain. Deep learning models like FinBERT (Araci, 2019; Liu et al., 2020; Yang et al., 2020; Hazourli, 2022) and GPT-3 were used for text generation and sentiment analysis. An open-sourced database of human-annotated sentiments, the Financial Phrase Bank developed in Malo et al. (2013), was used to train the models. The goal was to turn negative sentiment into neutral or positive for adversarial attacks. The study focused on sentiment classification and used unlabelled data from earning calls for the second part.

Leippold (2023) used GPT-3, an advanced conversational AI model, to manipulate sentiment in financial news articles. The FinBERT-version of Araci (2019) was considered for experiments, as it outperformed other models. The author used two strategies to prompt GPT-3: generating a list of potential synonyms, filtering out suggested ones, and rephrasing the sentence. The disadvantage was that GPT-3 no longer had the context of the word for which it must generate the synonyms, but this method could reduce the cost of using the GPT-3 API.

This paper highlighted the capabilities of modern NLP methods, opening possibilities for developing more potent and sophisticated adversarial techniques. As the increasing reliance on AI-powered information processing amidst the ongoing information overload, exploring adversarial attacks and developing robust NLP models became a crucial area of research.

A 2-tuple fuzzy linguistic model for recommending health care services grounded on aspect-based sentiment analysis

The document included a study that assessed the quality of healthcare systems by analysing patient perceptions and attitudes. It highlighted the subjective factors that could complement objective factors in evaluating the quality of health care services. A multi-granular fuzzy language model was introduced to express patient opinions on many aspects of health care systems, with the goal of suggesting facilities based on user preferences. The model was tested on actual hospital feedback and surpassed other cutting-edge methods.

The challenges of evaluating the quality of health care and highlighted the importance of patient satisfaction and feedback in influencing their choices of medical institutions was highlighted. The role of the internet in enabling patients to share their opinions and the lack of platforms offering user feedback in the medical services domain was emphasized and a multi-granular fuzzy linguistic model for representing patient opinions about different features of health care systems was proposed, aiming to recommend hospitals based on user preferences. The approach was tested using real hospital opinions and was found to outperform other state-of-the-art approaches.

The methodology involved the classification of patient opinions into different aspects, the calculation of the polarity of each aspect, and the aggregation of these polarities using a weighted mean to compute the overall sentiment towards the hospital. The results were then ranked using the Promethee II algorithm. The proposed approach was compared with other fuzzy methods such as intuitionistic fuzzy and interval-valued Pythagorean fuzzy approaches, showcasing the superiority of the proposed model in ranking health care services according to patient priorities.

A novel multi-granular fuzzy linguistic model for evaluating health care services based on patient opinions was presented, which outperformed other state-of-the-art methods and demonstrated the significance of patient feedback in measuring the quality of health care systems. It offered valuable insights into the potential of sentiment analysis in the health care domain and provided a structured methodology for recommending hospitals based on patient preferences.

9.5.1 Sentiment analysis of COVID-19 cases in Greece using Twitter data

From 2004 to 2020, studies have used internet data to estimate and predict epidemics, with over half of the literature using social media messages for disease detection and prediction. Sentiment analysis in Twitter has gained interest, using knowledge-based systems and statistical machine algorithms. Recent research has focused on emotion models and artificial intelligence, as well as multilingual sentiment analysis.

Sentiment analysis uses a lexicon to identify polarity in text fragments, with popular tools like Vader Sentiment analysis tool and Nick Krystallis' Nkryst lexicon being used. Both lexicons can track negative, positive, and neutral sentiments, but only the second one can reveal specific sentiments and mood types due to its unique structure.

(Samaras, García-Barriocanal and Sicilia, 2023) used Twitter and the European Center for Disease Prevention and Control to analyse COVID-19 epidemiological data. Modified lexicons were used to track words in all grammatical types, including the term and sentimental ranking, called Polarity. The modified lexicon was based on the term theme or morpheme, which is the minimal component of the word with meaning or grammatical function. This method captured 109,121 words of the official Greek dictionary and 94,240 words for the translated Vader lexicon, capturing almost 40 times more than the original lexicon.

The analysis compared Vader and Nkryst lexicons, focusing on total polarity, positive and negative polarity, sentiments, statistical correlations, and technical measurement. Total polarity is the overall score calculated by both systems, while compound polarity is the overall score measured by the Vader system. The analysis also examined the relation between the statistical distributions of the pandemic and sentiment development within the same period of one year. The Vader system distinguished between fixed polarity and compound polarity, with the latter affecting the overall score. The analysis revealed that the Vader system measures a specific type of sentiment, while Nkryst measures the overall score.

The researchers found that negative sentiments dominate most in Vader and Nkryst, with surprise and disgust being the most frequent sentiments and sentiment analysis could capture around 100,000 words of the official Greek dictionary with different grammatical types or patterns. The study also explores the aggregation of semantic orientation and the concepts of unexpectedness and contradiction in expressions.

9.6 Deep Learning

9.6.1 ALDONAr: A hybrid solution for sentence-level aspect-based sentiment analysis using a lexicalized domain ontology and a regularised neural attention model

The article discussed the development of a hybrid model called ALDONAr for sentence-level aspect-based sentiment analysis which is an extension of their previous work on ALDONA (Meškelė & Frasincar, 2019). ALDONAr used a bidirectional context attention mechanism to measure the influence of each word in a sentence on an aspect's sentiment value and a classification module designed to manage the complex structure of a sentence. The model integrated a manually created lexicalized domain ontology to capture field-specific knowledge and achieved superior results on standard datasets compared to existing models.

The paper also presented a detailed methodology for ALDONAr, including the utilisation of a lexicalized domain ontology to extract domain-based knowledge, as well as the integration of a neural attention model extract an aspect's polarity value based on statistical relationships among an aspect and its context words. The neural attention model incorporated word embeddings, a bidirectional context attention mechanism, a sentence-level content attention module, and a classification module enhanced with two 1D CNN layers for additional flexibility in sentiment computations. The model's complexity was controlled using the dropout technique and L2 (Tikhonov) regularisation, and the objective function was defined using the cross-entropy loss.

A comparative analysis of ALDONAr with other benchmark models was provided, demonstrating its superior performance in terms of accuracy. The evaluation was based on two standard datasets, and ALDONAr outperformed other models, including the state-of-the-art CABASC, DBGRU, and ALDONA.

The development and methodology of ALDONAr, a hybrid solution for sentence-level aspect-based sentiment analysis, was demonstrated by its superior performance compared to existing models, highlighting its potential for accurate sentiment analysis in various contexts.

9.6.2 Automated sentiment analysis in social media using Harris Hawks optimisation and deep learning technique

The study by (Halawani et al., 2023) presented a new Harris Hawks Optimization with deep learning model for identifying and classifying sentiments in social media data. It processed raw social media text into a useful format and used fastText-based word embedding and skip-gram to reduce language processing dependency on data pre-processing. FastText is a fast, open-source, and efficient model that used skip-gram to inspect non-adjacent words

and skip some words. This technique was better than bag-of-word models in semantic-syntactic analogy tasks and was easier to implement, popular, and dependable.

The ABiLSTM model was used to effectively identify and categorize sentiments on social media. The LSTM control flow was like RNNs processing information and passing them forward. The Bi-LSTM model consisted of five gates: input gate, sequential gate, output gate, forget gate, and control gate. The ABiLSTM method used attention methods to provide different weights to words contributing to the emotion of a document. The weighted combination of all hidden states was used to estimate the last output in subsequent iterations.

The ABiLSTM model's hyperparameters were adjusted using the HH algorithm, a metaheuristic-optimized technique. HHs perched on arbitrary places, waited and monitored the desert for prey detection. Two perching approaches were dependent on the places of another family member and the prey or arbitrary tall trees, chosen according to an arbitrary q value. The HHO technique was a transition process in exploration to exploitation stages dependent on the escape energy of prey and altered the distinct exploitative performances as a function of the primary state of energy. During the exploitation stage, four distinct chase and attack approaches were presented based on the fundamental escape energy of the prey and chase styles of HHs. The escape energy parameter r was also employed for choosing the chase approach that represented the chance of prey from effectively escaping ($r<0.5$) or not ($r\geq0.5$) before attacks. The study showed extensive comparative results highlighted in the promising performance and was considered a valuable tool for the recognition and classification of sentiments.

9.6.3 A novel fusion-based deep learning model for sentiment analysis of COVID-19 tweets

The study presented a novel fusion-based deep learning model for sentiment analysis of COVID-19 related tweets from eight countries. The researchers highlighted the importance of understanding the sentiment of people expressed in their social media comments to monitor, control, and eradicate the disease. The proposed model involved a fusion of four deep learning and one classical supervised machine learning model, trained on a large, labelled dataset of tweets. Additionally, the study analysed coronavirus-related searches using Google Trends to better understand the change in sentiment patterns at separate times and places. The findings revealed that the coronavirus attracted the attention of people from different countries at separate times with varying intensities. The sentiment in their tweets was found to be correlated to the news and events that occurred in their respective countries, including the number of newly infected cases, number of recoveries,

and deaths. The researchers proposed that different social media platforms have a significant impact on raising people's awareness about the importance of the disease and promoting preventive measures among the community.

The study also provided insights into the dynamics of public responses to the COVID-19 pandemic, offering guidance to public health authorities to communicate effectively with people and provide public health responses to those most susceptible to the virus. The main contributions of the study included proposing a new fusion model for sentiment analysis of tweets by combining state-of-the-art transformer-based deep models, training and validating this model using large-scale Twitter dataset, and analysing coronavirus-related tweets and Google trend data for eight countries to find the sentiment of people in different time intervals and countries.

The researchers evaluated the proposed model's performance against several deep learning methods using the Stanford Sentiment140 dataset. The results showed that the proposed model outperformed other methods, demonstrating its effectiveness in analysing sentiment from tweets related to COVID-19. Additionally, the study provided detailed insights into the sentiment patterns and the impact of COVID-19 in different countries, shedding light on the varying levels of public sentiment and reactions to the disease across different regions. These findings contribute to a better understanding of public sentiment and the potential implications for public health policies and interventions.

In summary, the study provided a comprehensive analysis of sentiment patterns in COVID-19 related tweets from eight countries, offering valuable insights into the public response to the pandemic. The proposed fusion-based deep learning model demonstrated superior performance in sentiment analysis, providing a robust framework for understanding and monitoring public sentiment in the context of the global health crisis.

9.6.4 FABSA: An aspect-based sentiment analysis dataset of user reviews

ABSA is a method that extracts aspects of entities and classifies their polarity. FABSA is a new manually curated large-scale and multi-domain dataset of feedback reviews, consisting of approximately 10,500 reviews across 10 domains. Kontonatsios et al. (2023) conducted experiments to evaluate the performance of state-of-the-art deep learning models when applied to the FABSA dataset. The results showed that ABSA models can generalise across different domains when trained on the FABSA dataset, while the performance of the models is enhanced when using a larger training dataset.

The contributions include establishing a benchmark dataset, providing strong baseline models, and reporting experiments evaluating different dimensions of ABSA models. The FABSA dataset was constructed from three public data sources: Trustpilot, Google Play, and Apple App Store. The dataset is manually labelled against a hierarchical annotation scheme consisting of 7 parent and 12 child aspect categories, each associated with a sentiment label. A multi-label classification scheme is adopted, where each review is labelled with one or more aspect+sentiment labels.

Base methods include a logistic regression classifier using bag-of-words features and a Convolutional Neural Network classifier. The authors fine-tune six large language models, adopt two previously introduced ABSA methods, and fine-tune the GAS sequence generation model. They also apply preprocessing steps to the LogReg-BoW classifier and convert reviews into sparse document vectors for training a Logistic Regression classifier.

The GRU-CNN classifier is a model used to classify sentiment labels in short texts. It consists of four layers: a word embedding layer, a GRU, a CNN, and a fully connected layer. The GRU layer identifies long-range contextual features, while the CNN layer extracts local contextual features. Transformer models fine-tune encoder-only language models on the FABSA dataset. Single-sentence classification models adopt the same input and output representation, while sentence-pair classification models follow a sentence-pair classification approach. The FABSA dev dataset is used for identifying the best hyperparameter values for transformer models. The results of the ABSA models are reported after training and evaluation on the FABSA dataset.

The DeBERTa-pair-large model achieves the best F1 score performance, while the DeBERTa-single-large model yields the best recall. However, performance differences between the DeBERTa-single-large and DeBERTa-pair-large are statistically insignificant. The DeBERTa-pair-large model outperforms the BERT-single-large model by 2.1%. The large RoBERTa models also show significant performance improvements over their corresponding RoBERTa-base architectures. The DeBERTa-pair-large model yields substantial performance improvements over the DeBERTa-single-large when using a small training sample of 1,000 instances. The study concludes that large pre-trained transformer models obtain superior classification performance compared to other baseline ABSA models, but their performance decreases on rare aspect categories.

10 Research framework/model

10.1 Architecture

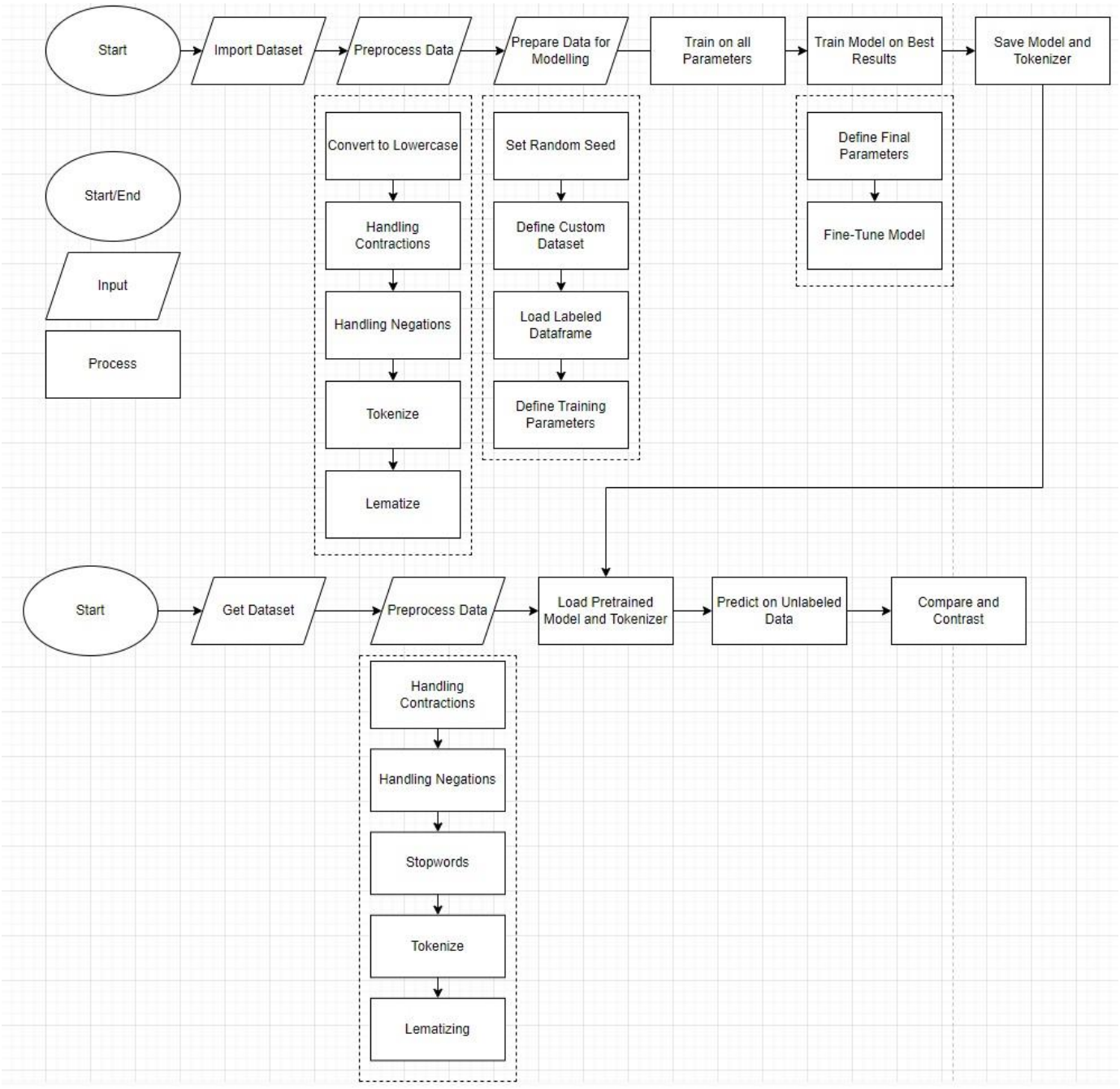

Fig. 1 Sentiment Analysis Thesis Architecture: Exploring Model Integration and Comparative Evaluation: The models are trained and saved separately

10.2 Exploratory Data Analysis

The shape of the dataframe is (79691, 2), indicating that it has 79,691 rows and 2 columns, i.e. ‘created_at’ and ‘description’. The information provided by the ‘.info()’ method revealed details about the dataframe structure. It specified that both columns had 79,691 non-null entries of object data type. Additionally, it highlighted the presence of 64 duplicated rows within the dataframe.

Further analysis involved the removal of duplicates, resulting in a total of 79,627 rows in the dataframe. The examination of null values showed that 0.0% of the dataframe contained NaN values. Any null values present were replaced during anonymisation, ensuring a more complete and consistent dataset. These actions and observations contribute to understanding the data's structure, cleanliness, and necessary pre-processing steps taken.

10.3 Preprocess Text

10.3.1 Eliminating Rows with Non-Evaluative Content

The examination of the dataframe's tail revealed a pattern in certain rows where the description simply contained the text 'conversation with <name> url: none'. Recognising that these sentences were neutral in nature and held little value for sentiment analysis evaluation, the decision was made to exclude them from the dataset. To achieve this the rows matching the specified condition were filtered out using Boolean indexing. The condition checked if the 'description' column started with 'conversation with' followed by any name and ends with 'none'. This process effectively removed the 2822 identified neutral sentences, changing the dimensions to (76805, 2).

10.3.2 Language

Within the dataset, entries encompass textual content in various languages. The goal is to conduct sentiment analysis exclusively on English text to facilitate application and ensure the uniform application of NLP techniques that rely on language-specific operations like tokenization. Two distinct methods were employed to identify the language of the 'description' column, utilising separate language detection libraries.

The langid library was employed first. Langid is a standalone Language Identification (LangID) tool which is currently trained on 97 languages (saffsd, 2021). A function named ‘detect_language’ was defined to classify the language using langid's ‘classify’ method. This function was then applied to the 'description' column, and the detected language was stored in a new 'language' column in the dataframe.

The langdetect library was used as an alternate option, this is a direct port of Google's language-detection library from Java to Python (Danilák, 2024). A function named ‘detect_language’ was defined using the langdetect library's ‘detect’ method for language detection. As with the previous library, the language detection function was applied to the 'description' column, and the detected language was stored in a different column named 'language2'.

The dataframe was subjected to two consecutive filtering operations based on the 'language' and 'language2' columns. The first operation created a new dataframe named 'different_values_df' by retaining only those rows where the values in the 'language' and 'language2' columns were different. The resulting dataframe 'different_values_df' contains these rows, displaying the instances where the language detection methods produced different results. Following this, the dataframe was filtered to include only rows where both 'language' and 'language2' are labelled as English ('en').

Despite the use of two language detection libraries, it was observed that certain instances existed where the rows were labelled as 'en' but contained text in a language other than English. Conversely, there were rows labelled with a language other than English, yet the ‘description’ row was in English. These observations indicate potential challenges or complexities in accurately identifying the language of the 'description' column, highlighting the need for further investigation or refinement in the language detection process.

As defined by Siino, Tinnirello and Marco La Cascia (2024): ‘Preprocessing, in this sense, can be summed up as the process of cleaning and preparing texts that will be tokenized for subsequent operations. Thus, the necessity for data cleaning and normalisation arises because the effectiveness of a model employed after the preprocessing stage depends critically on the quality of such data’.

Preprocessing text is a crucial step in NLP models for several reasons. Firstly, it helps standardise and clean the input data, ensuring consistency and reducing variability. Text data often contains noise, such as special characters, punctuation, or irrelevant information, which can adversely affect the model's performance. Preprocessing involves tasks like tokenization, removing stopwords, and stemming or lemmatization, which collectively streamline the text data, making it more manageable for the model to understand and extract meaningful patterns.

Secondly, preprocessing addresses the challenge of dealing with different linguistic variations. Languages contain various forms of words, such as verb conjugations or plural

forms, which can lead to an explosion of feature dimensions in the model. Techniques like lemmatization, which reduces words to their base or root forms, help mitigate this issue by consolidating related word forms, ultimately reducing the dimensionality of the input space.

Moreover, preprocessing aids in enhancing the model's generalisation by reducing redundancy and emphasising essential semantic information. By removing irrelevant or redundant terms, the model becomes more focused on capturing the underlying semantics of the text, making it more robust to variations in phrasing or wording.

10.4 Natural Language Process Preprocessing

Due to the sensitivity of the dataset, the details of the preprocess for anonymisation cannot be provided. However, several crucial steps were undertaken during the anonymisation process, which would typically have been included in the preprocessing phase. The dataframe was lowercased, where all text was converted to lowercase to ensure uniformity. This ensured that all necessary words were uniformly treated and removed.

Another significant step was how numbers were handled, with a decision made on whether to keep or remove them. In sentiment analysis, numerical values might not always contribute to sentiment, so the choice might be made to remove them. In this case, letter combinations were replaced with the letter 'x', particularly to remove phone numbers. Additionally, all email addresses and websites were completely removed from the text.

Customisation based on the domain or dataset was also considered. Any words related to the company name were removed, and first names were eliminated. These meticulous steps were taken to maintain the confidentiality and sensitivity of the dataset while ensuring that the text was appropriately processed for subsequent analysis.

10.4.1 Handling Contractions

A function called ‘expand_contractions’ was defined which takes text as input and uses the contractions library (Kooten, 2023) to expand contractions within the text. This library is ‘capable of resolving contractions (and slang)‘ (Kooten, 2023). Contractions are shortened forms of words or phrases, such as "we're" for "we are."

In the field of NLP, there are two primary reasons for addressing contractions. Firstly, computers do not inherently recognise that contractions represent abbreviations for specific word sequences. For example, "we're" and "we are" may be treated as distinct entities by a computer, despite having the same meaning. Secondly, contractions ‘contribute to the increase in dimensionality of the document-term matrix’ (Lukei, 2019), resulting in a more

computationally expensive process. (ScienceDirect, 2014) describes the document-term matrix as 'a matrix describing the frequencies of all terms occurring in the collection of text documents', and the expansion of contractions leads to an increase in frequencies, further complicating computations.

10.4.2 Handling Negations

According to (Mukherjee et al., 2021) 'an essential aspect of sentiment analysis is the identification of negation in written text. The presence of the word negation can change the polarity of the text, and if not handled properly, it will affect the performance of the sentiment classification'.

A function named 'handle_negations' was defined to process text data by identifying and handling negations. It created a set of negation words, such as "not", "no", "never", etc. The function then tokenized the input text and iterated through the tokens. When a token matched one of the defined negation words in a case-insensitive manner, it marked the subsequent tokens in the same sentence by adding a "not_" prefix to each of them until a punctuation mark (such as '.', ',', ';', '!', or '?') is encountered.

The resulting modified text was stored in a new column called 'modified_description'. Negations in the text data were recognised and marked, providing a modified version of the original text that reflects the presence of negations for the sentiment analysis.

10.4.3 Remove Stopwords, Tokenize, Lemmatize

Frequently used English words such as "an" and "the" are called stop words. These words are not particularly useful in NLP as they don't carry any content of their own and as they are therefore removed. Tokenization involves breaking text into individual words or tokens while lemmatization reduces words to their base form. Lemmatization assisted in capturing the essence of words - for example, the verb "walk" might appear as "walking," "walks" or "walked." Inflectional endings such as "s," "ed" and "ing" are removed. Lemmatization groups these words as its lemma, "walk". Stemming can also be used for this, but lemmatization tends to be more accurate (Saumyab271, 2022).

10.4.3.1 NLTK

NLTK library (NLTK, 2023) was used to preprocess text data, employing various techniques to enhance the quality of the text for further analysis. Initially, a WordNetLemmatizer was initialised to facilitate lemmatization. A function named 'preprocess_text' was created to take text as input and performs several preprocessing steps.

The text was tokenized using NLTK's 'word_tokenize' function. Stopwords and punctuation were then removed from the tokenized list, with stopwords defined using NLTK's set of English stopwords. The lemmatization process was then applied to the remaining alphanumeric tokens using the WordNetLemmatizer.

The preprocessing function was applied to the column 'modified_description' and stored in a new column called 'nltk_preprocessed_text'.

10.4.3.2 spaCy

The text preprocessed using the spaCy library, loaded the English language model ("en_core_web_sm"). A function named 'preprocess_text' was defined to process text data. Within the function, the spaCy model was applied to the input text, creating a spaCy Doc object.

Lemmatized tokens were then extracted from the spaCy Doc through the function, filtering out non-alphabetic tokens and stopwords. Each token's lemma was retrieved using the "lemma_" attribute, and the presence of alphabetic characters and non-stopword status was verified using the "is_alpha" and "is_stop" attributes. The resulting lemmatized tokens were joined into a string.

The preprocessing function was applied to the column 'modified_description' and stored in a new column called 'spacy_preprocessed_text.'

10.4.4 Sentiment Analysis

10.4.4.1 NLTK

Sentiment analysis was performed on the preprocessed column using the NLTK library, specifically utilising the SentimentIntensityAnalyzer class. A SentimentIntensityAnalyzer object, assigned to the variable 'sia', was created. A function named 'sentiment_analysis_nltk_binary' was defined to conduct binary sentiment analysis on a given text.

Within the function, the SentimentIntensityAnalyzer was employed to calculate the compound sentiment score for the input text using the 'polarity_scores' method. The compound score represents the overall sentiment polarity, ranging from -1 (most negative) to 1 (most positive). The function then returns 1 if the compound score was greater than or equal to 0.05, indicating a positive sentiment, and 0 otherwise, representing a negative or neutral sentiment. The function was applied to the 'nltk_preprocessed_text' column and the resulting polarity scores were stored in a new column called 'sentiment_nltk_binary'.

10.4.4.2 spaCy

Sentiment analysis was performed on the preprocessed text data using the spaCy library. A spaCy pipeline was loaded with a blank English model, and two components were added to the pipeline: the 'sentencizer' and 'asent_en_v1', which is a spaCy component for aspect-based sentiment analysis.

A function named ‘sentiment_analysis_spacy_binary’ was defined to perform sentiment analysis on the text using the spaCy pipeline. The function processes the input text using the loaded spaCy pipeline, and the sentiment polarities are obtained from the 'polarity' attribute of the document. The resulting sentiment polarities are then stored in a new column named 'sentiment_polarity' in the 'sentiment' dataframe.

Following this, the compound scores are extracted directly from the 'sentiment_polarity' column, and a threshold for positive sentiment is set at 0.05. A new column named 'compound' is created to store the compound scores. Finally, the 'compound' scores are converted into binary labels based on the specified threshold, and the binary labels are stored in a column named ‘sentiment_spacy_binary'.

10.5 Neural Network Models

10.5.1 Choosing NN Models

BERT and DistilBERT are both NLP models based on transformer architectures, but they differ in terms of their scale and computational efficiency.

BERT, introduced by Google in 2018 (Devlin et al., 2018), is a large-scale model that incorporates bidirectional context to capture intricate relationships within text. It is pre-trained on a massive corpus of data and can subsequently be fine-tuned for various NLP tasks. BERT's strength lies in its ability to understand context comprehensively, but its size makes it computationally intensive and resource demanding.

On the other hand, DistilBERT, introduced by Hugging Face in 2019 (Sanh et al., 2019) , is a distilled version of BERT designed to be more lightweight and efficient. It retains the bidirectional context understanding but reduces the number of parameters, making it faster and requiring less computational resources. DistilBERT achieves a balance between model size and performance, making it suitable for scenarios where computational efficiency is a priority.

10.5.2 Import Dataset

A labelled dataset (Kotzias,Dimitrios. (2015). Sentiment Labelled Sentences. UCI Machine Learning Repository. https://doi.org/10.24432/C57604.), known as the "Sentiment Labelled Sentences" dataset, was used to train the model on. This dataset is intended for sentiment analysis tasks. The dataset consists of labelled sentences, where each sentence is associated with a sentiment label indicating whether the sentiment expressed in the sentence is positive or negative and is licensed under a Creative Commons Attribution 4.0 International (CC BY 4.0) license.

The dataset is divided into three sub-datasets, each originating from different sources: Amazon customer reviews, Yelp reviews, and IMDb movie reviews. These three sub-datasets are stored in separate text files and each sentence is labelled with binary sentiment values, 0 for negative or neutral sentiment and 1 for positive sentiment.

The data consisted of three tab-separated text files which were loaded and concatenated into a single dataframe using the pandas library (The pandas development team, 2020) with a continuous index. The text files columns were renamed to indicate the value that they held.

10.5.3 Preprocess Labelled Text

To ensure uniformity between the labelled and unlabelled datasets, the entire labelled dataset is converted to lowercase by using the 'applymap' function which applies the specified lambda function to each element in the dataframe. The lambda converts an element to string after confirmation that the string is lowercase. The techniques to handle both contractions and negations that were used on the labelled dataset are applied to the unlabelled dataset ensuring that the preprocessing of text is consistent.

10.5.4 Set a random seed for reproducibility

Setting a random seed is crucial in machine learning and it 'ensures consistency between different runs. It also allows you to reproduce errors and other people to reproduce your results. (Huyen, 2022). It ensures that, given the same initial conditions and parameters, the random processes within these libraries produce the same results each time they are run. The 'set_seed' function allows the user to control the randomness and obtain consistent results across different runs and different libraries.

10.5.5 Define your custom dataset

'torch.utils.data.Dataset' is an abstract class representing a dataset which is used to facilitate the creation of PyTorch-compatible datasets for NLP tasks. A custom dataset class

named ‘CustomDataset’ was defined to inherit ‘torch.utils.data.Dataset’, and ‘override the methods __len__ so that len(dataset) returns the size of the dataset and __getitem__ to support the indexing such that the dataset can be used to get ith sample’ (PyTorch, n.d.). Inside the ‘__getitem__’ method, the input text at the specified index is tokenized using the provided tokenizer. The tokenization includes truncation and padding, ensuring uniform sequence length. The resulting tokenized input is then returned as a dictionary containing 'input_ids' (flattened), 'attention_mask' (flattened), and the corresponding label as a PyTorch tensor. The constructor ‘__init__’ method of the class takes four parameters: ‘texts’ representing the input texts, ‘labels’ for the corresponding labels, ‘tokenizer’ for tokenization, and an optional parameter ‘max_length’ to set the maximum sequence length.

10.5.6 Load your labelled dataframe

The data from the labelled dataframe was loaded into two separate Python lists, and assigned to the variables texts and labels, corresponding to their column names. Because BERT models typically require input data in the form of tokenized texts and corresponding labels, this process serves the purpose of organising the data in a format suitable for training the model. By converting the ‘text’ column to a list of texts, the subsequent tokenization process is facilitated.

In the training process of a BERT model, the use of a train-test split is essential for evaluating the model’s performance and preventing overfitting. The dataset was split into training and validation sets with a test size of 0.2, ensuring that a portion of the data is reserved for validation. The training set was used to train the models parameters while the validation set allowed monitoring on its performance on data it had not been exposed to during training. This helps to identify potential issues like overfitting, where the model memorises the training data but fails to generalise to new examples. This is a fundamental step in ensuring reliability and effectiveness of a BERT model for NLP tasks.

The tokenizer for the model was initialised. The tokenizers used are the ‘BertTokenizer’ class and ‘DistilBertTokenizer’ class from the Hugging Face transformers library. The tokenizer is created with the ‘from_pretrained’ method, specifying the 'bert-base-uncased' and 'distilbert-base-uncased' models.

The ‘base-uncased' model is a pre-trained BERT model that has been trained on a large corpus of text data. It was first introduced by Devlin et al. (2018). The 'uncased' variant indicates that the model is case-insensitive, meaning it treats uppercase and lowercase

letters as equivalent. This model is a widely used and well-established variant of the models, providing a strong baseline for various natural language processing tasks.

The role of the tokenizer is to preprocess and tokenize input text into a format suitable for BERT model input. It breaks down the text into subword or word-level tokens, adds special tokens (such as [CLS] for classification and [SEP] for separation), and converts the tokens into numerical representations that can be fed into the neural network.

10.5.7 Define training parameters

The parameters for training the model are defined and remain the same between both the BERT and DistilBERT models. Three key parameters are specified: batch size, learning rate and epoch.

According to (Ozdemir, 2023) 'the learning rate determines the step size of the model's weight updates, while the batch size refers to the number of training examples used in a single update. The number of epochs denotes how many times the model will iterate over the entire training dataset.'

The parameters were influenced by (Devlin et al., 2018) where it was found that although 'the optimal hyperparameter values are task-specific…the following range of possible values to work well across all tasks': Batch sizes of 16 or 32, Learning rate (using an AdamW optimiser) of 5e-5, 3e-5, or 2e-5 and Number of epochs 2, 3, or 4.

The loops for the BERT and DistilBERT models were executed in 6h 11m and 3h 11m respectively, indicating a 3 hour difference.

An empty dataframe named 'results_df' was created using the pandas library (pandas, 2018). The dataframe stored and organised the results of training both the BERT and DistilBERT models across the various iterations with each row representing the results obtained from a specific parameter configuration. The results in their respective dataframes are presented in Table 1 and Table 2.

	Epoch	Batch Size	Learning Rate	Seeds Used	Train Loss	Val Accuracy	Model
0	1	16	0.00005	2147483648	0.325437	0.930909	BERT
1	2	16	0.00005	2147483648	0.314238	0.927273	BERT
2	3	16	0.00005	2147483648	0.328250	0.945455	BERT
3	4	16	0.00005	2147483648	0.328673	0.949091	BERT
4	5	16	0.00005	2147483648	0.319029	0.943636	BERT
5	1	16	0.00003	2147483648	0.339349	0.952727	BERT
6	2	16	0.00003	2147483648	0.336808	0.943636	BERT
7	3	16	0.00003	2147483648	0.329735	0.949091	BERT
8	4	16	0.00003	2147483648	0.320888	0.954545	BERT
9	5	16	0.00003	2147483648	0.314941	0.932727	BERT
10	1	16	0.00002	2147483648	0.360117	0.956364	BERT
11	2	16	0.00002	2147483648	0.349788	0.947273	BERT
12	3	16	0.00002	2147483648	0.355272	0.952727	BERT
13	4	16	0.00002	2147483648	0.352019	0.952727	BERT
14	5	16	0.00002	2147483648	0.325361	0.929091	BERT
15	1	32	0.00005	2147483648	0.345364	0.949091	BERT
16	2	32	0.00005	2147483648	0.359487	0.936364	BERT
17	3	32	0.00005	2147483648	0.362033	0.943636	BERT
18	4	32	0.00005	2147483648	0.332453	0.941818	BERT
19	5	32	0.00005	2147483648	0.354859	0.932727	BERT
20	1	32	0.00003	2147483648	0.383935	0.945455	BERT
21	2	32	0.00003	2147483648	0.362989	0.930909	BERT
22	3	32	0.00003	2147483648	0.375167	0.949091	BERT
23	4	32	0.00003	2147483648	0.377124	0.936364	BERT
24	5	32	0.00003	2147483648	0.352360	0.936364	BERT
25	1	32	0.00002	2147483648	0.403606	0.954545	BERT
26	2	32	0.00002	2147483648	0.393336	0.943636	BERT
27	3	32	0.00002	2147483648	0.421302	0.945455	BERT
28	4	32	0.00002	2147483648	0.414784	0.929091	BERT
29	5	32	0.00002	2147483648	0.395206	0.941818	BERT

Table 2 Overview of the performance of the BERT model across various hyperparameter settings

	Epoch	Batch Size	Learning Rate	Seeds Used	Train Loss	Val Accuracy	Model
0	1	16	0.00005	2147483648	0.308052	0.925455	DistilBERT
1	2	16	0.00005	2147483648	0.312383	0.940000	DistilBERT
2	3	16	0.00005	2147483648	0.307658	0.941818	DistilBERT
3	4	16	0.00005	2147483648	0.336024	0.938182	DistilBERT
4	5	16	0.00005	2147483648	0.309528	0.938182	DistilBERT
5	1	16	0.00003	2147483648	0.332779	0.938182	DistilBERT
6	2	16	0.00003	2147483648	0.333633	0.943636	DistilBERT
7	3	16	0.00003	2147483648	0.325204	0.943636	DistilBERT
8	4	16	0.00003	2147483648	0.347166	0.934545	DistilBERT
9	5	16	0.00003	2147483648	0.317752	0.936364	DistilBERT
10	1	16	0.00002	2147483648	0.360377	0.938182	DistilBERT
11	2	16	0.00002	2147483648	0.352040	0.938182	DistilBERT
12	3	16	0.00002	2147483648	0.350005	0.938182	DistilBERT
13	4	16	0.00002	2147483648	0.365804	0.930909	DistilBERT
14	5	16	0.00002	2147483648	0.347183	0.930909	DistilBERT
15	1	32	0.00005	2147483648	0.335974	0.930909	DistilBERT
16	2	32	0.00005	2147483648	0.348147	0.934545	DistilBERT
17	3	32	0.00005	2147483648	0.337022	0.905455	DistilBERT
18	4	32	0.00005	2147483648	0.361858	0.940000	DistilBERT
19	5	32	0.00005	2147483648	0.337575	0.934545	DistilBERT
20	1	32	0.00003	2147483648	0.382866	0.940000	DistilBERT
21	2	32	0.00003	2147483648	0.376241	0.932727	DistilBERT
22	3	32	0.00003	2147483648	0.367105	0.925455	DistilBERT
23	4	32	0.00003	2147483648	0.388295	0.929091	DistilBERT
24	5	32	0.00003	2147483648	0.367987	0.929091	DistilBERT
25	1	32	0.00002	2147483648	0.436538	0.934545	DistilBERT
26	2	32	0.00002	2147483648	0.422213	0.930909	DistilBERT
27	3	32	0.00002	2147483648	0.406769	0.920000	DistilBERT
28	4	32	0.00002	2147483648	0.427547	0.930909	DistilBERT
29	5	32	0.00002	2147483648	0.408878	0.925455	DistilBERT

Table 3 Overview of the performance of the DistilBERT model across various hyperparameter settings

10.5.8 Fine-tune the model on labelled data

A nested training loop iterated over the different parameter combinations for fine-tuning that were saved in the variables. With each iteration a model was initialised for sequence classification using the pre-trained model. The AdamW optimiser was employed, and a linear learning rate scheduler with warm-up was defined.

The training loop iterated through the training dataloader, computing and backpropagating the loss to update the model's parameters. The process was repeated for the specified number of epochs. The trained model was evaluated on a separate validation dataset, and the accuracy and loss were calculated. The results were appended to the 'results_df' dataframe, which was continually updated for each hyperparameter combination, allowing for comprehensive tracking of training outcomes.

The results, including epoch number, batch size, learning rate, seed used for reproducibility, average training loss, validation accuracy, and the model identifier ('bert-uncased'), are appended to a results dataframe ('results_df'). This dataframe is continuously updated for each hyperparameter combination, facilitating a comprehensive record of the model's performance across different configurations. The BERT and DistilBERT models took 6h 11m and 3h 11m respectively to loop through the variables.

To fine-tune the model efficiently, the optimal hyperparameters from identifying the highest validation accuracy were selected. The selected hyperparameters, including batch size, learning rate, and seed used, were then used to define the training parameters for the subsequent model fine-tuning process. The BERT model had the highest accuracy of 0.956364 with 1 epoch, a batch size of 16 and a learning rate of 2e-5 and the lowest minimum loss of 0.314238 for this model was achieved with 2 epochs, a batch size of 16 and a learning rate of 5e-5. The DistilBERT model had the highest accuracy of 0.943636 with 2 epochs, a batch size of 16 and a learning rate of 3e-5 and the lowest minimum loss of 0.307658 for this model was achieved with 3 epochs, a batch size of 16 and a learning rate of 5e-5. Although the training loop was designed to explore various hyperparameter combinations, the model that the loop created with the optimal results proved to be computationally intensive for the machine. Consequently, it was decided to prioritise efficiency and the choice was made to utilise the best performing hyperparameters to train a model on the labelled dataset. This approach allowed a balance to be struck between achieving high performance and avoiding computational resource constraints.

```
Maximum Validation:  Epoch                      1
Batch Size                 16
Learning Rate         0.00002
Seeds Used         2147483648
Train Loss           0.360117
Val Accuracy         0.956364
Model                     NaN
model_BERT       BERT-uncased
Name: 10, dtype: object
Mimimum Loss:  Epoch                      2
Batch Size                 16
Learning Rate         0.00005
Seeds Used         2147483648
Train Loss           0.314238
Val Accuracy         0.927273
Model                     NaN
model_BERT       BERT-uncased
Name: 1, dtype: object
```

Fig. 2. Optimal Hyperparameter Configurations for BERT Model Training

```
Maximum Validation:  Epoch                      2
Batch Size                 16
Learning Rate         0.00003
Seeds Used         2147483648
Train Loss           0.333633
Val Accuracy         0.943636
Model               DistilBERT
Name: 6, dtype: object
Mimimum Loss:  Epoch                      3
Batch Size                 16
Learning Rate         0.00005
Seeds Used         2147483648
Train Loss           0.307658
Val Accuracy         0.941818
Model               DistilBERT
Name: 2, dtype: object
```

Fig. 3. Optimal Hyperparameter Configurations for DistilBERT Model Training

The 'save_pretrained' method was used on both model and tokenizer instances. This ensured that the pretrained models and tokenizers were saved independently allowing for later retrieval and usage without the need to retrain or recreate the model. The path files indicate where the model weights, configuration and tokenizer vocabulary are stored.

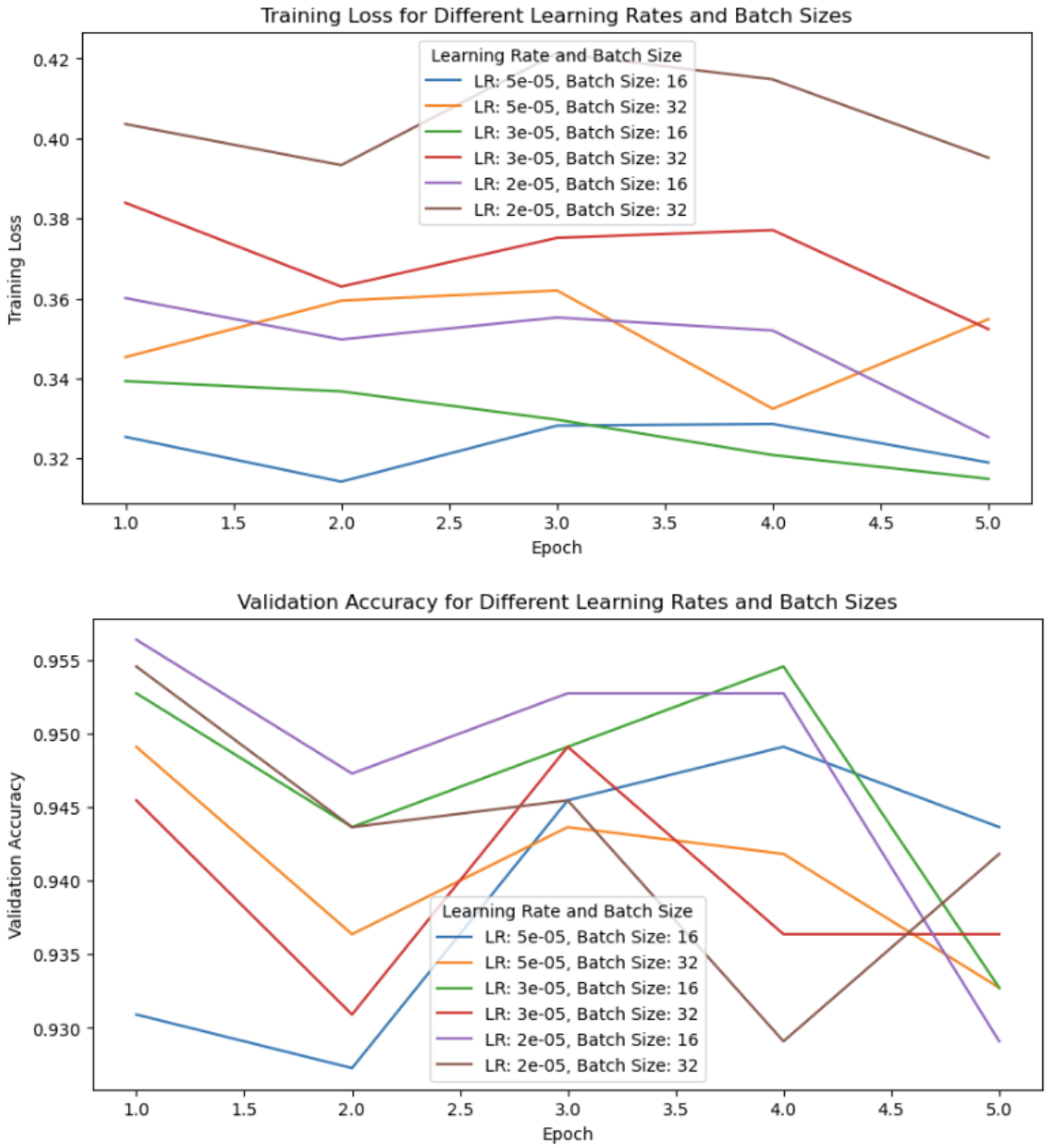

Fig. 4. Comparative Analysis of Training Loss and Validation Accuracy Across Learning Rates and Batch Sizes for BERT

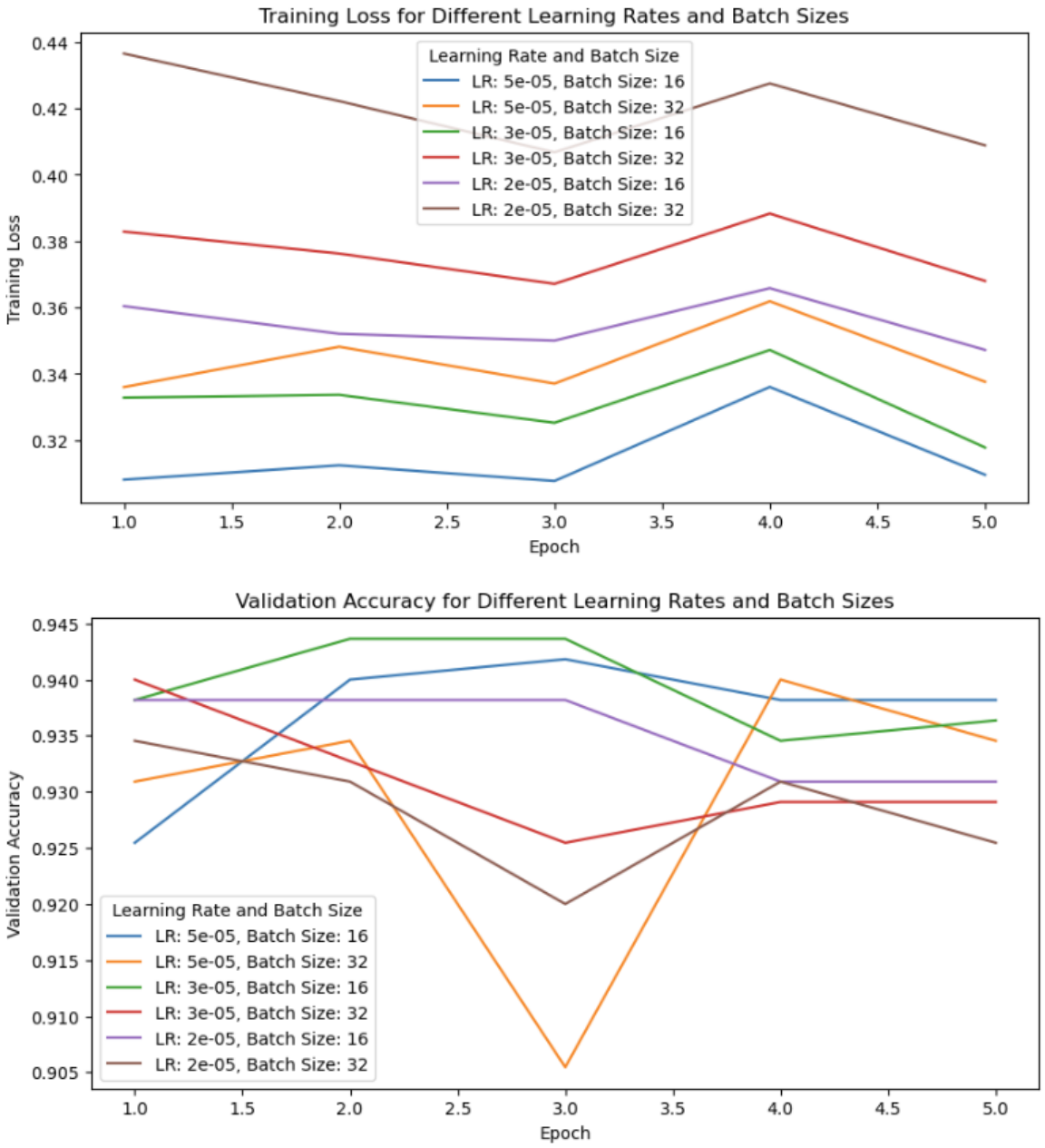

Fig. 5. Comparative Analysis of Training Loss and Validation Accuracy Across Learning Rates and Batch Sizes for DistilBERT

10.6 Use the trained model for inference on the unlabelled data

The trained models and tokenizers are loaded from the specified directories and the unlabelled dataset is prepared. Each text in the unlabelled dataset had the tokenizer applied to convert it into a format suitable for the model. The model is then utilised to obtain predictions with a max length of 128. This is the default number for max length and the saved models indicated the model could be adapted up to 512, but once the number increased from the default the model became too computationally intensive. The predicted

label, which has been determined by the class with the highest probability, is appended to the predictions list. A column with the predicted label is added to the 'sentiment' dataframe, containing the model's predictions for each text in the unlabelled dataset.

The BERT and DistilBERT model took 1h 38m 54s and 51m 58s to run respectively indicating a 46m 56s difference between the two models run times.

10.7 Comparing Models

In the evaluation of sentiment analysis performance on the customer query dataset, a thorough comparison was conducted among four distinct tools: spaCy, NLTK, BERT and DistilBERT. Notably, NLTK and spaCy were initially applied to the dataset revealing a high prevalence of positive sentiments. However, when both BERT models were trained on a labelled dataset and subsequently tested on the same query dataset, the overall sentiment shifted towards more negative and neutral results.

The confusion matrix results provide insights into the performance of sentiment analysis models (NLTK, spaCy, BERT, and DistilBERT) on the given dataset. Each confusion matrix represents the counts of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions for the corresponding pair of models.

When compared to spaCy, NLTK showed a high number of true positives and true negatives, with only 0.014 false positives and 0.0385 false negatives. BERT and DistilBERT exhibited lower agreement with the other models. BERT, when compared to NLTK, had a high number of false positives (0.487) and false negatives (0.033). DistilBERT, compared to NLTK, also had a substantial number of false positives (0.396) and false negatives (0.039).

The comparison between BERT and DistilBERT shows that these transformer-based models achieve higher accuracy when compared to one another rather than to NLTK and spaCy with 0.131 false positives and 0.034 false negatives and a high number of true positives and negatives (0.223 and 0.612 respectively).

Cohen's Kappa is a statistical measure of inter-rater agreement for categorical items which was developed by Jacob Cohen in 1960 (McHugh, 2012). Like most correlation statistics, it is a score that can range from -1 to 1. 'Cohen's Kappa does, however, have a drawback where it tends to return lower values when our classes are more imbalanced' (George, 2021). Accuracy is 'is an evaluation metric that allows you to measure the total number of predictions a model gets right' (D, 2019).

The comparison between NLTK and spaCy revealed a very high Cohen's Kappa (0.8819) and Accuracy (0.9501), indicating robust agreement between the two models. However, when comparing NLTK to BERT, the Cohen's Kappa (0.1370) and Accuracy (0.4799) values were lower, suggesting a weaker level of agreement. The NLTK vs. DistilBERT comparison showed moderate Cohen's Kappa (0.2256) and Accuracy (0.5651), indicating a moderate level of agreement between NLTK and DistilBERT models. Moving on to spaCy, the comparison with BERT yielded low Cohen's Kappa (0.1493) and Accuracy (0.4986) values, indicating weaker agreement. However, spaCy and DistilBERT demonstrated moderate agreement with Cohen's Kappa (0.2447) and Accuracy (0.5818). Finally, the BERT vs. DistilBERT comparison showed high Cohen's Kappa (0.6168) and Accuracy (0.8356), signifying strong agreement between BERT and DistilBERT models.

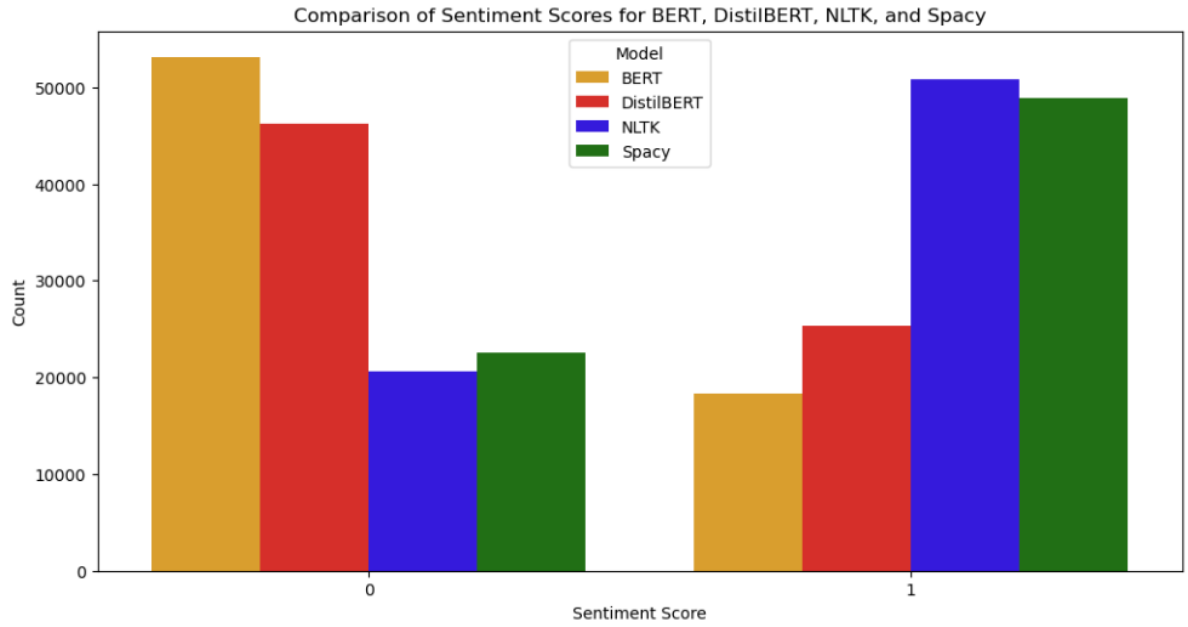

Fig. 6 Total Sentiment Counts Comparison Across NLTK, spaCy, BERT, and DistilBERT Models

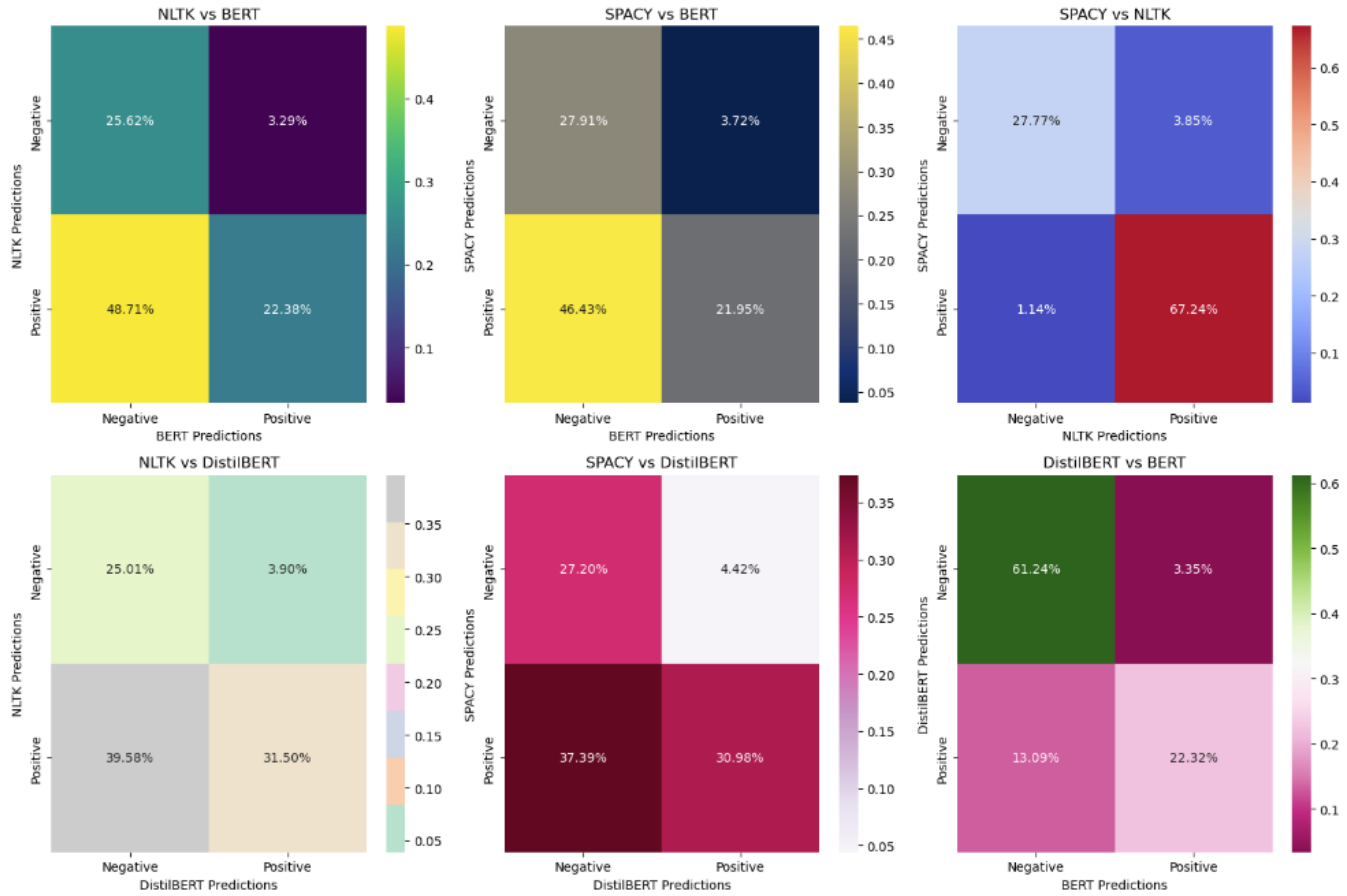

Fig. 7 Percentage Correlation Confusion Matrix Across NLTK, spaCy, BERT, and DistilBERT Models

	Model 1	Model 2	Cohen's Kappa	Accuracy
0	Nltk	Spacy	0.881899	0.950102
1	Nltk	Bert	0.137020	0.479960
2	Nltk	Distilbert	0.225584	0.565127
3	Spacy	Bert	0.149266	0.498560
4	Spacy	Distilbert	0.244679	0.581825
5	Bert	Distilbert	0.616752	0.835594

Table 4 Comparison of Sentiment Analysis Models: Cohen's Kappa and Accuracy Scores

	Time Comparison	
	BERT	DistilBERT
Training Loop	06:11:32	03:11:04
Fine Tune Model (Total)	00:12:26	00:12:39
Fine Tune Model (per epoch)	00:12:26	00:06:20
Apply Model	01:38:54	00:51:58

Table 5 Run Time Comparison of Sentiment Analysis Models: BERT vs. DistilBERT

11 Conclusions and Future Research

The comparative analysis of NLTK, spaCy, BERT, and DistilBERT on customer queries and feedback revealed unique trends. The noted similarities between BERT and DistilBERT, as well as NLTK and spaCy, offered insights into the similarities between these models.

To improve the precision and reliability of future models, a sub-set of the data should be hand labelled. This data will allow us to train the models on accurately labelled data allowing for more specific performance metrics.

To further enhance sentiment capabilities, future work could involve the inclusion of GPT. Because this model is known for its ability to understand and generate human like text, it could offer novel insights into the challenges and opportunities that are presented with sentiment analysis as well as increase the accuracy of sentiment predictions.

12 Evaluation of ethical and legal issues of the project

The company-specific data utilised in this project holds a high degree of sensitivity and is strictly proprietary, rendering it unsuitable for public use. Ensuring compliance with ethical standards and legal regulations is imperative to safeguarding the privacy and integrity of the company's information.

13 Bibliography

Agarwal, S. (2019). Deep Learning-based Sentiment Analysis: Establishing Customer Dimension as the Lifeblood of Business Management. *Global Business Review*, p.097215091984516. doi:https://doi.org/10.1177/0972150919845160.

Ahn, J., Son, H. and Chung, A.D. (2021). Understanding public engagement on twitter using topic modeling: The 2019 Ridgecrest earthquake case. *International Journal of Information Management Data Insights*, 1(2), p.100033. doi:https://doi.org/10.1016/j.jjimei.2021.100033.

Basiri, M.E., Nemati, S., Abdar, M., Asadi, S. and Acharrya, U.R. (2021). A novel fusion-based deep learning model for sentiment analysis of COVID-19 tweets. *Knowledge-Based Systems*, 228, p.107242. doi:https://doi.org/10.1016/j.knosys.2021.107242.

Bhandari, A. (2020). *Confusion matrix for machine learning*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2020/04/confusion-matrix-machine-learning/.

Blei, D.M., Ng, A.Y. and Jordan, M.I. (2003). Latent Dirichlet Allocation. *Journal of Machine Learning Research*, [online] 3, pp.993–1022. Available at: https://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf [Accessed 11 Feb. 2024].

Boulianne, S., Minaker, J. and Haney, T.J. (2018). Does compassion go viral? Social media, caring, and the Fort McMurray wildfire. *Information, Communication & Society*, 21(5), pp.697–711. doi:https://doi.org/10.1080/1369118x.2018.1428651.

Bowman, S., Angeli, G., Potts, C. and Manning, C.D. (2015). *A large annotated corpus for learning natural language inference*. [online] aclanthology.org. doi:https://doi.org/10.18653/v1/D15-1075.

Capuano, N., Greco, L., Ritrovato, P. and Vento, M. (2020). Sentiment analysis for customer relationship management: an incremental learning approach. *Applied Intelligence*, 51. doi:https://doi.org/10.1007/s10489-020-01984-x.

Carvalho, T. (2021). *Visualizing the Nothing*. [online] Medium. Available at: https://towardsdatascience.com/visualizing-the-nothing-ae6daccc9197 [Accessed 13 Feb. 2024].

Chen, T., Xu, R., He, Y. and Wang, X. (2017). Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Systems with Applications*, 72, pp.221–230. doi:https://doi.org/10.1016/j.eswa.2016.10.065.

D, E. (2019). *Accuracy, Recall & Precision*. [online] Medium. Available at: https://medium.com/@erika.dauria/accuracy-recall-precision-80a5b6cbd28d.

Danilák, M. (2024). *Mimino666/langdetect*. [online] GitHub. Available at: https://github.com/Mimino666/langdetect?tab=readme-ov-file [Accessed 20 Feb. 2024].

Devlin, J., Chang, M.-W., Lee, K. and Toutanova, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. [Paper] Available at: https://arxiv.org/abs/1810.04805 [Accessed 18 Feb. 2024].

Diekson, Z.A., Prakoso, M.R.B., Putra, M.S.Q., Syaputra, M.S.A.F., Achmad, S. and Sutoyo, R. (2023). Sentiment analysis for customer review: Case study of Traveloka. *Procedia Computer Science*, [online] 216, pp.682–690. doi:https://doi.org/10.1016/j.procs.2022.12.184.

Dimitrios Kouzis-Loukas (2018). *Learning Scrapy -.* Packt Publishing.

End User 1 (2024). *Primary Research*. 12 Feb.

End User 2 (2024). *Primary Research*. 14 Feb.

George, N. (2021). *Practical data science with Python : learn tools and techniques from hands-on examples to extract insights from data*. Birmingham: Packt Publishing.

Girdher, H. (2021). *TDM (Term Document Matrix) and DTM (Document Term Matrix)*. [online] Analytics Vidhya. Available at: https://medium.com/analytics-vidhya/tdm-term-document-matrix-and-dtm-document-term-matrix-8b07c58957e2.

González-Fernández, M. and González-Velasco, C. (2020). An alternative approach to predicting bank credit risk in Europe with Google data. *Finance Research Letters*, 35, p.101281. doi:https://doi.org/10.1016/j.frl.2019.08.029.

Halawani, H.T., Mashraqi, A.M., Badr, S.K. and Alkhalaf, S. (2023). Automated sentiment analysis in social media using Harris Hawks optimisation and deep learning techniques. *Alexandria Engineering Journal*, [online] 80, pp.433–443. doi:https://doi.org/10.1016/j.aej.2023.08.062.

Hinduja, S., Afrin, M., Mistry, S. and Krishna, A. (2022). Machine learning-based proactive social-sensor service for mental health monitoring using twitter data. *International Journal of Information Management Data Insights*, [online] 2(2), p.100113. doi:https://doi.org/10.1016/j.jjimei.2022.100113.

Honnibal, M. and Johnson, M. (2015). *An Improved Non-monotonic Transition System for Dependency Parsing*. [online] ACLWeb. doi:https://doi.org/10.18653/v1/D15-1162.

Hugging Face (2022). *bert-base-uncased · Error: 'Some weights of the model checkpoint were not used'*. [online] huggingface.co. Available at: https://huggingface.co/bert-base-uncased/discussions/4 [Accessed 7 Feb. 2024].

Hugging Face (2024). *google-bert/bert-base-uncased · Hugging Face*. [online] huggingface.co. Available at: https://huggingface.co/google-bert/bert-base-uncased [Accessed 18 Feb. 2024].

Huggingface (2023). *BERT fine tuning low epochs?* [online] Hugging Face Forums. Available at: https://discuss.huggingface.co/t/bert-fine-tuning-low-epochs/54869 [Accessed 13 Feb. 2024].

Huyen, C. (2022). *DESIGNING MACHINE LEARNING SYSTEMS : an iterative process for production-ready applications.* S.L.: O'reilly Media, Inc, Usa.

Individual with expertise in deep learning (2024). *Primary Research*. 12 Feb.

Jirasatjanukul, K., Nilsook, P., & Wannapiroon, P. (2019). Intelligent human resourcemanagement using latent semantic analysis with the internet of things. InternationalJournal of Computer Theory and Engineering, 11(2).

Kontonatsios, G., Clive, J., Harrison, G.R., Metcalfe, T., Sliwiak, P., Tahir, H. and Ghose, A. (2023). FABSA: An aspect-based sentiment analysis dataset of user reviews. *Neurocomputing*, 562, pp.126867–126867. doi:https://doi.org/10.1016/j.neucom.2023.126867.

Kooten, P. van (2023). *contractions*. [online] GitHub. Available at: https://github.com/kootenpv/contractions.

Kotzias, D. and UCI Machine Learning Repository (2015). *Sentiment Labelled Sentences*. [online] archive.ics.uci.edu. Available at: https://archive.ics.uci.edu/dataset/331/sentiment+labelled+sentences.

Kotzias,Dimitrios. (2015). Sentiment Labelled Sentences. UCI Machine Learning Repository. https://doi.org/10.24432/C57604.

Lal, U. (2022). *Increasing Accuracy of Sentiment Classification Using Negation Handling*. [online] Medium. Available at: https://towardsdatascience.com/increasing-accuracy-of-sentiment-classification-using-negation-handling-9ed6dca91f53 [Accessed 14 Feb. 2024].

Landauer, T. K. (2002). Applications of latent semantic analysis. Proceedings of theannual meeting of the cognitive science society, 24

Leippold, M. (2023). Sentiment Spin: Attacking Financial Sentiment with GPT-3. *SSRN Electronic Journal*. doi:https://doi.org/10.2139/ssrn.4337182.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. [online] doi:arxiv:1907.11692.

Lukei (2019). *Dealing with contractions in NLP*. [online] Medium. Available at: https://medium.com/@lukei_3514/dealing-with-contractions-in-nlp-d6174300876b.

Mahdikhani, M. (2022). Predicting the popularity of tweets by analyzing public opinionand emotions in different stages of COVID-19 pandemic. International Journal of Information Management Data Insights, 2(1), 100053.

McHugh, M.L. (2012). Interrater reliability: the kappa statistic. *Biochemia Medica*, [online] 22(3), pp.276–282. doi:https://doi.org/10.11613/bm.2012.031.

McKinney, W. (2010). Data structures for statistical computing in python. *Proceedings of the 9th Python in Science Conference*, [online] 445, pp.56–61. doi:https://doi.org/10.25080/Majora-92bf1922-00a.

Mehta, S. (2022). *When to use negation handling in sentiment analysis?* [online] Analytics India Magazine. Available at: https://analyticsindiamag.com/when-to-use-negation-handling-in-sentiment-analysis/#:~:text=Negation%20handling%20is%20a%20method.

Meškelė, D. and Frasincar, F. (2020). ALDONAr: A hybrid solution for sentence-level aspect-based sentiment analysis using a lexicalized domain ontology and a regularized neural attention model. *Information Processing & Management*, 57(3), p.102211. doi:https://doi.org/10.1016/j.ipm.2020.102211.

Mukherjee, P., Badr, Y., Doppalapudi, S., Srinivasan, S.M., Sangwan, R.S. and Sharma, R. (2021). Effect of Negation in Sentences on Sentiment Analysis and Polarity Detection. *Procedia Computer Science*, 185, pp.370–379. doi:https://doi.org/10.1016/j.procs.2021.05.038.

Neogi, A. S., Garg, K. A., Mishra, R. K., & Dwivedi, Y. K. (2021). Sentiment analysis andclassification of indian farmers' protest using twitter data. International Journal of Information Management Data Insights, 1(2), 100019.

Nguyen, T.T., Meng, H.-W., Sandeep, S., McCullough, M., Yu, W., Lau, Y., Huang, D. and Nguyen, Q.C. (2018). Twitter-derived measures of sentiment towards minorities (2015–2016) and associations with low birth weight and preterm birth in the United States. *Computers in Human Behavior*, [online] 89, pp.308–315. doi:https://doi.org/10.1016/j.chb.2018.08.010.

NLTK (2023). *nltk.tokenize package — NLTK 3.5 documentation*. [online] www.nltk.org. Available at: https://www.nltk.org/api/nltk.tokenize.html.

Obembe, D., Kolade, O., Obembe, F., Owoseni, A., & Mafimisebi, O. (2021). COVID-19 andthe tourism industry: An early stage sentiment analysis of the impact of social mediaand stakeholder communication. International Journal of Information Management DataInsights, 1(2), 100040.

Ozdemir, S. (2023). *Quick Start Guide to Large Language Models*. Addison-Wesley Professional.

Python (2008). *UnicodeDecodeError - Python Wiki*. [online] wiki.python.org. Available at: https://wiki.python.org/moin/UnicodeDecodeError [Accessed 20 Feb. 2024].

PyTorch (n.d.). *Writing Custom Datasets, DataLoaders and Transforms — PyTorch Tutorials 1.10.1+cu102 documentation*. [online] pytorch.org. Available at: https://pytorch.org/tutorials/beginner/data_loading_tutorial.html.

Radford, A., Wu, J., Child, R., Amodei, D., Sutskever , I. and Luan, D. (2019). *Language Models are Unsupervised Multitask Learners*.

Ridhwan, K. M., & Hargreaves, C. A. (2021). Leveraging twitter data to understand publicsentiment for the COVID-19 outbreak in singapore. International Journal of InformationManagement Data Insights, 1(2), 100021.

Rodríguez-Ibanez, M., Casanez-Ventura, A., Cuenca-Jim´enez, P.-M. and Castejon-Mateos, F. (2023). A review on sentiment analysis from social media platforms. *Expert Systems with Applications*, 223, p.119862. doi:https://doi.org/10.1016/j.eswa.2023.119862.

saffsd (2021). *saffsd/langid.py*. [online] GitHub. Available at: https://github.com/saffsd/langid.py.

Samaras, L., García-Barriocanal, E. and Sicilia, M.-A. (2023). Sentiment analysis of COVID-19 cases in Greece using Twitter data. *Expert Systems with Applications*, [online] 230, p.120577. doi:https://doi.org/10.1016/j.eswa.2023.120577.

Sanh, V., Debut, L., Chaumond, J. and Wolf, T. (2019). *DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter*. [Paper] Available at: https://arxiv.org/abs/1910.01108 [Accessed 19 Feb. 2024].

Saturn Cloud (2023). *A List of Pandas readcsv Encoding Options | Saturn Cloud Blog*. [online] saturncloud.io. Available at: https://saturncloud.io/blog/a-list-of-pandas-readcsv-encoding-options/ [Accessed 13 Feb. 2024].

Saumyab271 (2022). *Stemming vs Lemmatization in NLP: Must-Know Differences*. [online] Analytics Vidhya. Available at: https://www.analyticsvidhya.com/blog/2022/06/stemming-vs-lemmatization-in-nlp-must-know-differences/#:~:text=While%20stemming%20involves%20chopping%20off [Accessed 23 Feb. 2024].

Sazzed, S. (2020). Cross-lingual sentiment classification in low-resource bengali language. In Proceedings of the sixth workshop on noisy user-generated text (W-NUT 2020)(pp. 50–60).

Sazzed, S. (2021). A hybrid approach of opinion mining and comparative linguistic analysis of restaurant reviews. In Proceedings of the international conference on recent advancesin natural language processing (RANLP 2021): 4 (pp. 1281–1288). Elsevier.

Sazzed, S. (2022a). Identifying neutral reviews from unlabeled data: An exploratory studyon user ratings and word-level polarity scores. In Proceedings of the 33rd ACM conference on hypertext and social media (pp. 198–202).

Sazzed, S. (2022). Impact of demography on linguistic aspects and readability of reviews and performances of sentiment classifiers. *International Journal of Information Management Data Insights*, 2(2), p.100135. doi:https://doi.org/10.1016/j.jjimei.2022.100135.

Sazzed, S., & Jayarathna, S. (2019). A sentiment classification in bengali and machinetranslated English corpus. In 2019 IEEE 20th international conference on informationreuse and integration for data science (IRI) (pp. 107–114). IEEE.

Sazzed, S., & Jayarathna, S. (2021). SSentia: A self-supervised sentiment analyzer for classification from unlabeled data. Machine Learning with Applications, 100026.

Schwartz, R., Dodge, J., A. Smith, N. and Etzioni, O. (2019). Green AI. [online] doi:arxiv:1907.10597.

ScienceDirect (2014). *Term Matrix - an overview | ScienceDirect Topics*. [online] www.sciencedirect.com. Available at: https://www.sciencedirect.com/topics/mathematics/term-matrix [Accessed 18 Feb. 2024].

Serrano-Guerrero, J., Bani-Doumi, M., Romero, F.P. and Olivas, Á. (2024). A 2-tuple fuzzy linguistic model for recommending health care services grounded on aspect-based sentiment analysis. *Expert Systems with Applications*, 238, pp.122340–122340. doi:https://doi.org/10.1016/j.eswa.2023.122340.

Siino, M., Tinnirello, I. and Marco La Cascia (2024). Is text preprocessing still worth the time? A comparative survey on the influence of popular preprocessing methods on Transformers and traditional classifiers. *Information Systems*, 121, pp.102342–102342. doi:https://doi.org/10.1016/j.is.2023.102342.

SkimAI (2020). *Tutorial: Fine-tuning BERT for Sentiment Analysis*. [online] Skim AI. Available at: https://skimai.com/fine-tuning-bert-for-sentiment-analysis/.

Spacy (n.d.). *Linguistic Features · spaCy Usage Documentation*. [online] Linguistic Features. Available at: https://spacy.io/usage/linguistic-features#how-tokenizer-works.

Strubell, E., Ganesh, A. and McCallum, A. (2019). Energy and policy considerations for deep learning in nlp. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, [online] pp.3645–3650. doi:https://doi.org/10.18653/v1/P19-1355.

Sugimura, M. (2019). *BERT Classifier: Just Another Pytorch Model*. [online] Medium. Available at: https://towardsdatascience.com/bert-classifier-just-another-pytorch-model-881b3cf05784 [Accessed 18 Feb. 2024].

Suleman, R.M. and Korkontzelos, I. (2020). Extending latent semantic analysis to manage its syntactic blindness. *Expert Systems with Applications*, p.114130. doi:https://doi.org/10.1016/j.eswa.2020.114130.

Takahashi, B., Tandoc, E.C. and Carmichael, C. (2015). Communicating on Twitter during a disaster: An analysis of tweets during Typhoon Haiyan in the Philippines. *Computers in Human Behavior*, 50, pp.392–398. doi:https://doi.org/10.1016/j.chb.2015.04.020.

The pandas development team (2020). pandas-dev/pandas: Pandas. *zenodo.org*, [online] 2.0.3. doi:https://doi.org/10.5281/zenodo.10537285.

Vetterle, J. (2020). *How To Make The Most Out Of Bert Finetuning*. [online] Medium. Available at: https://towardsdatascience.com/how-to-make-the-most-out-of-bert-finetuning-d7c9f2ca806c [Accessed 13 Feb. 2024].

Vrana, S. R., Vrana, D. T., Penner, L. A., Eggly, S., Slatcher, R. B., & Hagiwara, N. (2018).Latent semantic analysis: A new measure of patient-physician communication. SocialScience & Medicine, 198, 22–26.

Wegba, K., Lu, A., Li, Y. and Wang, W. (2018, March). Interactive Storytelling for MovieRecommendation through Latent Semantic Analysis. In 23rd Internationalconference on intelligent user interfaces (pp. 521–533)

Weichselbraun, A., Gindl, S. and Scharl, A. (2014). Enriching semantic knowledge bases for opinion mining in big data applications. *Knowledge-Based Systems*, 69, pp.78–85. doi:https://doi.org/10.1016/j.knosys.2014.04.039.

Young, P., Lai, A., Hodosh, M. and Hockenmaier, J. (2014). From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2, pp.67–78. doi:https://doi.org/10.1162/tacl_a_00166.

14 Appendix

14.1 Interviews

14.1.1 Candidate 1 - Individual with expertise in deep learning

Can you provide a brief overview of your experience in deep learning?

The first project using Neural Networks was using an LTSM to try estimate time dependent variables based on other time dependent variables. I have also worked on the generation of a chatbot for the company I work for. The chatbot was trained on the company literature and will be allowed to connect to the database of clients. The bot can write SQL queries to get information from the clients database. We are feeding the chatbot validated papers only, therefore information output is based on validated papers which is a way to control the output – we always provide references for the bot to provide the information. The bot will always be a beta version due to the level of uncertainty regarding the output.

How did you get started with deep learning, and what motivated your interest in this field?

The project itself. Speed cannot be calculated indoors because of satellites and therefore the metrics cannot be calculated. Inertial measurement units in our device calculate speed and distance, LSTM can store previous information, so it was great for time dependent variables. The chatbot was because the market is opening to AI. One was guided by the problem, and one was guided by the market.

When working on deep learning tasks, how do you go about selecting the appropriate model and architecture?

I looked at previous instances of time without using assumptions, I did not want to be constrained or limited by a model that will always make assumptions on your predictions.

What strategies do you use for training deep learning models, and how do you handle issues like overfitting or underfitting?

LSTMs were trained on azure cloud with 100 epochs. The results plateaued after 10 epochs. I determined the percentage of data that was needed and how many cycles needed to be run to not overfit. The chatbot has no way to confirm accuracy as it is not classifying, the results generated are from client information. The user experience is used to determine the accuracy.

Have you encountered challenges in fine-tuning models for specific tasks?

With the LSTM there was a large amount of data to train on. I manipulated the data with epochs, saw the best results and then reduced the epochs.

In your work with deep learning, what ethical considerations do you consider?

LSTM needs to use anonymised data, so the inputs are person agnostic. This means it cannot be trained on specific data. With the chatbot – the company does not own the client data. Clients store their own data in the cloud and the company works on ids (which the client can relate back to the person). The current issue we are facing is how to connect encrypted databases to the encryption on the clients end.

How do you address potential biases or fairness issues in the models you develop?

With the LSTM – it has been a struggle to get a database that is big and diverse enough to train on. With the chatbot we are using generative AI which has purposeful bias. It is trained on the company's own literature and therefore the answers are constrained to the literature used.

What approaches do you use for hyperparameter tuning when working on models?

LSTM can control stored sample data. Because it always stores the previous second it will be enough to train on speed.

How do you approach the challenge of making deep learning models for more interpretable?

With the LSTM the result outputs are speed and no not need to be manipulated.

What lessons have you learned from your experiences in applying deep learning to tasks?

How to use an LSTM being exposed to time dependent data. Normally with text or image recognition, the biggest advantage is signal prediction (usually in the stock market for LSTMs). I have learned a lot from the exposure to smaller models that are not traditionally taught in tertiary education, especially when dealing with signals and continuous data with time dependent variables.

(Individual with expertise in deep learning, 2024)

14.1.2 Candidate 2 - End User

Can you share your general experience with customer query resolution services in terms of communication and assistance?

Our team needs to understand customer pull periods, finance and tech. They use a lot of macros (which are template responses to frequently asked questions) for query resolution with customers. Our queries from the cloud based customer service platform are mainly written queries.

How satisfied are you with the current methods of resolving queries or issues?

The platform needs to be used more as a tool with more reliance on the macros to assist with resolving queries, we currently have six more macros that need to be implemented.

Can you recall specific instances where you felt particularly positive or negative about the resolution process?

We currently have an excellent escalation route to our tech expert using a decision tree and the team needs to have all their information before they approach the tech expert. If we could surmise the information without human intervention, the relevant queries could go straight to the tech expert.

Do you have a preferred communication channel for addressing queries or issues (e.g., live chat, email, phone)?

Ideally written queries. Very few come through on the phone. Livechat is no longer used, rather we have a bot with messaging which takes away time pressures to be present. The bot has populated mandatory fields which get push onto a ticket for one of our agents to resolve.

How does your choice of communication channel impact your overall sentiment during the query resolution process?

Our communication channels in order of most used are our contact form, email, messaging, social media (Instagram and Facebook. Depends on the period, during the sales periods (which are the busier periods) the preference is towards instant one touch communication. Communication and the sentiment attached to it is dependent on if we are in a sales period or not.

How effective do you find the responses provided during the query resolution process?

They could still be improved. We are currently trying to standardise responses, and document processes based on feedback from escalation manager.

Are there specific factors that contribute to your perception of response effectiveness?

Sentiment. Ticket numbers are given more focus than sentiment. When the sentiment is negative it would be nice to see if an assumed sentiment and our resources match the analysed sentiment.

We do find more value in interpreting the sentiment of our customers from websites like trustpilot.

What are your expectations when reaching out for query resolution assistance?

The platform must work harder for us, automations and triggers need to be used more effectively. The agent needs to do less one touch response interactions so that there is more availability for the complex tickets coming through which, if not handled correctly, will impact the overall sentiment.

How important is personalization and context awareness in the responses you receive during query resolution?

Increasingly important. There is an assumption when queries, especially for returns, come through social media that the company knows the customer details without them being actively supplied. Certain account managers want access to older tickets to understand the history of issues with the club. The platform can look at tags, keywords or phrases that could help for customer sentiment.

14.1.3 Candidate 3 - End User

Can you share your general experience with customer query resolution services in terms of communication and assistance?

There is still more to explore on how we can address queries coming through. So far the solution is effective for the issues that are raised especially the technical ones. The process that we currently have works well, unless there are extreme cases which are time critical in which case a live call would be more appropriate.

How satisfied are you with the current methods of resolving queries or issues?

It can obviously be improved, and we have been constantly improving. The sentiment that we have received from customers is that sometimes it takes too long to resolve. An improvement that I would suggest would be for the first level to have a better technical base, so that they could analyse the ticket better so that by the time it reaches me the customers mood is still the same.

How would you describe the overall sentiment of interactions when seeking assistance with a query or issue?

Neutral going towards positive most of the time. During the sales period the ticket numbers increase, the sentiment veers towards negative especially if the tickets are not responded to in time.

Can you recall specific instances where you felt particularly positive or negative about the resolution process?

There is more positivity when there are returning customers. They seem to return because the experience was positive, and their queries were resolved in a timely manner and trouble shooting was done well.

Do you have a preferred communication channel for addressing queries or issues (e.g., live chat, email, phone)?

Technical issues – I prefer emails, because its easier for us to get details from them. When a customer is really struggling with the technology, I find that giving the customer a phone call to guide them through works well. Pre-sale periods queries are generally delivery related; they are small queries which are easy to resolve through live chat.

How does your choice of communication channel impact your overall sentiment during the query resolution process?

Yes. Especially when there are steps that are too complex for the customer. When a difficult query is converted to a phone call the sentiment improves and this is visible through the follow up communication with the customer.

How effective do you find the responses provided during the query resolution process?

There is room for improvement. For existing issues we have tools in place to sort easily, new issues can be trickier, and the focus needs to be on gleaning what the customer has given us for the details.

Are there specific factors that contribute to your perception of response effectiveness?

There is no specific tool, we can determine it from customer replies. There is an option to rate a ticket after it has been resolved and this feedback helps us with improvement. Trustpilot also gives us an indication into how customers feel about the responses.

How important is personalization and context awareness in the responses you receive during query resolution?

Very important. Its important to make the customers feel like they have been heard. There are customers who don't like automated messages and need more personalisation.

Do you adapt your resourcing according to busier periods when the sentiment tends to be lower?

Yes, there are fewer holidays and more agents during the periods that have historically had the highest ticket counts. We are also trying to manoeuvre customers to use the self help options that we have so that agents are available for more complicated queries.

To what extent does the query resolution experience and customer sentiment impact your perception of the brand or company?

The Trustpilot and app reviews impact the sentiment towards the brand. Customers tend to vocalise their opinion when they are experiencing negative sentiment, so I try to ensure that the ticket is not the source of that negativity, and that the customer is aware that we are actively working on a resolution.

How do you think the general sentiment of customer queries can benefit query resolution?

I believe that pinpointing the keywords, specifically by department, can highlight issues. The finer detail will enable us to pinpoint the issues more accurately and deal with them more effectively. The NLP work can give us insights into any times outside of sale periods that there is negative sentiment.

How do you deal with queries that are not in English?

The platform auto translates the query for us.