

UNIVERSITÀ DEGLI STUDI “G. d’Annunzio”
Chieti-Pescara

SCUOLA SUPERIORE “G. d’Annunzio”

DOCTORAL PROGRAM IN

Business, Institutions, Markets

CYCLE

XXXVII

**DEEP LEARNING IN ASPECT-BASED SENTIMENT
ANALYSIS: FROM TEXT TO INSIGHTS UNDERSTANDING**

Department of Legal and Social Sciences
Academic Discipline INFO-01/A

PhD Student:
Loris Di Quilio

Loris Di Quilio

Doctoral Program Chair:
Prof. Marco Angelone

Marco Angelone

Tutors:

Prof. Fabio Fioravanti

Fabio Fioravanti

Prof. Luca Moscardelli

Luca Moscardelli

Prof. Maurizio Parton

Maurizio Parton

Academic years 2021/2024



UNIONE EUROPEA
Fondo Sociale Europeo



Ministero dell'Università
e della Ricerca



PON
RICERCA
E INNOVAZIONE
2014-2020

REACT EU

La borsa di dottorato è stata cofinanziata con risorse del
Programma Operativo Nazionale Ricerca e Innovazione 2014-2020, risorse FSE REACT-EU
Azione IV.4 “Dottorati e contratti di ricerca su tematiche
dell’innovazione” e Azione IV.5 “Dottorati su tematiche Green”

Dedicated to my family.

Abstract

As the volume of unstructured text data generated globally and on the Internet continues to grow, the need to intelligently process this data and extract various types of knowledge from it becomes increasingly critical. This thesis delves into the task of Sentiment Analysis (SA) by leveraging advanced Natural Language Processing (NLP) techniques, with a focus on the specialized field of Aspect-Based Sentiment Analysis (ABSA) and its sub-tasks.

The work begins with an overview of NLP, its applications, and state-of-the-art methods in Sentiment Analysis, setting the foundation for exploring ABSA. The thesis distinguishes between single and multiple extractions in ABSA, where the latter involves the identification of complex relationships such as Aspect-Category Opinion Sentiment (ACOS) quadruples. A dedicated benchmark study on a custom dataset for ACOS is presented.

Additionally, the thesis examines Aspect-Category Sentiment Analysis (ACSA), developing a framework to convert data between tasks and different tools available in the literature for various ABSA tasks, to annotate datasets manually and semi-automatically through the constructed model, and to generate sentiment reports on the topics starting from a list of reviews, covering the entire application flow. It also introduces a dataset in the Beauty and Personal Care domain, on which the implemented model is trained.

To expand the applicability of ABSA, a case study in the Beauty and Personal Care domain demonstrates the practical implementation of sentiment analysis techniques. By leveraging consumer reviews and exploring their implications for sales forecasting, this

thesis also develops an approach to quantify the relationship between product features and market performance. This fits into the initial phase of product design: the collection of user requirements, thus enabling a deeper understanding of consumer preferences. To achieve this, ACSA was used within a sales forecasting model, resulting in improved sales predictions.

The proposed methodologies aim to provide new insights for researchers and professionals, highlighting the relevance of ABSA in addressing real-world challenges in the e-commerce sector and beyond.

Keywords: Sentiment Analysis, Aspect-based Sentiment Analysis, Aspect-Category Sentiment Analysis, Aspect-Category-Opinion-Sentiment, Deep Learning, Transformers, Sales Forecasting, Predictors impact

Contents

Abstract	i
Contents	iii
1 Introduction	1
1.1 Thesis structure	1
1.2 Natural Language Processing (NLP)	3
1.3 Deep Learning (DL) for NLP	4
1.4 Transformers	7
1.4.1 Architecture	7
1.4.2 Attention mechanism	10
1.5 Text-to-Text Transfer Transformer (T5)	12
1.5.1 Flan-T5	13
1.5.2 Application tasks	15
2 Sentiment Analysis	21
2.1 Introduction to the Sentiment Analysis	21
2.2 Levels of Sentiment Analysis	23
2.2.1 Document level analysis	23
2.2.2 Sentence level analysis	24
2.2.3 Aspect level analysis	24
2.3 Evolution of Sentiment Analysis Methods	25
2.3.1 Traditional Lexicon-based Approaches	25
2.3.2 Machine Learning Approaches	27
2.3.3 Hybrid Approaches	28
2.3.4 Neural Networks Approaches	28
2.3.5 The advent of Transformers	29
3 Aspect-Based Sentiment Analysis	31

3.1	ABSA elements	32
3.2	Single ABSA tasks	33
3.2.1	Aspect Term Extraction (ATE)	34
3.2.2	Aspect Category Detection (ACD)	35
3.2.3	Opinion Term Extraction (OTE)	36
3.2.4	Aspect Sentiment Classification (ASC)	39
3.3	Multiple ABSA tasks	41
3.3.1	Aspect-Opinion Pairs Extraction	42
3.3.2	End-to-End ABSA (E2E-ABSA)	44
3.3.3	Aspect Category Sentiment Analysis (ACSA)	46
3.3.4	Aspect Sentiment Triplet Extraction (ASTE)	49
3.3.5	Aspect-Category-Sentiment Detection (ACSD)	50
3.3.6	Aspect-Category Opinion Sentiment (ACOS)	51
3.4	Datasets and Metrics	52
3.4.1	SemEval Datasets	53
3.4.2	ACOS Datasets	55
3.4.3	Other Datasets	56
3.4.4	Evaluation Metrics	57
4	Applied ACOS Benchmark	59
4.1	Dataset and annotation tool	59
4.2	Experimental Evaluation	62
4.2.1	Paraphrase modeling	62
4.2.2	Extract Classify-ACOS	63
4.2.3	PyABSA	65
4.3	Results	67
5	PyACSA: A Comprehensive Framework for ACSA	69
5.1	Beauty Dataset	69
5.2	PyACSA tool	73
5.3	ACSA Utilities	74
5.3.1	Data Converter	75
5.3.2	Data annotation	76
5.3.3	Report Generation	78
5.4	Experimental evaluation	80
5.5	Experimental settings and results	81
6	An Application of ACSA to Consumer Review Analysis to Inform Prod-	

uct Design	85
6.1 Data-driven product design	87
6.1.1 Product design based on consumer reviews	88
6.1.2 Effect of product review on sales	89
6.2 Methodology	90
6.2.1 Data Collection	91
6.2.2 Data preprocessing	93
6.3 Top review detection	96
6.3.1 Data Preparation and Feature Engineering	97
6.3.2 Classification model	98
6.4 Aspect Category Sentiment Analysis model	99
6.5 Sales Forecasting with ACSA	102
6.5.1 Data Transformation	104
6.5.2 Sales forecasting model: Multilayer Perceptron (MLP)	105
6.6 SHAP Analysis	107
6.7 Results and discussion	108
6.7.1 Top reviews detection results	108
6.7.2 Sales Forecast with ACSA results	109
6.7.3 SHAP Analysis results: Predictor impact and Conclusions	111
7 Conclusion and future work	123
Bibliography	127
8 Appendix	161
List of Figures	163
List of Tables	165
Acknowledgements	167

1 | Introduction

1.1. Thesis structure

This thesis aims to address one of the main challenges related to Aspect-Based Sentiment Analysis (ABSA), which does not lie so much in the development of new technologies to tackle it, but in the difficulty of their practical application. Often, these techniques remain confined to very academic contexts, limited to a single methodology or a single step of the application chain, without ever translating into concrete solutions that can influence real decision-making processes. Starting from this issue, the objective is to develop methodologies and tools that allow addressing the practical challenges of ABSA and extending its applicability to solve real problems, especially in the online sales sector.

In the following Chapter, after a general overview of the fundamentals of Natural Language Processing and Deep Learning (DL), highlighting their importance in modern applications, the architecture of *Transformers* is explored, focusing on their attention mechanism. Finally, the *Text-To-Text (T5)* model and its variant *Flan-T5* are presented, analyzing their main application fields.

Chapter two provides a comprehensive overview of sentiment analysis, describing the various levels at which it can be applied: document, sentence, and aspect levels. The chapter explores the evolution of the methods used, from lexicon-based approaches to modern models based on neural networks and *Transformers*.

Chapter 3 focuses on the literature review of the research field, *Aspect-Based Sentiment*

Analysis (ABSA). The tasks of this field are explored, differentiating between single and multiple extractions, as well as the relationships between them. Single extraction focuses on identifying individual aspects within the text, such as aspect term extraction (*ATE*) or category detection (*ACD*), while multiple extractions involve more complex activities, such as extracting pairs, triplets, and quadruplets of elements (e.g., E2E-ABSA, ASTE, and ACOS).

Chapter 4 conducts an applied benchmark for the most complex task in this field: *Aspect-Category Opinion Sentiment (ACOS)*. The details of the dataset and the annotation tool used are presented, followed by an experimental evaluation among various models available in the literature.

Chapter 5 deals with *Aspect-Category Sentiment Analysis (ACSA)*, introducing the development of a model (PyACSA) and a framework with various utilities, including a framework to annotate data both manually and semi-automatically using the model and a section to generate a report from a list of reviews. The thesis also introduces a dataset based on the *Beauty and Personal Care* domain, where the model is trained with specific categories relevant to this field. The chapter concludes with a comparison, on different datasets, with a state-of-the-art tool that uses the same approach.

Chapter 6 explores how sentiment analysis and in particular the ACSA task can serve as a valuable tool to amplify the voice of consumers and as a tool to improve the decision-making process in Product Design. In particular, the model developed in the previous chapter is used in another model (*Sales Forecast*) to obtain the most impactful categories for consumers, by category. This is because not all product features are equally important to users; great performance of a certain feature may not necessarily lead to higher demand for the product, but may even lead to deterioration if combined with other features (categories). This thesis proposes a method to quantify the effect of consumer evaluations of product features on future sales. By classifying product features based on their impact on sales, this method helps prioritize product development. Additionally, an improvement

in sales forecasts is observed thanks to the introduction of sentiment categories into the model.

All aspects covered in this thesis are based on e-commerce reviews.

1.2. Natural Language Processing (NLP)

Imagining a world where machines fully understand, interact, and communicate with humans just as we do with each other is no longer so far-fetched. This increasingly present reality is the goal that Natural Language Processing (NLP) strives to achieve, a field of Artificial Intelligence (AI) that has shown the most evolution in recent years. Natural Language Processing (NLP) is a subfield of computer science and artificial intelligence (AI) that uses machine learning to enable computers to understand and communicate using human language [65]. However, people communicate in various ways: speaking and listening, making gestures, using specialized manual signals, using sign language, or through various forms of text. By text, we mean words written or printed on a flat surface (paper, billboards, and so on) or displayed on a screen or electronic device to be read by the intended recipient (or anyone who happens to pass by). Text is the primary source of language on which NLP is based, relevant since its inception with the *Turing* test, where the ability to generate and understand text indistinguishably from a human is fundamental to demonstrating artificial intelligence [116]. This thesis will focus on this latter source, text, which plays an important role within companies that have a significant amount of data in this format. For this reason, NLP, as an emerging sector, plays a crucial role within these companies. It aids in the continuous search for innovative and digital methods to adapt to evolving environments, thus improving operational efficiency, increasing productivity, and automating business operations, allowing them to gain a competitive advantage. Indeed, the adoption of NLP is set to grow in various sectors, thanks to the continuous development of new applications and techniques that meet the diverse needs of companies and their customers. In recent years,

NLP has received significant interest thanks to the advent of Large Language Models (LLMs). These models have revolutionized the way NLP operates, enabling exceptional performance across a wide range of linguistic tasks. Today, when referring to NLP, one immediately thinks of LLMs (such as GPT-4 released by OpenAI in 2023), which represent the state of the art in this field. However, it is important to recognize that NLP has a long history of approaches and methodologies that have contributed to its evolution, starting from the earliest applications that were based on simple if-then decision trees, using hand-written grammatical and syntactic rules. Over the years (1970s and 1980s), the advent of statistical models enabled the development of techniques for syntactic and semantic language analysis, such as syntactic parsers. These models mapped linguistic elements with vectors, allowing the use of mathematical methods for their analysis, such as regression or Markov models. In the 2000s, Deep Learning began to emerge in this field, starting with recurrent neural networks (RNN) [72] and Long Short-Term Memory (LSTM) [62], leading to significant advancements in NLP. These advancements greatly improved performance in tasks like machine translation. Subsequently, Word Embedding techniques, such as Word2Vec and GloVe, allowed words to be represented as vectors in a continuous space, capturing the semantic relationships between them. Finally, the arrival of Transformers and Fine-Tuning marked a further advancement in the field.

These latter developments will be explored in the following section.

1.3. Deep Learning (DL) for NLP

Deep Learning refers to a subset of machine learning that uses artificial neural networks to model and solve complex problems. Neural networks are computational systems inspired by the structure and function of the human brain [97]. They consist of layers of interconnected nodes (neurons) that process and transform data, learning patterns through training. In their *deep* variant, these networks include multiple hidden layers, enabling them to capture hierarchical and abstract representations of data. Within NLP,

deep learning technologies, such as convolutional neural networks (CNN) [97] and recurrent neural networks (RNN), have driven advancements in understanding and generating human language

Regarding artificial neural networks, these existed even before the advent of Artificial Intelligence, but the first deep architectures emerged in the 1970s. However, these were difficult to train due to the problems of exploding (when the correction signals become so strong that the network makes huge adjustments) and vanishing gradients [21, 101] (which happen when these signals fade so much that the network struggles to learn from connections between distant events in time). In the mid-2000s, solutions such as unsupervised pre-training were proposed, where a network was initially trained on a large unlabeled dataset and then fine-tuned for a specific problem [16].

These advancements led to growing interest from the Machine Learning community. In many fields, statistical solutions that were previously state-of-the-art have been replaced by deep neural networks, thanks to their end-to-end architecture that allows them to be designed in a task-agnostic manner, as they can handle large amounts of training data [4].

Deep learning, therefore, can be defined as the bridge between Artificial Intelligence and the enormous volume of available data that is rapidly increasing, as well as the application of deep neural networks to large amounts of data to learn a procedure aimed at handling a task. The task can range from simple classification to complex reasoning. One version of the goal or ambition behind AI is to enable a machine to surpass human brain performance. Deep learning is a means to achieve this goal [137].

In today's era, one of the most common uses of Deep Learning is within the broader field of Natural Language Processing [51]. This field has been revolutionized by Deep Learning with approaches and technologies far superior to those of a few years ago. These include recurrent neural networks (RNN), convolutional neural networks (CNN), and long short-

term memory networks (LSTM), which have been used to capture sequential dependencies in textual data.

Starting with the former, recurrent neural networks (RNN), which differ from feed-forward neural networks (where connections go in one direction), were introduced with connections that form a directed cycle. This allows RNNs to process sequences of data and maintain context, thus improving the understanding of word meanings.

However, despite these advantages, RNNs have some significant limitations. As cited in the previous rows one of the main issues is the vanishing gradient problem. The gradient is the measure of how much the loss function changes with respect to the model weights. During training, the backpropagation process uses these gradients to update the network weights, which, already small due to activation functions (which compress the input into a limited range), are further reduced to extremely small values. This makes it very difficult to learn long-term dependencies in deep networks, as relevant information is not effectively propagated through the network. Convolutional networks (CNN) suffer less from this problem due to their simpler and shallower structure. These networks have been adopted for NLP, especially for tasks requiring local feature extraction, such as text classification and named entity recognition. They use mathematical operations that combine multiple sets of information (called convolutions). In CNNs, this operation is performed by applying a weight matrix smaller than the input data (kernel filter), which through dot products between this filter and the portion of input covered by the filter produces feature maps. However, these networks are mostly used with images and are usually combined with RNNs for text data tasks.

To address the vanishing gradient problem, more advanced RNN models such as long short-term memory networks (LSTM) have been developed, which use gating mechanisms to maintain and update relevant information. Fundamentally, they use three types of gates to manage the flow of information: Input Gate, Output Gate, Forget Gate. The first one, determines which new information should be stored in the memory cell, the

second decides which information from the memory cell should be used as output, and the third establishes which information should be forgotten from the memory cell. The information is updated at each time step, allowing the network to maintain context and dependencies, and, during backpropagation, the gates help mitigate the gradient problem.

Nevertheless, LSTMs also have their issues, often related to their poor parallelization capability (since they process data sequentially) and low computational efficiency, especially for very long sequences.

In 2017, driven by the need for more efficient and scalable models for NLP, the so-called *Transformers* were introduced. These have decisively revolutionized the field of NLP. In the next section, we will understand why.

1.4. Transformers

For the first time, an architecture is proposed that avoids recurrence (where the output from previous steps is fed back into the network to process sequences of data and maintain context over time) and convolution (applying convolutional filters to input data to capture spatial hierarchies and patterns) and is entirely based on an attention mechanism to draw global dependencies between input and output [142]. Additionally, this model allows for significantly greater parallelization by processing all the words in a sequence simultaneously. This approach reduces the number of operations needed to relate signals from any two positions to a constant number, improving computational efficiency.

1.4.1. Architecture

The Transformer architecture consists of two main parts: the encoder and the decoder. Both are composed of a series of identical, but separate, layers.

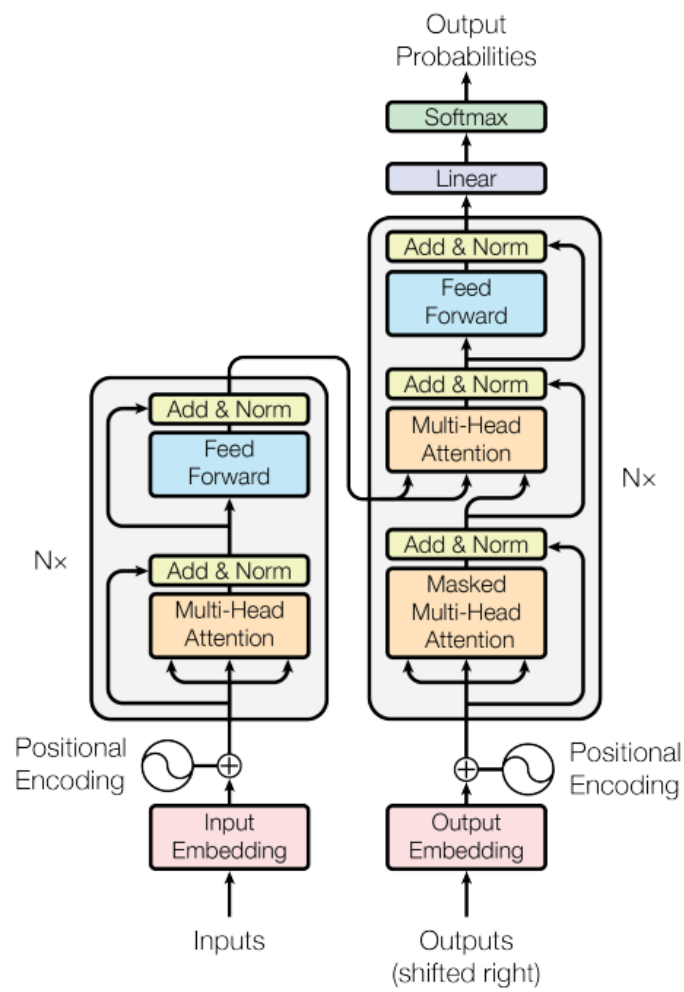


Figure 1.1: Transformers architecture

Encoder

The encoder is responsible for transforming the input into an intermediate representation. Each layer of the encoder consists of two main sub-layers:

- **Self-Attention Mechanism:** This sub-layer allows the model to focus on different parts of the input sequence for each position in the sequence.
- **FeedForward Network:** After self-attention, the data passes through a fully connected feedforward neural network, which applies non-linear transformations.

Each sub-layer is followed by a normalization¹ operation and a residual connection², which helps stabilize the training.

Decoder

The decoder transforms the intermediate representation produced by the encoder into an output sequence. Each layer is similar to that of the encoder, but with an additional sub-layer:

- **Self-Attention Mechanism:** Similar to the encoder, it allows the decoder to focus on different parts of the output sequence generated so far;
- **Encoder-Decoder Attention Mechanism:** This sub-layer allows the decoder to focus on the relevant parts of the input sequence, using the intermediate representations produced by the encoder.
- **FeedForward Network:** As in the encoder, the data passes through a feedforward neural network.

Here too, normalization operations are adopted.

Fundamentally, the encoder transforms a sequence of input symbols $(x_1, \dots, x_n)$ into

¹Normalization: A technique used to improve the stability and speed of neural network training by normalizing the output of a layer.

²A direct connection that skips one or more layers in a neural network, helping to mitigate the vanishing gradient problem and stabilize training.

a sequence of continuous representations $(z_1, \dots, z_n)$. From this sequence, the decoder generates an output sequence $(y_1, \dots, y_m)$ one element at a time. The model is thus autoregressive, meaning it uses previously generated symbols as input to generate the next symbol [142].

1.4.2. Attention mechanism

The attention mechanism on which these models are based allows assigning different weights to different parts of the input, determining which parts are more relevant for generating the output at a given moment. This is particularly useful for capturing long-range relationships and complex dependencies in the text.

This mechanism has also evolved, from the standard *Scaled Dot-Product Attention* to *Multi-Head Attention*.

Scaled Dot-Product Attention

This mechanism receives three types of input: queries (Q), keys (K), and values (V). Essentially, the query is the word for which information is being sought, the keys are the words in the sentence, and the values are the information associated with each word. The queries and keys have a dimension of d_k , while the values have a dimension of d_v .

To understand how relevant each word (key) is to the word of interest (query), the dot product between the query and each key is calculated. To prevent the numbers from becoming too large, the result of the dot product is divided by the square root of the key dimension ($\sqrt{d_k}$). A *softmax* function³ is applied to the scaled results, turning them into probabilities and assigning a weight to each word (forming the so-called attention weights). The higher the weight, the more important that word is to the query.

Finally, these attention weights are used to combine the values associated with the keys,

³Softmax function: A mathematical function primarily used in neural networks to convert a vector of values into a probability distribution.

producing a weighted representation of the input.

Multi-Head Attention

Multi-head attention is an extension of the standard attention mechanism, designed to enhance the model's ability to capture complex relationships in the data. With this approach, the input information (queries, keys, and values) is divided into smaller parts, and each of these parts is treated as a separate "attention head". It's like having multiple people read the same sentence, but each focusing on different aspects.

For each head, attention is calculated using the mechanism mentioned in the previous paragraph (*Scaled Dot-Product Attention*). This means that each head independently calculates the attention weights, focusing on different parts of the input.

Once these are calculated, the results are concatenated together. This creates a combined representation that incorporates different perspectives of the input. Finally, this combined representation is passed through a linear projection to obtain the final output. Essentially, this step helps to combine the information gathered from all the attention heads.

This approach allows the model to capture a wider range of relationships and dependencies, thereby improving its ability to generalize to new data. Additionally, by having reduced dimensions, it reduces the risk of the scalar product values becoming too large and thus encountering issues with the *softmax* function.

Among the *Transformer* models, this thesis will illustrate the model with *T5* architecture, which is at the center of the development of the work.

1.5. Text-to-Text Transfer Transformer (T5)

The *T5* (*Text-to-Text Transfer Transformer*) [113] model was developed by *Google AI* and introduced in 2019. This model was designed as a *Transfer Learning* methodology⁴, to unify various NLP tasks into a single text-to-text framework, where both inputs and outputs are treated as text string [113].

In the same year, *GPT-2* was developed by *OpenAI* as a generative language model based on *Transformer*, following the introduction of *BERT* [142] in 2018, which revolutionized the NLP field with the “Masked Language Modeling” technique that allows the model to learn the bidirectional context of words in a sentence. Following these developments, *T5* was introduced with this unified approach. This model was pre-trained on a vast dataset called *Colossal Clean Crawled Corpus* (*C4*), which contains hundreds of gigabytes of clean data from the publicly available web archive “Common Crawl”, which produces about 20 TB of text data each month. However, much of this data is often incomprehensible and not in natural language, so it was cleaned using specific criteria that we will not mention here. This pre-training allowed the model to acquire a general understanding of the language, which can then be adapted to specific tasks through fine-tuning. The basic idea, as we have already mentioned, is to treat every text processing problem as a “text-to-text” problem, i.e., taking text as input and producing new text as output. Essentially, with this framework, it is possible to apply the same model, objective, training procedure, and decoding process to every task considered. Like the original *Transformer*, *T5* uses an *encoder-decoder* architecture, where an input sequence of tokens is mapped to a sequence of embeddings, which is then passed to the encoder, which, as described in the previous section, consists of a self-attention layer followed by a small feed-forward network. A simplified version of the layers is used where the ac-

⁴Transfer Learning: a machine learning technique where a model trained on one task is reused as a starting point for a related task. This approach is particularly useful when there are large amounts of data for pre-training but only a limited amount of data for the specific task.

tivations are only scaled. After this, a residual skip connection adds the input of each subcomponent to its output. Within the feed-forward network, on the skip connection, on the attention weights, and at the input and output of the entire stack, Dropout⁵ [127] is applied to prevent the overfitting⁶. Like the original *Transformers*, the decoder has a similar structure to the encoder but includes a standard attention mechanism after each self-attention layer, which focuses on the encoder’s output. This mechanism in *T5* is causal, allowing the model to consider only previous outputs, thus ensuring temporal coherence in sequence generation. After each self-attention layer, a standard attention mechanism is applied that focuses on the encoder’s output, so the decoder uses relevant information from the input sequence. The final output of the decoder passes through a dense layer with a *softmax* output, whose weights are shared with the input embedding matrix. This approach ensures consistency between input and output representations.

1.5.1. Flan-T5

In this thesis, an improved version of *T5*, *Flan-T5* (*Fine-tune Language Net*) [156], will be used. *Flan-T5* is available in various sizes, including small, base, large, XL, and 11B parameters. Therefore, it is important to illustrate this advanced approach as well. It has been shown that increasing the size of these language models does not automatically make them better at following user intent [98]. For example, large models can generate outputs that are false and unhelpful. For this reason, an approach was created to align language models using human feedback. This process includes collecting demonstrations of desired behavior from labelers and using these demonstrations to train the model through supervised learning. Subsequently, model response ratings are collected to further refine the model using reinforcement learning based on human feedback. This process

⁵Dropout: is a technique that randomly removes units (along with their connections) from the neural network during training, preventing units from co-adapting and thus preventing overfitting.

⁶Overfitting: phenomenon which occurs when a machine learning model learns the training data too well, memorizing various peculiarities of the training data rather than finding a general predictive rule [46]

improves generalization to unseen tasks, allowing the model to perform tasks it was not explicitly trained on. This ability to follow instructions becomes very similar to how humans approach problem-solving. Generally, these checkpoints have strong zero-shot, few-shot, and CoT (Chain of Thought⁷) capabilities, surpassing previous public *T5* checkpoints (Scaling Instruction-Finetuned Language Models [40]). For example, *Flan-T5 11B* outperforms *T5 11B* with double-digit improvements and even surpasses PaLM (Pathways Language Model)⁸ 62B in some challenging BIG-Bench tasks [126]. *Flan-T5* is based on the *T5* (C4) model with additional training using *instruction fine-tuning*. As described in Scaling Instruction-Finetuned Language Models [156], this final fine-tuning was performed on 1,836 tasks, mixing different previous works:

- Muffin: 80 tasks including 62 from Wei et al. [156] and 26 tasks including dialogue data (Byrne et al., 2019 [26]; Anantha et al. [14], Dai et al. [44])) and program synthesis (Yasunaga and Liang [173], Li et al. [77]);
- T0-Sans-Flan: 193 tasks derived from the model with which T0 was trained (Sanh et al. [117]);
- NIV2: 1554 tasks from Wang et al. [154] in which 44 tasks related to MMLU (Massive Multitask Language Understanding) were removed because this task was used for evaluation;
- CoT: 9 datasets that include step-by-step reasoning annotations for complex tasks. These include activities such as arithmetic reasoning (Cobbe et al. [41]), multi-hop reasoning (Geva et al. [54]), and natural language inference (Camburu et al. [31]).

The fine-tuning process was standardized for all models, with some variations in hyperparameters such as learning rate, batch size, dropout, and the number of fine-tuning

⁷Chain of Thought: a method that guides the model through a series of intermediate logical steps to arrive at a conclusion, instead of asking the model to directly provide the answer to a complex problem, such as in a math problem.

⁸PaLM: a large language model developed by Google, designed to perform a wide range of NLP tasks with cutting-edge performance.

steps. The Adafactor optimizer [122]) with a constant learning rate was used to optimize training. An interesting technique used during fine-tuning is “packing” (used in the *T5* standard), which allows combining multiple training examples into a single sequence, separating inputs from targets with an end-of-sequence token. To prevent tokens from influencing each other across the boundaries of packed examples, masking is applied. During fine-tuning, a single checkpoint was used for all evaluations for each model. The optimal step was chosen based on periodic evaluations of validation tasks conducted every 2000-10000 steps depending on the model size.

1.5.2. Application tasks

This type of language model and its variants (*T5 architecture*) have been designed to tackle a wide range of NLP tasks. As already mentioned, its strength lies in its unified approach, which allows it to treat each task as a “text in input” and “text in output” problem [113]. This feature makes it a powerful and scalable tool that has achieved state-of-the-art results in numerous tasks, ranging from automatic translation to text generation, classification, and complex natural language tasks such as question answering, summarization, and sentiment analysis where a deep semantic understanding of the text is required.

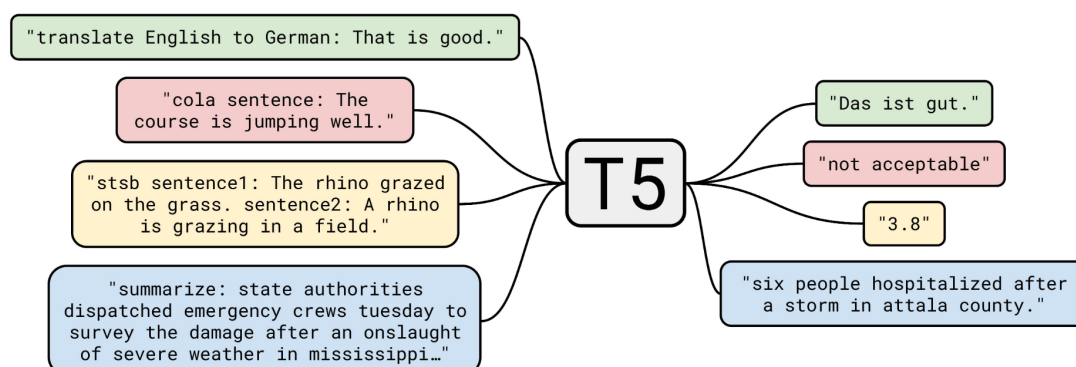


Figure 1.2: Task examples with T5

Automatic Translation

Automatic translation (MT) is one of the most studied tasks in the field of NLP and concerns the ability of a system to automatically translate text from one language to another. In recent years, deep learning models based on Transformer architectures have significantly improved the quality of these translations, surpassing traditional approaches. In particular, thanks to its architecture, *T5* can produce fluid and coherent translations, avoiding the limitations of more traditional models that focus more on syntactic analysis than on natural text generation. In fact, for this task, the text in a source language is provided as input, and the model generates the translation in the target language as output. The results shown by Raffel et al. [113] are remarkable, particularly the results obtained on benchmarks such as Workshop on Machine Translation (WMT) [22] and International Workshop on Spoken Language Translation (IWSLT) [50]. The advantage of using *T5* is its generalization, which allows it to compete with models specialized in translations, such as *mBART* (Liu et al. [83]). Although *mBART* excels in handling multiple languages, *T5* has demonstrated greater flexibility thanks to its text-to-text architecture, which allows it to tackle the translation task without needing to be specifically adapted for it.

Text Summarization

Automatic summarization is a fundamental task in many NLP applications, such as information retrieval or document processing. This model is particularly powerful for abstractive summarization, where the model does not just extract the most important sentences but creates a complete synthesis of the text. Thanks to its encoder-decoder architecture, it can generate coherent and well-formulated summaries, highlighting the model's success in summarization tasks compared to other approaches. The input is provided with the full text, preceded by a prefix like “summarize:”, and the model generates the summary as output. The results reported in [113] are impressive, demonstrating high

ROUGE scores on standard benchmarks. Specifically, the model excels on the CNN/-Daily Mail dataset, which consists of news articles used to generate concise summaries, and the Gigaword dataset, which focuses on very short summaries, often reduced to a single sentence..

Question Answering

Question answering (QA) is another fundamental NLP task, which involves providing a direct and accurate answer to a question based on a given context. Here too, the question and context are provided as input, preceded by a prefix like “question:”, and the model generates the answer as output. Its flexibility has allowed the model to adapt to different QA variants such as Closed-book QA (answering questions without an explicit context, based solely on knowledge learned during training) and Open-domain QA (answering questions using information provided in a context).

Unlike traditional models that focused on extracting answers directly from the context, *T5* generates the output in a simple yet powerful way: the context is provided along with the question (e.g., “*question: What is the capital of France? context: France is a country in Europe with Paris as its capital.*”), and it generates the answer “*Paris*” as output. Thanks to this architecture, the model can understand and connect information, producing natural and coherent answers. The results on benchmarks such as SQuAD [114], Natural Questions [70], and TriviaQA [67] have demonstrated how competitive *T5* is. Although models like *BERT* and *RoBERTa* [82] are extremely effective in extractive QA tasks (where the answer is a portion of the provided text), *T5* excels in generative QA, where greater synthesis and reformulation capabilities are required. A particularly interesting aspect is the comparison with large-scale models like *GPT-3*. While *GPT-3* excels in generating fluid responses thanks to its enormous size, *T5* often proves to be more precise, especially in contexts where detailed understanding of the context is required. Additionally, *T5* is more interpretable and manageable in terms of resources,

making it a practical choice for many applications.

Thanks to its flexibility and precision, *T5* has become an ideal solution for QA systems in academic, medical, and business fields.

Text Classification

Text classification, which involves assigning one or more categories/labels to a given text, is another central application task for *T5*. Although the model treats each task as text generation, classification is formulated as the generation of a label (e.g., a category or a probable class of the text). In Raffel et al. [113], *T5* was compared with other models on various benchmarks, including GLUE (General Language Understanding Evaluation) [148], which includes several text classification tasks. The results showed that *T5* outperformed previous models like *BERT* and *RoBERTa* in multiple tasks, including Natural Language Inference (NLI) and sentiment analysis. In Wang et al. [150], which examined the performance of different models on SuperGLUE [149], the model outperformed *RoBERTa* in some classification tasks, such as WiC (Word-in-Context) [106], which requires deeper semantic understanding.

Sentiment Analysis

Another crucial task, though part of the broader text classification domain, is sentiment analysis. This task, which has gained much attention in the last decade especially in the context of social media and online reviews aims to determine the sentiment or emotional tone expressed in a given text. Typically, sentiment analysis involves categorizing input text into predefined sentiment classes such as “positive”, “negative”, or “neutral”. With *T5*, sentiment analysis is approached primarily in a generative manner, leveraging its flexibility and ability to process inputs and generate outputs in natural language.

As with other tasks, this formulation simplifies the pipeline while allowing the model to adapt to various sentiment analysis variants, including fine-grained sentiment classi-

fication (e.g., rating levels) and aspect-based sentiment analysis, where sentiments are assigned to specific aspects of the text. This versatility is particularly relevant because sentiment analysis will be the focus of this thesis and will be explored in greater detail in the following chapters.

T5 has demonstrated competitive performance on benchmark datasets such as SST-2 [125], Yelp Reviews [186], and IMDB [90]. Its ability to adapt to domain-specific sentiment analysis tasks makes it a versatile tool. Compared to other transformer-based models like *BERT*, which excels in token-level classification tasks, *T5* performs better in scenarios requiring a combination of generative output and deeper contextual understanding.

2 | Sentiment Analysis

2.1. Introduction to the Sentiment Analysis

Sentiment analysis (SA) is one of the most popular and studied tasks in the field of NLP, particularly within the broader task of text classification. As mentioned in the previous chapter, sentiment analysis, also known as *opinion mining*, aims to identify and classify opinions, emotions, or attitudes expressed in a text. Its applications range from analyzing product and service reviews to monitoring public opinion on social media, and even advanced applications within business decision-making processes.

The main goal of sentiment analysis is to determine the polarity of the text, typically classifying it into categories such as *positive*, *negative*, or *neutral*. In more complex contexts, the task can extend to extracting specific emotions (e.g., joy or anger) or understanding the intensity of opinions.

Opinions play a central role in almost all human activities, influencing behaviors and decisions [80]. From choosing a product to expressing political preferences, the ability to access others' opinions has become increasingly essential. Traditionally, people sought advice from friends and family, while companies relied on surveys and focus groups to gather feedback. However, the explosion of social media and online platforms has radically transformed this process. Today, reviews, forum discussions, blog posts, and social network comments provide an unprecedented amount of opinion data accessible to anyone. This abundance of information, while valuable, presents significant challenges: the vastness of the content makes it difficult for a human to effectively identify, extract, and

synthesize relevant opinions. Therefore, due to the extensive collection of opinions on the web, some form of opinion synthesis has become necessary (Hu and Liu [63]). This is where automated sentiment analysis, a set of techniques and tools capable of classifying and interpreting sentiments expressed in textual data, comes into play.

Early research on sentiment analysis dates back to the 1990s [57, 60, 157–159], when scholars began exploring how computers could detect and classify emotions and opinions in text. However, this research was very limited, both due to the complexity of the necessary techniques and the lack of large-scale data. With the acceleration of the digital era in the early 2000s, the field underwent significant expansion. The availability of vast amounts of online data made it possible to develop more advanced approaches and allowed companies to recognize the commercial potential of sentiment analysis. Studies on product review classification, such as those by Pang et al. [99] and Turney [139], demonstrated the effectiveness of these approaches, paving the way for numerous innovative applications. Beyond the commercial sphere, sentiment analysis has had a profound social impact. For example, during the Arab Spring of 2011, social media posts mobilized masses and influenced political changes. In academia, sentiment analysis has continued to evolve with significant contributions, such as using Twitter to correlate sentiment with public opinion polls (O’Connor et al. [96]), predicting box office revenues based on reviews and social media (Asur and Huberman [17]), and analyzing emotions in literary texts (Mohammad [93]).

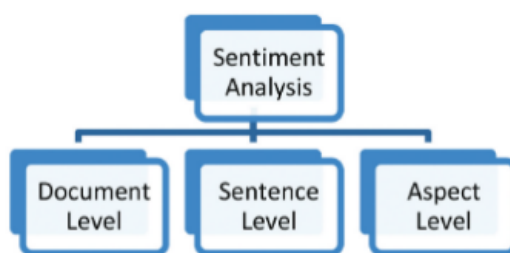
The evolution of sentiment analysis reflects a convergence between practical needs and academic interests, transforming it into an indispensable tool not only for understanding the contemporary digital world but also for addressing the complex challenges of modern society.

In Chapter, the evolution of methods up to the introduction of Transformer models, such as *T5*, was discussed. As reiterated multiple times, this type of model has revolutionized the way the task is approached. Compared to traditional approaches, based on

rules or standard machine learning techniques, Transformers offer greater generalization capacity and better capture the semantic nuances of natural language. In the next sections of the chapter, the problem definition will be described in detail, followed by the evolution of techniques for this task, delving into the techniques that have led to the state of the art.

2.2. Levels of Sentiment Analysis

Sentiment analysis is mainly divided into three different levels of granularity [80]. These levels are commonly categorized into: document-level analysis, sentence-level analysis, and aspect-level analysis.



2.2.1. Document level analysis

The task at this level focuses on identifying the overall sentiment expressed in an entire text [99, 139]. This approach is useful when documents express opinions about a single entity, such as a restaurant review that may conclude the document expresses a positive sentiment based on the overall tone. A different situation arises if the same document contains contrasting opinions, for example, in a review that praises the food but criticizes the service. Document-level analysis might return a mixed result, which is not very useful for detailed analysis. At this level, both supervised and unsupervised learning approaches can be used to classify the document.

2.2.2. Sentence level analysis

When a document contains multiple opinions, sentence-level analysis offers greater granularity. In this case, each sentence is treated as a separate unit, allowing the identification of contrasting opinions within the same document. This is particularly useful when a document presents a mix of sentiments. Sentence-level analysis is associated with subjectivity classification [39, 159], which distinguishes sentences expressing subjective opinions and viewpoints from those expressing factual information, the *objective sentences*. Each sentence's polarity is determined independently using the same methodologies as document-level analysis but with more training data. In this way, the polarity of each sentence can be aggregated to find the document's sentiment or used individually. Sentence-level analysis can be crucial when document-level analysis is insufficient, such as when dealing with conditional sentences or ambiguous statements [155].

2.2.3. Aspect level analysis

Also called Aspect-based sentiment analysis (ABSA), which will be discussed extensively in this chapter as it is the research field on which this thesis is based, the analysis is performed for each aspect mentioned in the sentence. This level of granularity allows understanding what people liked and disliked, limitations found in the other two levels mentioned above. This approach does not limit itself to identifying the overall or sentence-level sentiment but associates the sentiment with specific aspects or entities mentioned in the text. For example, in the sentence “*The phone's display is excellent, but the battery life is disappointing.*”, it evaluates two aspects, the “*phone's display*” and the “*battery life*” of the phone (entity). The sentiment on the “*phone's display*” is positive, but the sentiment on the “*battery life*” is negative. In this way, it is possible to summarize opinions on entities and their aspects in a structured manner, useful for conducting qualitative and quantitative analyses. This task requires advanced techniques for the automatic identification of aspects of multiple elements within the text and is

the subfield with the most research evolution. Indeed, it will be the main topic of this thesis. It is important to mention that this is the level of granularity that companies are now interested in because it offers high practical utility for improving various business aspects, from research and development to marketing, to the overall improvement of customer experience, as well as responding promptly to market needs. Before delving deeply into aspect-based sentiment analysis, let's take a brief overview of the sentiment analysis techniques that have been developed over the years.

2.3. Evolution of Sentiment Analysis Methods

Following the methodologies developed over the years in the field of Natural Language Processing (NLP), techniques and approaches to tackle the task of sentiment analysis have evolved. In the following sections, we briefly describe the approaches that have led to the state of the art, specifically leading us to the field of Aspect-based sentiment analysis and the multiple tasks that derive from it.

2.3.1. Traditional Lexicon-based Approaches

The first approaches to sentiment analysis were based on lexical methods and predefined rules. A lexicon is a set of words and their properties, such as meaning, usage, and in this case, emotional polarity. Lexicon-based approaches use these resources to map words to numerical values, often representing sentiment polarity (positive, negative, neutral) [155]. These scores are assigned to each word, and through an aggregation process, the overall sentiment of a sentence or an entire document is determined. These approaches have the advantage of being quick to apply as no training data is needed, but they could be considered an unsupervised technique. On the other hand, the main disadvantage lies in domain dependency, as a positive word in one domain can be negative in another.

Corpus-based Approaches

Corpus-based approaches are based on the idea of building a sentiment lexicon from a large set of data, i.e., a corpus of texts, to deduce the polarity of words through *statistical* and *semantic* techniques. The key idea is that the context in which a word appears can provide clues about its polarity. For example, in positive texts, terms like “*fabulous*” tend to co-occur frequently with other positive words. A common approach is to analyze the co-occurrence frequency of terms: if a word often appears next to terms known to have positive polarity, it is likely that it also has the same orientation. This technique was explored by Turney and Littman [140], who used statistical measures to calculate co-occurrence relationships between words. Another method, of a semantic nature, relies on tools like WordNet, which allows the identification of synonyms and antonyms. Here, words are grouped based on their semantic similarity: for example, synonymous terms like “*fantastic*” and “*excellent*” will have similar polarity, while antonymous terms like “*excellent*” and “*terrible*” will have opposite polarity. This approach also allows expanding the lexicon through automatic deductions, overcoming the limitations of static dictionaries. However, it requires a large corpus and computational resources to effectively analyze the context of words.

Dictionary-based Approach

The dictionary-based approach uses predefined lists of words that have already been assigned a polarity value. These lists can be created manually or derived from resources like Thesaurus¹ or WordNet². The process of building a dictionary often starts with an initial set of manually annotated words, which is then expanded using synonyms and antonyms found in lexical resources. For example, starting from the word “*good*”, synonyms like “*excellent*” and “*great*” can be added as positive terms, and antonyms

¹Thesaurus: a lexicon of terms related to a general or specific field of knowledge, connected in a hierarchical and relational network

²WordNet: a lexical-semantic database for the English language that aims to organize, define, and describe the concepts expressed by words

like “*bad*” and “*poor*” as negative terms. Resources like SentiWordNet [18] are well-known examples of dictionaries built in this way. This approach is very useful where simplicity and transparency are fundamental. However, one of the main problems with this approach is that, being static, a dictionary may not be adequate for all domains. Additionally, identifying new sentiment words is complex and requires manual or semi-automatic updates.

2.3.2. Machine Learning Approaches

With the introduction of Machine Learning (ML) algorithms, sentiment analysis has made significant progress, allowing models to be trained directly on data and classify texts based on numerical representations of their features. Although supervised learning is the most common approach, unsupervised methods are also applied in specific contexts, especially when labeled data is not available. Supervised methods address sentiment classification as a common text classification problem, leveraging syntactic and/or linguistic information. A common representation of texts in this context is the *bag-of-words (BoW)*, which converts a document into a numerical matrix based on the frequency or presence of words, without considering their order. This technique, while simple, represents one of the first steps towards automating text analysis and is used as input for models such as Support Vector Machine (SVM), Random Forest (RF), and K-Nearest Neighbors (KNN). Sentiment analysis algorithms are trained to interpret not only literal meanings but also contextual elements such as sarcasm or ambiguous terms, albeit with limitations due to the loss of global context in BoW-based models. The unsupervised approach relies on the analysis of unlabeled data, using techniques such as clustering or association to identify hidden patterns in texts. These methods often rely on linguistic resources such as ontologies, semantic databases, and predefined lexicons to determine sentiment. Methods based on clustering, such as K-means, and those of latent semantic analysis (LSA) fall into this category.

2.3.3. Hybrid Approaches

Hybrid approaches combine the power of Machine Learning models with lexicon-based linguistic resources, aiming to leverage the strengths of both methods. This integration allows exploiting the predictive power of statistical algorithms along with the semantic knowledge contained in lexicons, ensuring greater robustness. Lexicons, such as the already mentioned in Introduction Section, provide a predefined reference for determining word polarity, while machine learning models analyze more complex patterns in the data, learning directly from numerical features.

2.3.4. Neural Networks Approaches

Initially, traditional machine learning methods dominated the field but presented several limitations. These models required intensive manual work to extract useful features from textual data and had difficulty capturing the context of sentences. Moreover, they were particularly vulnerable to data imbalances and struggled to generalize when applied to new domains or languages. The introduction of neural networks, such as Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) [61] and later GRU (Gated Recurrent Unit), introduced a completely new way of handling text, shifting the focus from manual feature extraction to automatic learning of representations directly from data. With the advent of Word Embeddings³ like Word2Vec and GloVe, models were able to represent words in a continuous space, capturing semantic similarities and contextual relationships. Recurrent networks (RNN) and their variants (LSTM and GRU) allowed processing text as a sequence, remembering previous information to build a global meaning. These models addressed many of the shortcomings of machine learning methods, but not without problems. As mentioned in Chapter 1, traditional RNNs struggled with handling long-term dependencies due to the *vanishing gradient* phenomenon,

³Word Embeddings: an NLP technique that allows storing both semantic and syntactic information of words starting from an unannotated corpus and building a vector space where word vectors are closer if the words occur in the same linguistic contexts, i.e., if they are recognized as semantically more similar.

thus limiting their ability to analyze complex and long texts.

2.3.5. The advent of Transformers

It is in this context that *Transformers* revolutionized the field, representing a milestone for sentiment analysis and NLP in general. Thanks to the self-attention mechanism, these models can capture global relationships between words without depending on the sequential structure of the data. This makes them not only more powerful but also more efficient, as they can process information in parallel. Models like *BERT*, *RoBERTa*, and *T5* have demonstrated exceptional performance on benchmarks such as IMDB [90], SST-2 [125]. A key aspect that accelerated their adoption was the introduction of Transfer Learning, which allowed overcoming the main obstacle of deep learning, namely the need for large amounts of annotated data.

3 | Aspect-Based Sentiment Analysis

Aspect-Based Sentiment Analysis (ABSA) refers to the task of identifying and analyzing the sentiment associated with each aspect mentioned within a text. However, the term ABSA is not limited to this specific task but represents, more generally, the entire field of fine-grained sentiment analysis, which includes a variety of sub-tasks. These sub-tasks involve identifying various sentiment elements at the aspect level, such as aspect term extraction, opinion term identification, aspect categorization, and polarity assignment. Consequently, we consider ABSA as a research field and will delve into the various sub-tasks, with particular interest in some of these, which form the basis of this thesis.

ABSA plays a crucial role in the business context, especially in sectors such as retail, hospitality, marketing, and technology. It allows companies to analyze customer feedback in detail, identifying areas for improvement and optimizing products and services. For example, in the e-commerce sector, ABSA helps understand opinions on different aspects of a product, such as “quality”, “price”, or “ease of use”.

In this section, ABSA will be discussed both as a specific task and as a broader research field, exploring related sub-tasks and its impact on various business sectors. The discussion will delve into common research approaches and methodological challenges, highlighting how this technology can transform business operations and improve customer experience. To do this, the logic and nomenclature adopted in the survey by W. Zhang et al. [185] will be followed.

Identifying aspects in a text and determining their sentiment has been a challenge over the years. Early approaches to ABSA focused on identifying each sentiment element separately. Two of the most relevant initial tasks included aspect term extraction (ATE) and aspect sentiment classification (SC). The ATE task aimed to identify specific terms representing aspects in a text, while the sentiment classification task focused on predicting the sentiment polarity associated with a specific aspect.

For example, in a sentence like *“The pizza is delicious”*, the term *“pizza”* is an aspect, while *“delicious”* represents the positive sentiment towards that aspect. However, fully understanding a person’s opinion requires a more advanced approach that not only extracts individual elements but also explores their relationships. To this end, composite ABSA tasks have emerged in recent years, dealing with the simultaneous prediction of multiple elements, allowing the extraction of not only aspects and sentiment but also the connections between them. In parallel, the development of pre-trained language models has led to advancements in this field.

Therefore, the primary distinction in Aspect-Based Sentiment Analysis is between tasks that extract of a single element and those that extract multiple elements.

3.1. ABSA elements

Regardless of the type of task, in general, we can find four elements within these:

- Aspect (*a*): An aspect represents the entity, object, or term that refers to an aspect in the given text. For example, in the review:

“The pizza was delicious, but the service was slow”

The aspects are *“pizza”* and *“service”*;

- Category (*c*): The aspect category represents a predefined class that defines an aspect of an entity or product. These categories are used to classify aspects within

a predefined set, chosen based on the domain of interest. Each category represents a thematic area that may be relevant for sentiment analysis. For example, “AMBIENCE”, “PRICE”, “FOOD” could be categories for the restaurant domain;

- Opinion terms (*o*): The opinion term is the expression that the author of the text uses to express a sentiment or evaluation towards an aspect. Returning to the previous example, “*The pizza was delicious*”, the opinion terms are “*delicious*” and “*slow*”, which respectively expresses a positive evaluation of the aspect term “*pizza*” and a negative evaluation of the aspect term “*service*”.
- Sentiment Polarity (*s*): Sentiment polarity describes the orientation of the sentiment towards an aspect term or category. It can be positive, negative, or neutral. Polarity helps determine the overall evaluation of the aspect expressed in the text. For example, in the sentence “*the pizza was delicious*”, the polarity would be positive, while in the sentence “*the service was slow*”, the polarity would be negative.

3.2. Single ABSA tasks

As mentioned earlier, the first subdivision we can make is to divide ABSA sub-tasks into individual and multiple. The former focus on a single analysis element, such as extracting an aspect term (e.g., “*pizza*”) or classifying a sentiment regarding a specific aspect (such as the positive or negative sentiment associated with the word “*pizza*”). In the following sections, the sub-tasks belonging to this category will be briefly described along with the research.

Table 3.1: Single ABSA Tasks

Task	Input	Output
Aspect Term Extraction (ATE)	sentence	pizza (a), service (a)
Aspect Category Detection (ACD)	sentence	food (c), service (c)
Aspect Opinion Co-Extraction (AOCE)	sentence	{pizza (<i>a1</i>), service (<i>a2</i>)}, {delicious (<i>o1</i>), slow (<i>o2</i>)}
Target-oriented Opinion Words Extraction (TOWE)	sentence, pizza (<i>a1</i>)	delicious (<i>o1</i>) slow (<i>o2</i>)
Aspect Sentiment Classification (ASC)	sentence, pizza (<i>a1</i>) sentence, service (<i>a2</i>)	positive(p) negative(p)

3.2.1. Aspect Term Extraction (ATE)

This task aims to extract terms representing aspects from a text, such as “*pizza*” and “*service*” in the example given in the previous section. Supervised methods, central to aspect term extraction, have evolved through various studies. For example, Yin et al. [175] leveraged grammatical dependency paths to enhance lexical representations, while Xu et al. [166] proposed a CNN model based on double embeddings, one general and one domain-specific. Additionally, Xu et al. [167] introduced fine-tuning of *BERT* on domain-specific data, improving the model’s ability to understand context.

Innovative architectural designs have been introduced by authors such as Wang et al. [152] who modeled the relationships between aspects and opinions, or Ma et al. [88], who transformed aspect term extractions into a *Seq2Seq* problem, allowing a broader understanding of contextual meaning.

Several semi-supervised approaches have been developed to overcome the need for large amounts of data in supervised methods. These methods combine labeled datasets with unlabeled datasets. Strategies such as progressive self-training by Wang et al. [151], or the

conditional generation of pseudo-labeled sequences proposed by Li et al. [73], represent significant advances. Additionally, Chen and Qian [38] introduced soft prototypes to improve the model’s efficiency in recognizing new terms.

Unsupervised methods, on the other hand, aim to extract aspect terms without annotated data. Among, He et al. [58] proposed ABAE (Attention-Based Aspect Extraction), an autoencoder that uses attention to improve the coherence of extracted aspects. Luo et al. [86] introduced the use of sememes to enrich lexical representations, while Liao et al. [79] combined local and global contexts to discover aspect terms.

A simpler approach, like that of Tulkens and van Cranenburgh [138], used only a POS tagger to select candidate aspects and subsequently a contrastive attention mechanism to confirm relevant aspects.

3.2.2. Aspect Category Detection (ACD)

Aspect Category Detection involves identifying aspect categories within a text, as well as the classes to which aspect terms belong. In other words, while Aspect Term Extraction (ATE) focuses on identifying terms that represent aspects (such as “pizza” or “service”), ACD focuses on assigning a specific category to these aspects, such as “food” or “service”. This has the dual advantage of allowing for a more aggregated analysis rather than single aspect analysis (when not explicitly mentioned in the text). Another advantage is that we can detect categories even when they are not explicitly mentioned in the text.

In the literature, approaches are divided into supervised and unsupervised. In the first case, ACD is generally formulated as a multi-label classification problem, where each aspect category is treated as a separate label. For example, one of the earliest approaches, developed by Zhou et al. [189] uses a model based on word embeddings to handle noisy data, improving the ability to identify aspect categories in online reviews. Additionally, feed-forward networks has allowed the extraction of hybrid features that

improve categorization accuracy.

Subsequently, models that leverage attention mechanisms have improved performance by focusing on different parts of the text based on aspect categories, as done by Movahedi et al. [94]. Specifically, this work proposes an innovative approach, focusing on text sections relevant to each category. Word co-occurrence patterns have also been used to improve category detection by exploiting the relationship between aspect terms and their contexts (Schouten et al. [120]). Another important approach was presented by Ghadery et al. [55]. In this case, a matching method between the text and a set of representative words for each category was introduced, further improving the model's reliability in identifying categories.

Regarding unsupervised approaches, one of the foundational works is by R. He et al. [59] which forms the basis of unsupervised approaches. The key idea is that, once aspect terms are extracted using a neural architecture with an attention mechanism, classification or clustering can be performed to categorize them without requiring direct supervision or predefined labels for training. Labels can be manually assigned to each cluster or, more effectively, inferred using cosine similarity between the sentence vector and the category vector [138]. Shi et al. [123] introduced High-Resolution Selective Mapping (HRSMap), a technique that improves the association of automatically detected aspects with predefined categories. It uses advanced similarity and filtering strategies to ensure greater accuracy in categorization, reducing errors and making the process more effective compared to traditional methods.

3.2.3. Opinion Term Extraction (OTE)

Opinion Term Extraction (OTE) involves identifying words within a text that express a specific aspect. Returning to our example, “*the pizza was delicious but the service was slow*”, OTE's task is to extract the opinions, which in this case would be “delicious” and “slow”. Since opinion terms and aspects frequently coexist, extracting only opinion terms

without considering the related aspect is often not very meaningful. Consequently, OTE is often divided into the following two main tasks.

Aspect Opinion Co-Extraction (AOCE)

In this task, the goal is to simultaneously extract aspect and opinion terms. For example, in the example sentence from the table, the output would include the aspects “*pizza*” and “*service*”, along with the opinions “*delicious*” and “*slow*”. Although it involves two distinct elements, AOCE is considered a joint extraction task, but not relational, as the relationship between terms (e.g., “*delicious*” describing “*pizza*”) is not explicitly modeled. AOCE is often formulated as a token-level classification problem, using specific labels to identify the beginning (*B*) or inside (*I*) of an aspect (*-A*) or an opinion (*-O*), or *N* representing neither.

To address this problem, there have been mainly two strategies. Separate label approaches assign distinct sets of labels to separately identify aspect and opinion terms. For example, Yu et al. [177] proposed a multitask neural network (MTNN) model designed to simultaneously infer these two types of terms while keeping their categories separate. This framework shares hidden layers between the two tasks but uses separate, task-specific output layers to predict labels for each. In this way, the model can leverage shared information between the two tasks, improving overall performance while maintaining the flexibility to assign different labels to each word based on its role in the sentence context.

Alternatively, there are unified label approaches that combine information about aspects and opinions into a single representation. For example, Wu et al. [161] used a unified label scheme based on the BIO (Beginning, Inside, Outside) coding to represent both types of terms in a single sequence. This approach, integrates deep neural networks with a weighted MaxSAT layer¹, incorporating domain knowledge in the form of logical

¹MaxSAT layer: a component in neural networks that integrates domain knowledge through logical rules, optimizing the satisfaction of these rules to improve model performance.

rules. This allows the model to capture relationships between aspects and opinions more effectively.

Target-oriented Opinion Words Extraction (TOWE)

This task focuses on extracting opinion terms associated with a specific predefined aspect. For example, given the aspect “*pizza*” in the sentence “*the pizza was delicious*”, the model should predict the opinion term “*delicious*”, thus with already known aspect terms. It is also formulated as a token-level classification problem, but with the main goal of capturing specific representations for the given aspect. The input is represented by the sentence and the aspect term, while the output is a sequence of labels identifying the words that form the opinion.

Here too, various approaches have been used. Fan et al. [48] introduced a model that incorporates aspect information using an *inward-outward (IOG)* mechanism based on LSTM to generate an aspect-enriched context. This dual encoding process allows aspect information to propagate throughout the sentence and generate aspect-specific contextual representations for each word. Z. Wu et al. [164] proposed a model called Latent Opinions Transfer Network (LOTN) that transfers knowledge from latent opinions in review classification datasets, which are rich in information, to the TOWE task, which is low-resource. The attention mechanism in this model helps extract potential opinion words through probabilistic weights. To adapt this global information for TOWE, a transformation method was designed to convert the global attention distribution on words into target-dependent latent opinions, by incorporating positional information of the target and surrounding words. considering the position information of the target or other words. Veyseh et al. [145] proposed a model called ONG (Ordered-Neuron LSTM and Graph Convolutional Neural Networks) that incorporates the syntactic structures of sentences into the model’s computation for the task. They used dependency trees to model the syntactic distance between the aspect term and opinions, providing a structural indication of

which terms are more likely to be opinion expressions. ONG uses two types of syntactic information: syntax-based possibility scores (integrated with ON-LSTM) and syntactic connections between words (applied with Graph Convolution Networks). To further improve performance, ONG introduces a regularization technique that distinguishes word representations in TOWE. Additionally, Mensah et al. [92] studied the importance of positional embeddings in TOWE and found that BiLSTM-based methods show an effective inductive bias for the task. They observed that adding graphical structures such as graph convolutional networks (GCN) only brings marginal improvements.

3.2.4. Aspect Sentiment Classification (ASC)

This task concludes the series of tasks for predicting single elements in the field of ABSA. Aspect Sentiment Classification (ASC) is the task that determines the polarity (e.g., positive, negative, and neutral) of an opinion expressed in relation to a specific aspect. Essentially, it receives the same inputs found in OTE, such as a sentence or text containing aspect terms (or possibly aspect categories), and the model must predict the sentiments associated with those aspects (or categories).

The introduction of neural networks has revolutionized the approach to this task, replacing traditional methods based on manual feature extraction with models capable of autonomously learning contextual and semantic relationships. Among the various interesting works is that of D. Vo and Y. Zhang [146], which focuses on extracting a rich set of automatic features without relying on syntactic analysis (which can be problematic with tweets, the subject of the work). The system divides a tweet into a left and right context relative to a given target and uses distributed word representations and neural pooling functions to extract features. Both sentiment-guided embeddings and standard embeddings are used, and a rich set of neural pooling functions is explored. Additionally, sentiment lexicons are used as an additional source of information for feature extraction.

Another notable work is by D. Tang et al. [132], who proposed two extensive LSTM

models: TD-LSTM (Target-Dependent LSTM) and TC-LSTM (Target-Connection LSTM). The first models the preceding and succeeding contexts surrounding the target string using two neural networks, one left (LSTML) and one right (LSTMR). The second further extends TD-LSTM by incorporating a target connection component, which explicitly uses the connections between the target word and each context word during the composition of a sentence representation.

Subsequently, the complexity of the interaction between the aspect and the context led to the widespread adoption of attention mechanisms. These allow models to focus on the parts of the sentence most relevant to the given aspect. Among the most representative works is the ATAELSTM (Attention-based LSTM with Aspect Embedding) model developed by Y.Wang et al. [153], which adds the aspect embedding vector to each word input vector, allowing the model to better model the interdependence between words and the input aspect. Another example is the IAN (Interactive Attention Network) by D-Ma et al. [87], which uses two attention networks to jointly model both the target and the context. With word embeddings as input, the IAN employs LSTM networks to obtain hidden states of words at the word level for a target and its context, respectively. With this design, the target and context can interactively influence the generation of their representations, which are then concatenated as the final representation sent to a softmax function for sentiment classification.

In recent years, pre-trained language models like *BERT* have become fundamental. For instance, Sun et al. [129] introduced an innovative method to adapt *BERT* to this task, transforming it into a sentence pair classification task by constructing auxiliary sentences, allowing *BERT* to fully exploit its comprehension capabilities. Dai et al. [42] demonstrated that *RoBERTa*, an optimized version of *BERT*, can extract relevant syntactic information for this task without the use of external parsers, highlighting the central role of structural relationships between key words in improving the task.

Parallely, a line of research has emerged that explicitly integrates syntactic information

into the sentiment classification process. For example, models based on Graph Neural Networks (GNN), such as those proposed by Sun et al. [130] and Zhang et al. [178], leverage dependency trees to capture syntactic relationships between the aspect and associated words, highlighting the influence of these dependencies on sentiment. This approach complements more contextual solutions, such as those by Wu and Ong [163], who propose context-guided variants of *BERT* (*CG-BERT* and *QACG-BERT*), improving task accuracy through advanced attention mechanisms.

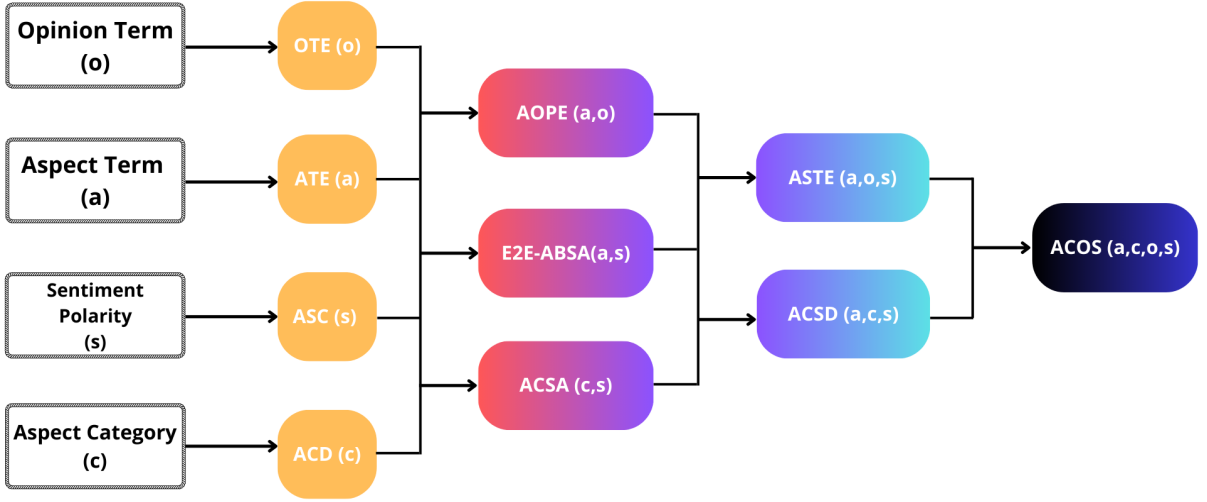


Figure 3.1: The relations of single and multiple ABSA tasks

3.3. Multiple ABSA tasks

Unlike individual ABSA tasks, multiple ABSA tasks focus on simultaneously addressing multiple interconnected elements of ABSA. These tasks aim to leverage the dependencies and relationships between different components to provide a more comprehensive and contextualized analysis. These tasks do not limit themselves to the isolated extraction of different sentiment components but seek to establish connections between them, organizing them into formats such as pairs (two elements), triplets (three elements), or

quadruples (four elements). The challenge of these tasks lies not only in identifying sentiment elements but also in modeling the interconnected dependencies between aspects, sentiments, and other components. Providing a unified solution to these interrelations represents a promising research direction, allowing for more effectively addressing the intrinsic complexity of sentiment analysis. In this section, we will delve into the methodologies and challenges associated with multiple ABSA tasks, placing particular emphasis on them. This choice is motivated by the fact that the subsequent chapters will explore the development and implementation of two specific tasks belonging to this category. These tasks highlight the potential of advanced approaches to integrate multiple elements into a unified framework, demonstrating how these solutions can revolutionize sentiment analysis in complex contexts and real-world applications.

3.3.1. Aspect-Opinion Pairs Extraction

This task focuses on the simultaneous identification of pairs composed of aspects and opinions. In the previous section, a very similar task (AOCE) was discussed, which can often be confusing. The difference lies in the output itself: AOCE focuses on the simultaneous but independent extraction of terms related to aspects and opinions present in a text, while AOPE shifts the attention to the identification of aspect-opinion pairs, explicitly establishing a connection between the two elements, which is why AOPE is considered a multiple task.

This distinction is crucial: while AOCE maps aspects and opinions without necessarily associating them, AOPE requires that each opinion be linked to the relevant aspect, highlighting semantic and contextual relationships that are fundamental for a deeper understanding.

Many works initially addressed the task using multiple sub-tasks. For example, Gao et al. [53] address the limitations of AOCE by introducing a new model: Question-Driven Span Labeling (QDSL) for aspect-opinion pair extraction. This model divides the

task into the extraction of aspect terms (ATE) using a span-based tagging scheme and then the extraction of opinion words corresponding to a given aspect (TOWE), using a question-answering (MRC) approach for better interpretability.

To avoid error propagation, which can affect these pipeline-based methods, unified approaches have been developed. Among these, Wu et al. [164] proposed an innovative tagging scheme, the Grid Tagging Scheme (GTS), labeling the relationships between all word pairs in a sentence. For example, in the sentence “*The pizza was delicious, but the service was slow*”, GTS creates a grid where each cell represents the relationship between a pair of words in the sentence, indicating whether the pair belongs to the same aspect, the same opinion, or forms an aspect-opinion pair. For instance, the relationship between “pizza” and “delicious” would be labeled as “P”, indicating they form an aspect-opinion pair. After completing the tagging and training the model, it predicts the GTS labels, and a decoding algorithm analyzes the predicted label grid to extract the aspect-opinion pairs.

Zhao et al. [187] proposed SpanMlt, a similar approach that relies on spans instead of tags. This method first identifies all possible word sequences (spans) in the sentence and then classifies each span as an aspect, opinion, or neither, and then evaluates if there is a relationship between span pairs, forming an aspect-opinion pair. This method is based on a multi-task framework, meaning it trains a model to perform both term classification and relationship identification.

S. Wu et al. [162] developed a model that integrates syntactic knowledge to help identify aspect and opinion words and how they are related. First, the model analyzes the sentence and identifies POS tags using a local attention mechanism that focuses on nearby words to determine the correct POS tag. The model also considers the sentence structure, such as dependency relationships between words and labels. This information is processed by a Graph Convolution Network (GCN), and finally, all this syntactic information is combined to create a representation of the sentence. After encoding the

syntactic information, the model seeks to identify word pairs that represent an aspect and its opinion using a higher-order scoring that considers relationships between more than two words at a time and a syntactically aware scoring that uses the syntactic information encoded in the previous sentence to evaluate the likelihood that two words are related as aspect and opinion.

3.3.2. End-to-End ABSA (E2E-ABSA)

The objective of this task is to identify aspect terms and their corresponding sentiments. Here too, the approach can involve dividing the task into sub-tasks, first applying an ATE (Aspect Term Extraction) and then an ASC (Aspect Sentiment Classification). Therefore, an intuitive and traditional approach is to perform the tasks sequentially, in a pipeline, where aspect terms are identified first, followed by the assignment of polarity. However, evidence has been developed indicating that aspect identification and polarity can mutually enhance each other. Inspired by this observation, several methods have been proposed to address the problem in an end-to-end manner. These methods are distinguished as joint and unified.

The first, Joint, tackles both sub-tasks (ATE and ASC) simultaneously, training the model to predict both the aspect boundaries and the sentiment polarity in the same step. The results of both tasks are then combined to generate the final prediction. This approach has been proposed in various works. For example, in Relation-Aware Collaborative Learning for Unified Aspect-Based Sentiment Analysis [38], the authors used a multi-task learning approach called RACL (Relation-Aware Collaborative Learning) in which aspect term extraction, opinion term extraction, and aspect-level sentiment classification work in a coordinated manner. RACL propagates interactive relationships between the sub-tasks through *relation propagation* mechanisms in a multi-layer network and uses *shared-private* schemes to extract both shared and task-specific features. H. Luo et al. [85] developed GRACE (GRAdient hArmonized and CascadED labeling model), a model

that addresses joint extraction, considering it as two *sequence labeling* tasks. This model uses a *cascaded* approach where aspect labels are used to enhance polarity labeling in a unified framework. GRACE employs two encoder modules with shared lower layers: one for aspect term extraction (ATE) and the other for aspect-level sentiment classification (ASC), with the latter enhanced by ATE labels. Additionally, it uses a “gradient harmonized” loss function to address label imbalance during training.

Y. Liang et al. [78] with IMKTN (Iterative Multi-Knowledge Transfer Network) aimed to leverage a full interaction mechanism, both among different sub-tasks and in transferring document-level knowledge. The model uses a routing algorithm to transfer task-specific knowledge between sub-tasks (each task receives input from the other tasks) and transfers fine-grained document-level knowledge to the sub-tasks. For example, if the document belongs to a specific domain, this is transferred to the ATE and OTE tasks, or the general sentiment of the document is passed to the aspect sentiment task. This is done iteratively, so in each iteration, the information from each sub-task is updated and exchanged to improve the overall understanding of the text.

The second, the unified approach, combines the two sub-tasks (ATE and ASC) into a single activity, using a unified labeling scheme. In this approach, each token in the sentence receives a tag representing both the aspect boundary and the sentiment polarity. This approach has proven effective due to its simplicity and ability to combine both labels into a single prediction. The use of this scheme has been explored in various works. X. Li et al. [74] used two stacked recurrent neural networks: one for the main task, which they prefer to call Target-Based Sentiment Analysis (TBSA), and the other for target boundary prediction, which serves as an auxiliary task to enhance the other network. The framework includes three main components: Boundary Guidance (BG) to guide the main network, Sentiment Consistency (SC) to maintain sentiment consistency within multi-word targets, and Opinion-Enhanced (OE) to improve the quality of boundary information. Some of the same authors, in [75], used *BERT* to tackle this task, surpassing

the results of other state-of-the-art models. The paper demonstrates how *BERT*, even with a linear classification layer, can capture complex semantic dependencies in the text. Among the different approaches, there is no clear difference in performance, so we cannot say which one is better.

3.3.3. Aspect Category Sentiment Analysis (ACSA)

Aspect Category Sentiment Analysis (ACSA) aims to jointly identify relevant categories within a text, based on a predefined set, and determine their associated sentiment polarities. Unlike Aspect-Based Sentiment Analysis (ABSA), which focuses on aspect terms explicitly mentioned in the text, ACSA operates at a more abstract level, concentrating on the semantic category that such terms represent rather than the specific term itself.

Thanks to this abstraction, ACSA is one of the most adopted tasks in the business context [25], where understanding opinions on different aspects of products or services is crucial for decision-making and marketing strategies. For this reason, ACSA will be the subject of particular focus in this thesis. Specifically, Chapter 5 will describe the development of an ACSA system designed for practical use within enterprises.



As with other tasks, the most straightforward method for this task seems to be the *pipeline* approach, i.e., first identifying the categories present in the text (ACD) and then predicting the sentiment polarity for each identified category (ASC). However, we have already noted that this approach can lead to error propagation and a lack of consideration of the relationships between the two phases, which are fundamental in some cases. An alternative solution is to integrate these two subtasks (ACD and ASC) into a multi-task learning framework, leveraging the interdependencies between the two tasks to improve

the performance of both. For example, Y. Ma et al. [89] developed Sentic LSTM, a model for tackling this task that integrates common sense affective knowledge (from *SenticNet*² and *AffectiveSpace*³). This model is based on the idea that external knowledge can enhance sentiment understanding, adding an additional layer of information. Sentic LSTM uses a hierarchical attention architecture, with target-specific and sentence-level attention, to identify the most relevant words and the relationships between targets and sentiment.

M. Hu et al. in CAN: Constrained Attention Networks for Multi-Aspect Sentiment Analysis [64] focused on attention regularization to improve accuracy in a multi-task environment, using an LSTM and constrained attention mechanisms. CAN aims to solve the problem of “noise” in sentences, which can distract attention from key words useful for determining sentiment towards a specific category. The authors introduced orthogonal regularization, allowing the model to focus on different parts of the sentence for different categories, and sparse regularization, which pushes the attention of each category to focus on only a few words.

Another work that deserves attention for this approach is by Z. Dai et al. [43], which introduces a new framework (CNE-net) for incremental learning, allowing the model to adapt to new categories without needing to be completely retrained from scratch. This model uses the same components to analyze all categories, instead of having different components for each, using “category name embedding”. Thus, the model receives the input sentence and, in addition, the category name, allowing it to learn specific representations for each category, even if it shares the same components. It also includes a shared *BERT* encoder that processes both the sentence and category names, and shared decoders, which aim to predict the sentiment.

²SenticNet: a knowledge base containing a significant number of concepts associated with affective properties. These properties provide not only a concept-level representation but also semantic links with aspects and sentiment.

³AffectiveSpace: an embedding space that maps SenticNet concepts into continuous low-dimensional vector representations, preserving the semantic and affective relationships of the original space.

The third approach, as already seen for other tasks, is to tackle the task by combining the two phases into a single model. Several works are representative of this approach, albeit with entirely different techniques. H. Wan et al. in Target-Aspect-Sentiment Joint Detection for Aspect-Based Sentiment Analysis [147] named their model TAS-BERT, based on *BERT* and designed to capture the dependency between aspects and categories for more accurate sentiment prediction. A key aspect of the work is the ability to handle implicit aspects, i.e., those not explicitly mentioned in the text, reducing the problem to a combination of text classification and sequence labeling problems, using a “yes/no” output to indicate the presence of a target and a sequence of tags for aspect extraction.

M. Schmitt et al. [119] use a CNN to perform this end-to-end approach. In this article, the *N/A* (not applicable) label is added in addition to positive, negative, or neutral, to indicate that a certain category is not present in the text. Subsequently, *fastText*⁴ is used as word embeddings, highlighting the advantage of using this for morphologically rich languages like German.

Cai et al. [27] reformulated the problem by creating a model called Hierarchical Graph Convolutional Network (Hier-GCN) for hierarchical category-sentiment prediction, with a two-level output structure. The lower level models the relationships between categories, while the upper level captures the relationships between categories and their sentiment. The idea is to consider that categories tend to co-occur and that sentiment depends on the identified category. For feature extraction, the authors use *BERT*, which allows generating contextual word representations, which are then used as input for the Hier-GCN model.

Instead of using classification models that add a specific output layer above pre-trained representations, J. Liu et al. [81] tackled the problem with a natural language generation *Seq2Seq* task, using the pre-trained generative model *BART* (Bidirectional and Auto-

⁴fastText: FastText is a lightweight library designed to help build scalable solutions for text representation and classification. It works on standard, generic hardware and can even fit on smartphones and small computers through functionality that reduces memory consumed by fastText models. [11]

Regressive Transformers) [71]. In this context, the input is a sentence, and the output is a sentence indicating the aspect category and associated sentiment. To adopt this approach, templates are used to indicate the format of the sentence to be generated, e.g., “*The sentiment polarity of <category> is <polarity>*”.

These modern approaches, particularly the unified ones, have achieved state-of-the-art performance in this task.

3.3.4. Aspect Sentiment Triplet Extraction (ASTE)

After exploring the fundamental tasks related to pair extraction, such as Aspect-Opinion and Aspect-Sentiment (E2E-ABSA), Aspect Sentiment Triplet Extraction (ASTE) emerges as a more complex challenge and expands these objectives. Its focus is the simultaneous identification of Aspect, Opinion, and Sentiment from the text, which highlights its value and complexity.

Below are some representative works addressing this task, using different paradigms. Peng et al. [104] introduced the task by adopting a two-phase pipeline approach. In the first phase, they identify aspects and sentiments through two sequence labelings: the first labels aspects and their sentiments with a unified tagging system (e.g., B-POS indicates the beginning of a positive aspect) using Bidirectional Long Short-Term Memory (BLSTM) neural networks, and the second labels opinion terms with a simple BIO tagging system using a GCN for dependency relations. In the second phase, the extracted aspects and opinions are matched using a binary classifier that evaluates the validity of each pair and constructs the final triplets. To address common issues in these approaches, unified tagging schemes have been developed. For instance, L. Xu et al. [169] implemented JET, a model based on a position-aware tagging scheme, including standard BIOES tags, sentiment polarity, and offsets, i.e., the distances between the beginning of the aspect and the ends of the opinion span⁵. The model’s output is a sequence of tags, where each

⁵opinion span: the part of the sentence that expresses an opinion or sentiment towards the aspect.

word is labeled according to this position-aware scheme. Wu et al. [164] also followed this approach with the Grid Tagging Scheme (GTS), which uses a grid to represent the relationships between all word pairs in a sentence.

Since these methods rely on the interaction between word pairs and may encounter issues when these pairs consist of multiple words, more advanced models have been developed that consider interactions between entire spans of aspects and opinions, such as the work by L. Xu et al. [170]. To manage the computational complexity arising from analyzing all possible spans, the model introduces a dual-channel pruning strategy, which is based on the supervision of target and opinion extraction tasks.

Among the *Seq2Seq* paradigms, there is the work by W. Zhang et al. [184], which proposes a generative approach that reformulates it as a text generation problem using *T5* as a pre-trained generative model. This approach is based on two main paradigms for triplet prediction: annotation-style modeling, where the input text is enriched with annotations that include the triplets, and extraction-style modeling, where the desired triplets are generated directly as output.

3.3.5. Aspect-Category-Sentiment Detection (ACSD)

Aspect-Category-Sentiment Detection (ACSD) focuses on the components of category, aspect, and sentiment polarity, which synthesize the semantic content and sentiment expressed in the text. This task arises from datasets constructed for the *SemEval-2015 Task 12* [108] and *SemEval-2016 Task 5* [109] competitions, where they are formalized as aspect-category-sentiment from a sentence. In these datasets, as mentioned in H. Wan et al. [147], sentiment polarity is detected based on both aspects and categories jointly. Therefore, considering only sentiment prediction from aspects or categories alone may lead to erroneous predictions. Specifically, when aspects are implicit, meaning they do not appear in the given text, methods that predict sentiment polarity based on aspects fail.

For this reason, this work (which also covers the ACSA task) proposes a new method for joint aspect-category sentiment detection using a model called TAS-BERT, which decomposes the joint detection problem into two sub-tasks, one for each category-sentiment pair. First, a binary sequential classification (*SeqClass*) is performed to determine if there is at least one aspect in the text for a given category-sentiment pair, including implicit targets. If an aspect is present, the associated terms are extracted as a conditional token classification task (*TokenClass*), thus forming the triplets.

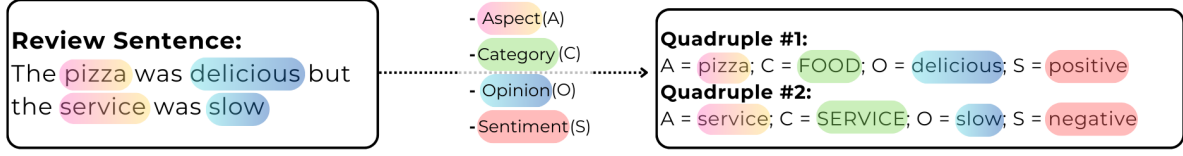
Similarly, Wu et al. [160] proposed MEJD using a GCN with an attention mechanism to capture contextual dependencies between the category and the text. By inputting a sentence and a specific category into the model, they decompose the problem by first predicting the polarity for the given category and then extracting the associated aspects. A key aspect of this work, as previously mentioned, is that the authors add the “N/A” label in the *SeqClass* sub-task, which allows identifying cases where no triplets exist for a given category.

Another approach adopted in the literature, as observed in the previous section in [184] by W. Zhang et al., is *Seq2Seq*-based models, where the task is transformed into a sequence generation problem, applied to various tasks including this one.

3.3.6. Aspect-Category Opinion Sentiment (ACOS)

This task represents the most complete and complex task in the field of Aspect-Based Sentiment Analysis, which aims to predict, in one go, all the elements described in section 3.1.1., forming quadruples given a text. Therefore, returning to our example, we will obtain two quadruples, shown in the following Figure.

As with other tasks, there are multiple approaches. Cai et al. [29] focused on the problem of implicit aspects and opinions by developing and adapting 4 pipeline approaches built for pair and triple tasks to the prediction of quadruples. Zhang et al. [183] instead,



with an end-to-end approach, introduced a strategy based on Paraphrase Modeling. The main idea is to transform the ACOS task into a paraphrase generation problem using a *Seq2Seq* model that takes the original text as input and outputs a textual representation of the quadruples in a predefined format, expanding the content into a sentence written in natural language. Tree-based generation approaches or structured schemes have also been developed to capture intricate relationships between elements. For example, Y. Mao et al. [91], with the *Seq2Path* model, instead of generating tuples as linear sequences, generate them as independent paths of a tree, using a discriminative token to select valid paths during inference. Similarly, X. Bao et al. [19] implemented a model that linearizes the opinion tree into a sequence of tokens and uses a constrained decoding algorithm to ensure the validity of the generated structure. Additionally, it employs a pre-trained model to integrate syntactic and semantic features, further improving output quality. In a recent work by H. Zao et al. [180], *ChatGPT* was also evaluated in this task. The results show that *T5* appears more performant and suitable for this task compared to *ChatGPT*, thanks to fine-tuning on labeled data for quadruple extraction.

3.4. Datasets and Metrics

Behind the development of all these tasks and approaches are the datasets used to implement them. Over the years, several datasets have been created and used for the development of new methodologies in all types of tasks analyzed in this thesis. This

section, presents the most popular ones divided by type.

3.4.1. SemEval Datasets

The *SemEval*⁶ (Semantic Evaluation) competitions have been a primary source of datasets in the field of ABSA. These datasets are widely used in research due to their comprehensive annotations and consistent evaluation benchmarks. For example, in *SemEval-2014 Task 4* [8, 107], four sub-tasks were presented: aspect term extraction, aspect term polarity, aspect category detection, and aspect category polarity on user reviews of restaurants and laptops, annotated by the organizers themselves using BRAT [128], a web-based annotation tool. For each sentence, aspect terms, their polarities, categories (only for restaurants), and category polarities (only for restaurants) were annotated in XML format.

Listing 3.1: Example of sentences annotated in the Restaurant dataset

```
<sentence id="813">
  <text>All the appetizers and salads were fabulous, the steak was mouth watering
    and the pasta was delicious!!!</text>
  <aspectTerms>
    <aspectTerm term="appetizers" polarity="positive" from="8" to="18"/>
    <aspectTerm term="salads" polarity="positive" from="23" to="29"/>
    <aspectTerm term="steak" polarity="positive" from="49" to="54"/>
    <aspectTerm term="pasta" polarity="positive" from="82" to="87"/>
  </aspectTerms>
  <aspectCategories>
    <aspectCategory category="food" polarity="positive"/>
  </aspectCategories>
</sentence>
```

Similarly, in *SemEval-2015 Task 12* [9, 108], the annotations of the restaurant and

⁶SemEval: a series of international research workshops on natural language processing whose primary mission is to advance the current state of the art in semantic analysis and to help create high-quality annotated datasets for an increasingly challenging range of problems. [7]

laptop datasets contain entire reviews, not isolated sentences. Here, when aspects are not explicitly mentioned, “NULL” is inserted, while categories are handled in Entity and Attribute ($E\#A$) pairs. For example, in the Restaurant dataset, there are 6 entities (FOOD, DRINKS, SERVICE, AMBIENCE, LOCATION, RESTAURANT) that can pair with 5 different attributes (GENERAL, PRICES, QUALITY, STYLE&OPTIONS, and MISCELLANEOUS).

In SemEval-2016 Task 5 [10, 109], additional domains (Hotels, Mobile Phones, Digital Cameras) and other languages of the existing datasets (Dutch, French, Spanish, Chinese, etc.) are added. In addition to sentence-level annotated datasets, text-level annotated datasets⁷ are provided. At text-level, given a customer review about a target entity, the goal is to identify a set of $\{category, polarity\}$ pairs which summarize the opinions expressed in the review.

Listing 3.2: Example of Restaurant review annotated at text-level

```
<Review rid="1090587">
<sentences>
<sentence id="1090587:0">
<text>Just went here for my girlfriend 23rd bday.</text>
</sentence>
<sentence id="1090587:1">
<text>If you've ever been along the river in Weehawken you have an idea of the top of
    view the chart house has to offer.</text>
</sentence>
<sentence id="1090587:2">
<text>Add to that great service and great food at a reasonable price and you have
    yourself the beginning of a great evening.</text>
</sentence>
<sentence id="1090587:3">
<text>The lava cake dessert was incredible and I recommend it.</text>
```

⁷text-level: The annotations are included all together at the end of each review, not divided sentence by sentence.

```

</sentence>
</sentences>
<Opinions>
<Opinion_category="LOCATION#GENERAL" polarity="positive"/>
<Opinion_category="SERVICE#GENERAL" polarity="positive"/>
<Opinion_category="FOOD#QUALITY" polarity="positive"/>
<Opinion_category="FOOD#PRICES" polarity="positive"/>
<Opinion_category="RESTAURANT#GENERAL" polarity="positive"/>
</Opinions>
</Review>

```

As we can see, these popular benchmarks lack opinion annotations. These were developed in [49] to perform the TOWE task for all three years. Subsequently, thanks to this, the *SemEval* data were reorganized by Xu et al. [168] into triplets. With this, a dataset containing annotations of aspect terms, opinion terms, and sentiment for the development and execution of ASTE tasks was finally obtained.

3.4.2. ACOS Datasets

Recently, datasets for the more advanced task that we have simply called ACOS in this thesis have been introduced, with which the four key elements of ABSA are annotated: aspects, categories, polarities, and opinions. There are several types of these, although they cover the same domains. Cai et al. [6, 28] created two new datasets, Restaurant-ACOS and Laptop-ACOS, for this task [181]. The first is derived from an expansion of the *SemEval 2016 Restaurant* datasets, while the second was collected from Amazon (2017-2018). The first comprises 2,286 sentences and 3,661 quadruples, while the second (Laptop) is larger, with 4,076 sentences and 5,773 quadruples. The Laptop-ACOS dataset has a higher percentage of implicit opinions and a higher density of quadruples per sample compared to the restaurant dataset. Both datasets include a mix of explicit and implicit aspects and opinions, as well as negative, neutral, and positive sentiments. Zhang et al.

[1, 182] developed the REST15 and REST16 datasets for the ACOS task in the restaurant domain. These datasets are based on *SemEval* datasets, enriched with category and opinion term annotations provided by Peng et al. [105] and Wan et al. [147]. Annotations for recurring aspects in a sentence were merged and consolidated, while new annotations were added for sentences without explicit aspects. Here, quadruples that included implicit opinions were excluded, and cases where an aspect was associated with multiple categories or opinions were manually reviewed to ensure correctness and consistency. The REST15 dataset comprises 1,080 sentences and 2,496 quadruples, while REST16 contains 2,124 sentences and 3,295 quadruples. Subsequently, Zhou et al. [188] introduced en-Phone, a dataset dedicated to the mobile phone domain and collected from e-commerce platforms, and zh-FoodBeverage, the first Chinese dataset for this task, focused on the food and beverage sector. Another ACOS dataset is the MEMD dataset [30], which comprises approximately 20,000 sentences and nearly 30,000 quadruples with implicit aspects and opinions, distributed across five different domains: books, clothing, hotels, restaurants, and laptops.

3.4.3. Other Datasets

In addition to traditional tasks and quadruples, other datasets have been designed to address specific issues. For example, there is the MAMS (Multi-Aspect Multi-Sentiment) dataset [5, 66], where each sentence contains at least two aspects with different polarities, making the task more complex compared to traditional datasets. This was developed for both ABSA and ACSA. Starting from *SemEval-2014*, Xing et al. developed ARTS (Aspect Robustness Test Set) [2, 165] to verify the robustness of models in aspect identification. Finally, there is the ASAP dataset [24], a large Chinese dataset of restaurant reviews, annotated with sentiment polarities for 18 categories, also containing ratings, useful for studying the relationship between coarse-grained and fine-grained sentiment analysis.

3.4.4. Evaluation Metrics

To evaluate the performance of models for all the tasks described, exact match evaluation is commonly adopted: a prediction is considered correct if all predicted elements exactly match the human annotations. Therefore, regardless of the task and the developed model, the most common metrics to evaluate these models are *Precision*, *Recall*, and *F1-score*.

Precision measures the proportion of correct positive predictions out of all positive predictions made by the model. In other words, the model correctly predicts the elements of ABSA, depending on the task:

$$\text{Precision} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Positives (FP)}}$$

High *Precision* implies that the model produces few false positives, i.e., incorrect predictions of non-existent aspects, polarities, categories, or opinions.

Recall, also known as sensitivity, measures the model's ability to identify all positive instances and thus evaluates the ability to correctly identify all existing elements of the task:

$$\text{Recall} = \frac{\text{True Positives (TP)}}{\text{True Positives (TP)} + \text{False Negatives (FN)}}$$

A model with low *Recall* might miss some elements present in the text, failing to identify the tuples.

F1-Score is the harmonic mean of Precision and Recall:

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

This metric is the most considered because it is particularly useful in the field of ABSA,

where a balance between complete identification of elements and reduction of prediction errors is necessary.

4 | Applied ACOS Benchmark

In this chapter, we fully report the work carried out in “Evaluating the Aspect-Category-Opinion-Sentiment analysis task on a custom dataset” [111], presented and published in the proceedings of the Seventh Workshop on Natural Language for Artificial Intelligence (NL4AI 2023) [20]. In this work, a practical benchmark of the tools available in the literature for the ACOS task is performed on a custom dataset provided by a local company, which cannot be published.

4.1. Dataset and annotation tool

The dataset used in this work is based on user reviews about skincare and pharmaceutical product supplied by a manufacturing company. The reviews have been scraped from e-commerce sites and are annotated using an open-source tool named *LabelStudio* [3].

LabelStudio is particularly useful for unstructured data annotation projects, such as text, images, and videos, and allows for flexible definition of the desired annotation type. In fact, it provides specific templates for different types of annotations but also allows for customization of the configuration, as done in the example we describe.

The annotation template used in LabelStudio is as follows:

```
<View>
<Labels name="lbl" toName="txt">
<Label value="aspect" background="#FF0000"/>
```

```

<Label value="opinion" background="#00FF00"/>
</Labels>

<Text name="txt" value="$sentences"/>

<Header value="Choose Sentiment"/>
<Text name="Sentiment" value="$textlabel"/>
<Choices name="sentiment" toName="text" choice="single" showInLine="true">
<Choice value="Positive"/>
<Choice value="Negative"/>
<Choice value="Neutral"/>
</Choices>

<Header value="Choose Category"/>
<Text name="text" value="$textlabel"/>
<Choices name="category" toName="text" choice="single" showInLine="true">
<Choice value="CATEGORY1"/>
<Choice value="CATEGORY2"/>
</Choices>
</View>

```

The template configuration was structured as follows:

- **Aspect and Opinion:** Labels for aspects and opinions were defined using the `Labels` component, which allows highlighting aspect and opinion terms within the text. Users annotate these terms with distinctive colors (red for aspect, green for opinion).
- **Sentiment:** The sentiment polarity associated with each review was labeled using a `Choices` component, where annotators must choose between *Positive*, *Negative*,

Figure 4.1: Example of annotation template in LabelStudio with hidden categories

and *Neutral* options.

- **Category:** Reviews were also classified into aspect categories by selecting an option from a predefined list of categories such as *CATEGORY1*, *CATEGORY2*, and so on. This allows categorizing each sentence based on the type of aspect mentioned.

With this configuration, it was possible to annotate all four elements necessary for the ACOS task: aspect term, opinion term, sentiment polarity, and aspect category. Each review was treated as a separate instance, with all elements annotated through a simplified and well-structured process.

These annotations show variations compared to datasets available in the literature due to the incorporation of numerous implicit aspects related to the provided product and opinion terms, often consisting of multiple words. The dataset (Table 4.1) includes 756 sentences and 1038 annotations, with the possibility that each sentence has multiple annotations (only one is shown in Figure 4.1).

	Train	Test	Total
Sentences	623	133	756
Annotations	881	157	1038

Table 4.1: Number of sentences and annotations in the training and test datasets

The annotations appears to be composed in a balanced way with regards to sentiment polarity (p); neutral sentiment is not calculated because predicting neutrality is not of interest in this case. As regards the categories, 13 classes were identified, encompassing both general and specific aspects of product performance.

The distribution of classes is mostly balanced, with the exception of the category pertaining to “general satisfaction of the final consumer” which happens to be the most frequent one.

The tools used for this benchmark are those available with the code. Since each tool requires a specific format to be used, modules in *Python* were developed to convert the JSON encoding of the dataset exported from Label Studio into other formats, each tailored to the specific tool.

4.2. Experimental Evalutation

In this section, there are the details of the experimental evaluation performed on the dataset built for the ACOS task. There have been selected three tools that have stemmed from significant studies in this field and for which the source code is publicly available online. All the selected tools leverage the fine-tuning of pre-trained models, specifically *T5* and *BERT*, as a crucial component of their functionality.

4.2.1. Paraphrase modeling

This model developed by W. Zhang et al. [183], already mentioned in section 3.3.6., proposes a new modeling paradigm called **PARAPHRASE**. Instead of treating the task as a series of separate classification or extraction problems, this model reformulates it as a paraphrase generation problem. In practice, this means that given an input sentence, the model generates a new sentence that paraphrases the original while simultaneously highlighting the sentiment elements.

This approach is based on the linearization of sentiment elements into a natural language sentence through projection functions. For example, a sentiment quadruple like “food quality”(c), “pasta”(a), “overcooked”(o), “negative”(s) is transformed into a sentence like “The food quality is bad because the pasta is overcooked”. This sentence, along with the original input sentence, is used to train the model.

In this way, the paradigm not only allows solving the task with an end-to-end approach, reducing the problem of error propagation that could occur in a pipeline approach, but also fully exploits the semantics of the sentiment elements, learning to generate them in natural language form.

The model used is based on a standard encoder-decoder architecture, specifically using *T5* [113] as pretrained. The encoder transforms the input sentence into a contextualized vector representation, while the decoder generates the paraphrase containing the linearized sentiment elements. The model is trained by maximizing the log-likelihood of the target sequences given the input sequences. Among the models considered, this is the only one that does not support implicit opinion terms.

4.2.2. Extract Classify-ACOS

The **Extract-Classify-ACOS** model is one of the four systems presented by Cai et al.[28] and is distinguished by its two-phase architecture, which allows for the separate but coordinated extraction of the fundamental components of an ACOS quadruple.

The first phase of the tool focuses on identifying aspect-opinion pairs within the text of a review. The process begins with a transformation of the input sentence, where two special tokens, [CLS], are inserted at the beginning and end of the sequence, which is important for allowing the model to understand the global context of the sentence. The transformed sentence is provided as input to *BERT*, which generates contextual representations for each token in the sentence, enabling the model to capture the semantic and syntactic

nuances of each word, essential for identifying aspects and opinions.

Subsequently, a Conditional Random Field (CRF)¹ layer is used to label the words in the sentence with a modified BIO (Begin-Inside-Outside) scheme. This scheme consists of five tags: *BA* (beginning of aspect), *IA*, *BO* (beginning of opinion), *IO* (inside of opinion), and *O* (other). The use of CRF allows for modeling dependencies between tags, improving extraction accuracy.

In addition to extracting explicit aspects and opinions, the model includes two binary classification tasks on the [CLS] tokens. These tasks aim to determine whether implicit aspects or opinions are present in the sentence (yes or no), and when the model identifies them, it labels them with the value “NULL”.

The second phase focuses on Category-Sentiment classification and quadruple formation. After identifying aspects and opinions (both explicit and implicit), the system creates a Cartesian product between the sets of aspects (S_A) and opinions (S_O), thus generating all possible candidate aspect-opinion pairs, as follows:

$$S_{AO} = \{a_1 - o_1, \dots, a_{|A|} - o_{|O|}\}$$

For each candidate aspect-opinion pair, the system performs a multi-label and multi-class classification process. Specifically, for each category, the average word vectors of the aspect and opinion are concatenated. The resulting vector is then fed into a fully connected layer with a *softmax* activation function. This layer determines the sentiment (positive, negative, neutral, or invalid) given the pair $a - o$ and c , or indicates an invalid quadruple.

¹Conditional Random Field: a class of statistical modeling methods used for structured prediction, where the goal is to predict a set of variables that depend on each other and on other observed variables. CRFs combine the ability of graphical models to compactly represent multivariate data with the ability of classification methods to make predictions using large sets of input features [131]

4.2.3. PyABSA

This tool, PyABSA [171, 172], is a variation of the original one, made for aspect-opinions pairs. There is no documentation about quadruple extraction because is a new feature. The data format of this tool was taken as a reference to transform the data once exported from the annotation tool (*Label Studio*). Also in this case *T5-base* is used as the pre-trained model. Fundamentally, the tool is designed to break down the ACOS task into a series of sub-tasks that are addressed modularly. This approach relies on specific instructions for each of the fundamental tasks.

The tool begins the process by preparing the dataset for the ATE task. Each sentence is transformed into input through a specific instruction that guides the model in recognizing the aspects present in the text. After extracting the aspects, the tool focuses on classifying the polarity associated with each aspect. Another instruction is used to construct the input by combining the original text with the extracted aspects. The goal is to assign a polarity to each identified aspect, creating labels in this way: *aspect:polarity*. Another crucial step involves identifying and associating opinions with the extracted aspects. Here too, using a dedicated instruction, the model detects the expressed opinions and links them to the corresponding aspects, represented as *aspect:opinion*, ensuring a clear relationship between aspect terms and associated opinions. Finally, the tool classifies the extracted aspects into a predefined category. This instruction uses the context of the sentence and the extracted aspects to assign a relevant category (e.g., “SERVICE” or “FOOD QUALITY”). Similarly to the others, the categories are represented as *aspect:category*, and the tool then combines the extracted information into final quadruples.

After testing these tools on this task with this dataset, further experiments were conducted using PyABSA. Specifically, (i) the tool was used with a larger pre-trained model, *T5-large*, which comprises 770 million parameters; (ii) a similarity threshold was applied between the true labels and those predicted by the model for one of the components: the

opinion term (o); (iii) the performance of PyABSA was evaluated using the *T5-large* model with the standard correctness criterion, without similarity, on the tuples of ACSA, E2E ABSA, ACSD, and ASTE, always performing the ACOS task, thus considering fewer tuples in the downstream metric calculations.

The second experiment is motivated by the fact that sentences in the domain of interest often contain implicit opinions, frequently composed of multiple words rather than single terms. Therefore, a relaxed correctness criterion was established to consider a prediction correct when it matches the reference standard in terms of aspect, category, and polarity, and when the similarity between the predicted opinion term and the real one is at least 70%. To calculate string similarity, the Python function *SequenceMatcher*² was used, based on an extension of the Ratcliff and Obershelp algorithm (“gestalt pattern matching”) [115], which compares pairs of sequences by finding the longest common subsequence while excluding uninteresting elements, with quadratic time complexity in the worst case.

In this way, for example, the prediction of the opinion “super practical to put in my bag” can be considered correct even if the real opinion is “practical to put in my bag”.

In Table 4.2, we show the tool settings we used for the experiments, including the batch-size, which indicates the number of training examples used in each iteration, the learning rate, a parameter in controlling the step size at each iteration while moving towards the minimum of a loss function and the number of epochs, representing the complete cycles through the training dataset.

Tool	batch-size	learning rate	epochs
Paraphrase modeling	16	3e-4	20
Extract Classify-ACOS	32{ <i>a</i> , <i>o</i> }, 16(<i>p</i>), 8(<i>c</i>)	2e-5{ <i>a</i> , <i>o</i> }, 3e5(<i>p</i>), (<i>c</i>)	20
PyABSA	16	5e-5	20

Table 4.2: Tool settings used for training

²<https://docs.python.org/3/library/difflib.html>

4.3. Results

To measure the performance of the models on the data, the most commonly used metrics to evaluate these types of tasks were computed, as described in the section, *precision*, *recall*, and *F1-Score*. The results are shown in Table 4.3. It should be noted that, with the exception of the last tool in the table where the similarity metric discussed above was used, the prediction of a quadruple is considered to be correct if and only if it is equal to the gold one in all its four components.

Tool	Precision	Recall	F1-score
Paraphrase modeling (T5-base)	0.373	0.382	0.377
Extract Classify-ACOS (BERT)	0.384	0.205	0.268
PyABSA (T5-base)	0.323	0.310	0.316
PyABSA (T5-large)	0.414	0.409	0.411
PyABSA (T5-large with similarity)	0.538	0.526	0.528

Table 4.3: Results of the experiments on the ACOS task with different tools

Among the tools with base pre-trained models (*T5-base* and *BERT*), the **Paraphrase modeling** tools seem to be the overall best, but the support for the implicit opinion, lacking from this tool, could be important for some application domains. The **Extract Classify-ACOS** tool seems to be slightly better than **Paraphrase modeling** in terms of precision but has a significantly lower value for recall. The last tool considered, **PyABSA**, is not the best in terms of performance but it turned out to be very well designed, allowing customization for performing further experiments using a larger pre-trained model (*T5-Large*) and employing a similarity criterion for one of the components. By using the larger model, the precision increased from about 32% to 41% using the standard correctness criterion, and to 54% using the relaxed correctness criterion based on similarity.

The results of the experiments using **PyABSA** with the *T5-large* model on some less complex ABSA tasks are reported in Table 4.4.

From the obtained results, it is evident that the model used by **PyABSA** performs well in

Task	Predicted elements	Precision	Recall	F1-score
ACSA	c,p	0.820	0.807	0.813
E2E ABSA	a,p	0.763	0.754	0.758
ACSD	c,a,p	0.662	0.654	0.658
ASTE	a,p,o	0.477	0.473	0.475
ACOS	c,a,p,o	0.414	0.409	0.411

Table 4.4: Results obtained by using the PyABSA tool with the *T5-large* model on some simpler ABSA tasks

predicting tuples, both in Aspect Category Sentiment Analysis (ACSA) and End-to-End ABSA (E2E ABSA). Furthermore, the model demonstrates good performance in extracting triples for Aspect Category Sentiment Detection (ACSD). However, it performs less effectively than the ACOS model with opinion term similarity set at 70% in Aspect Sentiment Triplet Extraction (ASTE). This observation implies that, within the framework of this model and the provided dataset, the primary limitation appears to be in the accurate identification of opinion terms. These terms, as previously discussed and as one might intuitively expect, are frequently composed of multiple words, posing a significant challenge for the model to predict them with absolute precision.

5 | PyACSA: A Comprehensive Framework for ACSA

In this chapter, we fully report the work carried out in the paper “A Comprehensive Framework for Aspect-Category Sentiment Analysis” [112], presented and published in the proceedings of the Eighth Workshop on Natural Language for Artificial Intelligence (NL4AI 2024) [23]. In this work, we developed tool (PyACSA) for ACSA by adapting one of the tools previously described for the ACOS task. Additionally, a dataset for this task on the Beauty and Personal Care domain we annotated for future research, and a web app was developed where the released dataset can be annotated both manually and semi-automatically. Furthermore, the tool is evaluated on other datasets available in the literature which are described along this thesis, in comparison with one of the state-of-the-art works that follows the same approach.

5.1. Beauty Dataset

Aspect Category Sentiment Analysis represents the right compromise between the value added and the difficulty of the task, as it allows understanding the topics within a text with the right granularity. By using categories, it is possible to capture even indirectly mentioned topics, including implicit aspects and opinions. To develop this model, a specific dataset was annotated for the use of the tool. The dataset is based on reviews from e-commerce sites such as *Amazon* or *Sephora* on *Beauty and Personal Care* products. The annotations were made in the same format as task 5 of *SemEval 2016 subtask 2* [10],

i.e., at the text level, explained in chapter 3. The dataset consists of twenty categories and the polarities can be of four types: *Positive*, *Negative*, *Neutral* when a category is mentioned without any sentiment, and *Conflict* when the same category is expressed in a positive and a negative inside the same text, but neither of the two is dominant.

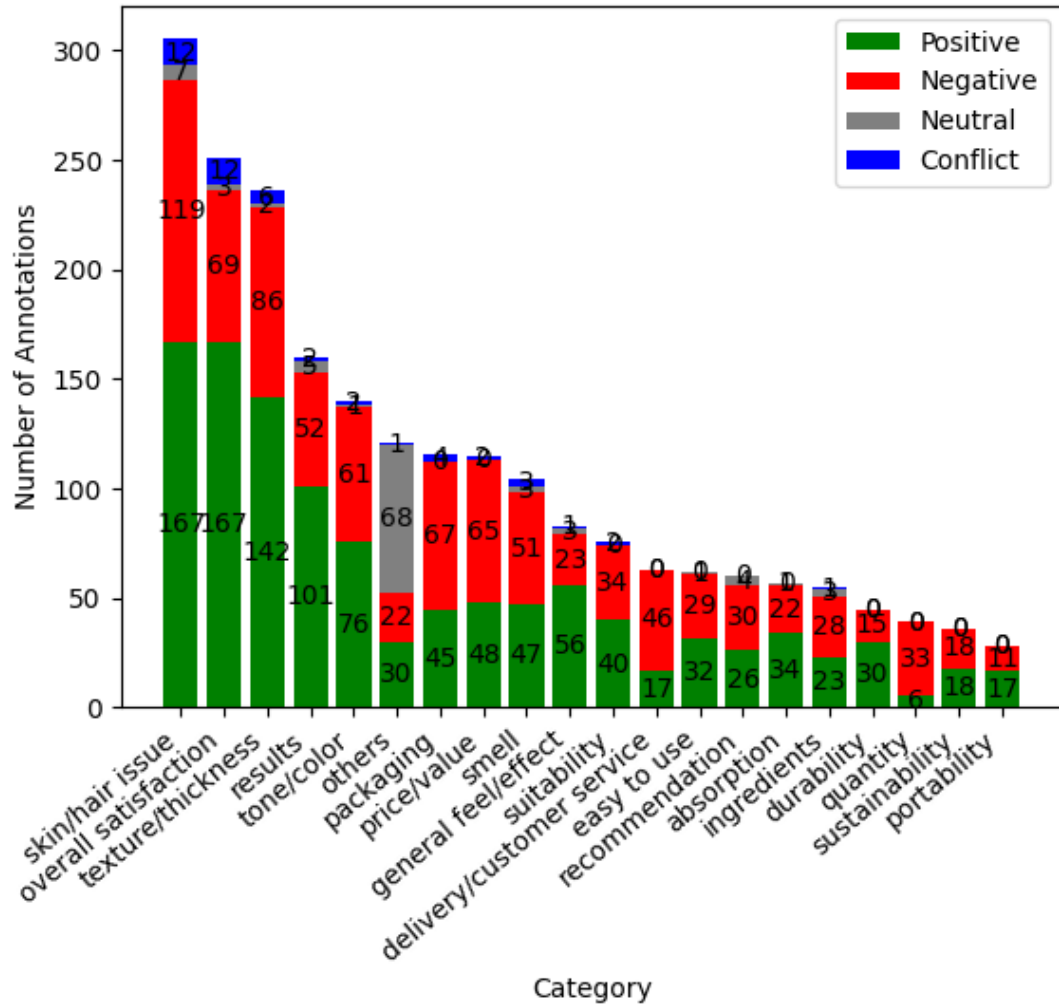


Figure 5.1: Annotations splitted by categories and polarities

In the following list there are the explanations of the single categories:

- **Skin/Hair Issue:** Any assessments of skin or hair issues This could include dryness, acne, redness, pores, or any other skin problem that the user attributes to the product;
- **General Feel/Effect:** Any assessments of general sensations that the user experi-

ences when using the product. This could include feelings of freshness, relaxation, tightness, or any other physical or emotional effect expressed in a general way.

- **Absorption:** Any assessments of the product's absorption into the skin or hair. This could include how quickly the product absorbs, whether it penetrates deeply into the skin or hair.
- **Texture/Thickness:** Any assessments about the consistency and the texture on the skin of the product. This could include whether the product is thick or thin, creamy or watery, smooth or grainy, etc.
- **Durability:** Any assessments about the product's durability upon application. This could include how long the product's effect lasts, whether it fades quickly, or whether it withstands sweat, water, etc. Not only, also if the mentions speak about the durability of the product (maybe because need only 1 small dose each time).
- **Suitability:** Annotate whether the product is compatible with certain skin or hair types. This could include whether the product is suitable for sensitive, oily, dry, combination, normal skin or for dry, oily, normal, curly, straight, colored hair, etc.
- **Tone/Color:** Any annotations related to skin tone and hair color. This could include whether the product makes light, medium, dark skin (also covering dark spot) or for blonde, brown, black, red, gray hair, etc.
- **Smell:** Any assessments of the product's smell. This could include whether the smell is pleasant, strong, light, natural, chemical, floral, fruity, spicy, etc.
- **Price/Value:** Any assessments about the cost of the product product or if the investment is worth it. This could include whether the price is high, medium, low, whether the product has good value for money, whether it's worth spending more for quality, etc.
- **Ingredients:** Any assessments related to the product's ingredients. This could

include whether the ingredients are natural, organic, chemical, hypoallergenic, not tested on animals, whether they contain parabens, sulfates, alcohol, essential oils, etc.

- **Packaging:** Any assessments related to the product's packaging. This could include whether the packaging is attractive, whether it's hard to open, whether the product leaks, whether it doses well or not, etc.
- **Sustainability:** Any assessments if the product is wasteful and any mentions about sustainability/recyclability. This could include whether the product is over-packaged, whether the packaging is recyclable, whether the product is biodegradable, and whether the company has sustainable practices. Referred both to the product and the packaging.
- **Easy to use:** Any assessments about the usability of the product. This could include whether the product is easy to apply, whether it takes a lot of time, whether it's easy to remove, easy to apply, etc.
- **Overall/Satisfaction:** Any assessment's overall thoughts about the product. This could include whether the user is generally satisfied with the product, whether they like the product, whether they would buy the product again, etc.
- **Portability:** Annotate whether the product is portable. This could include whether the product is compact, whether it's suitable for travel, etc.
- **Delivery/Customer Service:** Any assessments about delivery and customer service. This could include whether the delivery was fast, whether the product arrived in good condition, whether the customer service was helpful, whether the company has a good return policy, etc.
- **Recommendation:** Any assessments about whether consumers recommend the product or not. This could include whether the user would recommend the product to friends or family, whether the user has already recommended the product, etc.

- **Results:** Any assessments of whether the product works as expected. This could include whether the user noticed an improvement in their skin or hair, whether the product solved a problem (without specification), whether the user noticed improvement, etc.
- **Quantity:** Any assessments about the quantity and size of the product.
- **Others:** Any relevant information that does not fit into the previous categories. This could include mentions of promotions or discounts, comparisons with other products, shopping experiences, etc.

Listing 5.1: Example of the JSON dataset format for the PyACSA.

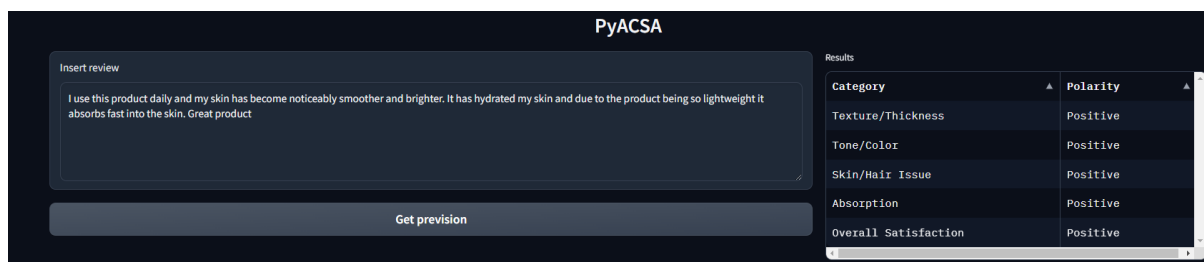
```
{
    "text": "I love how quickly this serum sinks into my skin",
    "labels": [
        {
            "polarity": "Positive",
            "category": "Absorption"
        }
    ]
}
```

5.2. PyACSA tool

As previously mentioned, to develop this tool, a tool cited in the previous chapter was adapted, implemented in PyTorch [102], fine-tuned on the dataset using several instructions, one for each element, for the more challenging ACOS task.

PyACSA uses the *Flan-T5* [156] with the *Seq2Seq* approach, in particular the *Flan-T5 XL*, pre-trained model described in the section 1.4 and using only one instruction to extract both categories and sentiment polarities. This method takes an input, categories,

and polarities, and returns a string that combines the instructions with the provided input. This formatted string is then fed to the model for training or prediction. A specific module facilitates the creation of a dataset for this task by preparing the data in required format, creating the training and the test datasets, and reading JSON data from a file. With this approach, implementing the ACSA framework is simplified by leveraging several utilities designed to facilitate the task. The tool’s integration with PyACSA provides a robust and flexible system tailored to specific needs, enabling the streamlining of data preparation and model training processes while exploring the application of a text-to-text model in a task where such models are not typically employed. The model is published at <https://huggingface.co/spaces/lorisdq/PyACSA>.



5.3. ACSA Utilities

This section presents the framework developed for the use and evaluation of the PyACSA tool. A web application in Python was implemented using the Flask package, with the aim of facilitating the use of this task, improving user interaction with the model, and making the entire process more accessible and efficient. The framework includes key components such as modules for data format transformation, which allow converting various input formats into the JSON format required by PyACSA, and tools for manual and semi-automatic data annotation. These features enable users to annotate data directly in the web application or to leverage the model’s predictions to assist the annotation process. This dual approach enhances flexibility and efficiency, adapting to the different needs and preferences of users, while the use of model-generated annotations

increases accuracy and speeds up the workflow. Additionally, the framework includes a module that generates a bar chart accompanied by a sentiment index (see Figure 3), which analyzes the categories of reviews entered into the system. This feature provides valuable insights into the distribution of sentiment across different aspects of the considered domain, supporting data analysis and interpretation. The developed code is available at <https://github.com/lorisdiquilio/ACSA-Framework-using-T2T-model>. The following paragraphs describe in detail the components of the framework, highlighting their functionalities and expected benefits.

5.3.1. Data Converter

The data conversion module offers several converters designed to facilitate the transformation of data between various formats, ensuring compatibility and ease of use for subsequent tasks. The module consists of several functions written in `Python` to transform data between tools, tasks, sub-tasks, and formats. Among the main converters developed are those for converting data from the XML format of *SemEval-14*, *SemEval-15*, and *SemEval-16* to the JSON format required by PyACSA, used to evaluate PyACSA on the *SemEval* datasets.

Data Converter

Select Conversion Type:

JSON to CSV(ACOS) ▼

Select Input File:

Scegli file Nessun file selezionato

Output Folder:

Enter output folder path

Convert

Please enter also the name of the file you are converting in the output folder

Function Descriptions - Data Converter Section

- JSON to CSV - ACOS Task:** Converts data from the ".jsonl" into a ".csv". If a sentence has multiple annotations, these are repeated for each of them. [See the input and output formats.](#)
- JSON to TXT - ACSD Task:** Converts data from the ".jsonl" to ".txt". It is used for the [Generative-ABSA tool](#) with the task (paradigm extraction). [See input and output formats.](#)
- JSON (Label-Studio) to SemEvalXML2014 - ABSA Task:** Converts a JSON file (from Label-Studio) to SemEval XML 2014 format. [See the input and output formats.](#)
- JSON (Label-Studio) to SemEvalXML2016 -Task 5 Subtask-1 - ACSA Task:** Converts a JSON file (from Label-Studio) to SemEval XML 2016 (Sub-task 1) format. [See the input and output formats.](#)
- JSON (Label-Studio) to SemEvalXML2016 -Task 5 Subtask-1 (opinion target version) - ACSA Task:** Converts a JSON file (from Label-Studio) to SemEval XML 2016 (Sub-task 1) format with the tag "opinion target". [See the input and output formats.](#)
- JSON (Label-Studio) to SemEvalXML2016 -Task 5 Subtask-2 - ACSA Task:** Converts a JSON file (from Label-Studio) to SemEval XML 2016 (Sub-task 2) format. [See the input and output formats.](#)

Figure 5.2: Data converter module

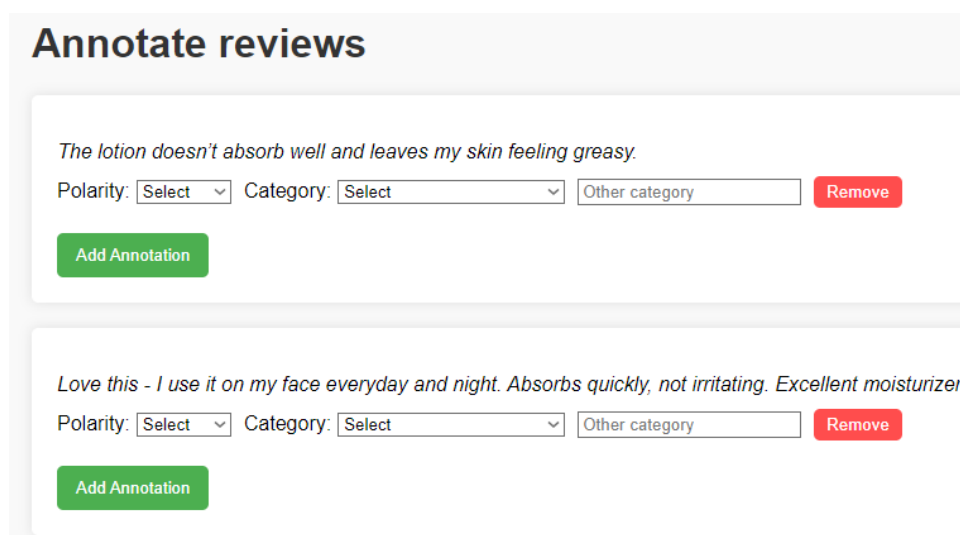
5.3.2. Data annotation

This module allows the user to annotate the dataset in the JSON format used by our tool (shown in Listing 1). Data can be annotated both manually and in a semi-automatic

way loading a file with a list of reviews.

Manual annotation

This module enables the uploading of a training file, either as an Excel file containing a list of unannotated reviews or a JSON file with pre-annotated text. Once the file is uploaded, each review can be annotated by adding new annotations through the selection of categories and polarities or by removing existing annotations, if needed. Additionally, the module allows for the addition of categories not yet recognized by the system (inserting the name in the tab “Other category”), ensuring flexibility and adaptability to various annotation requirements.



The screenshot displays the 'Annotate reviews' interface. It features a title 'Annotate reviews' at the top. Below the title, there are two review entries, each with a text area for the review, a 'Polarity' dropdown menu, a 'Category' dropdown menu, an 'Other category' text input, and a 'Remove' button. A green 'Add Annotation' button is located below each set of fields. The first review text is 'The lotion doesn't absorb well and leaves my skin feeling greasy.' and the second is 'Love this - I use it on my face everyday and night. Absorbs quickly, not irritating. Excellent moisturizer'.

Figure 5.3: Manual annotation module

When the annotation process is finished, it is possible to export the annotations in JSON format.

Semi-automatic annotation

This module is designed to leverage the initial trained model for further improvements. Once a trained model is available, through this module we can use the model itself to suggest annotations on new data, making the dataset creation process faster and more efficient. Similar to the manual annotation module, upon accessing this tab, the reviews

and model predictions will be displayed, with the initial prediction next to the text. Multiple predictions for the same sentence will appear below the text. After reviewing the model's predictions, necessary corrections can be made and saved back into the original training file (JSON).

This feature was used to annotate more data and improve model performance during experiments.

Semi-automatic Annotations

Reviews	Text	Polarity	Category
1	The lotion doesn't absorb well and leaves my skin feeling greasy.	Ne! ▼	Absorpti ▼ Other categ
Polarity: Negative ▼			
Category: Texture/Thickness ▼			

Figure 5.4: Semi-automatic annotation module. In this case the model predicts two tuples: {Absorption, Negative} and {Texture/Thickness, Negative}

In Figure 5.4, it is shown that the model predicts two correct tuples. This demonstrates how the annotation process becomes more efficient with the use of the model's predictions.

5.3.3. Report Generation

This module is designed to generate a report that provides a comprehensive assessment of the reviews for a product. A *sentiment index* is calculated to evaluate the overall quality of the product based on all its reviews within the relevant domain. In this context, the domain includes *Skin Care*, *Body Care*, and *Hair Care* products. The *sentiment index* is derived from the aggregated sentiment scores across various categories, offering a detailed

evaluation of the product's performance. The sentiment index is computed as follows:

$$\text{Sentiment Index} = \frac{\text{Positive} - \text{Negative}}{\text{Positive} + \text{Negative}} \quad (5.1)$$

Where *Positive* and *Negative* represent the number of positive and negative reviews, respectively. The sentiment index value ranges from -1 to 1.

Figure 5.5 illustrates the bar chart generated by the module, which displays the review polarities for each category of the selected product. This chart provides a visual representation of user perceptions of different aspects of the product, with each bar indicating the level of positive or negative sentiment associated with a specific category. Figure 5.6 presents both the *overall sentiment index* and the sentiment indices calculated for each category. The *overall sentiment index* offers a comprehensive view of the product's general reception, while the category-specific indices provide insights into particular aspects. This dual representation helps to understand not only the general acceptance of the product but also the specific areas where it performs well or encounters issues.

The results enable an evaluation of aspects that are perceived positively or negatively for the product. For example, the product is generally well-received due to its effectiveness, pleasant texture and smell, and convenient travel-sized packaging. However, users noted some drawbacks, such as the small quantity (for some), poor delivery and packaging, and the high cost of the product

The results allow for an evaluation of the categories that perform positively and negatively for the specific product. For example, it can be concluded that the product is generally well-received due to its effectiveness, pleasant texture and smell, and convenient travel-sized packaging. However, certain drawbacks are noted by users, including the small quantity (for some), poor delivery and packaging, and the high cost of the product.

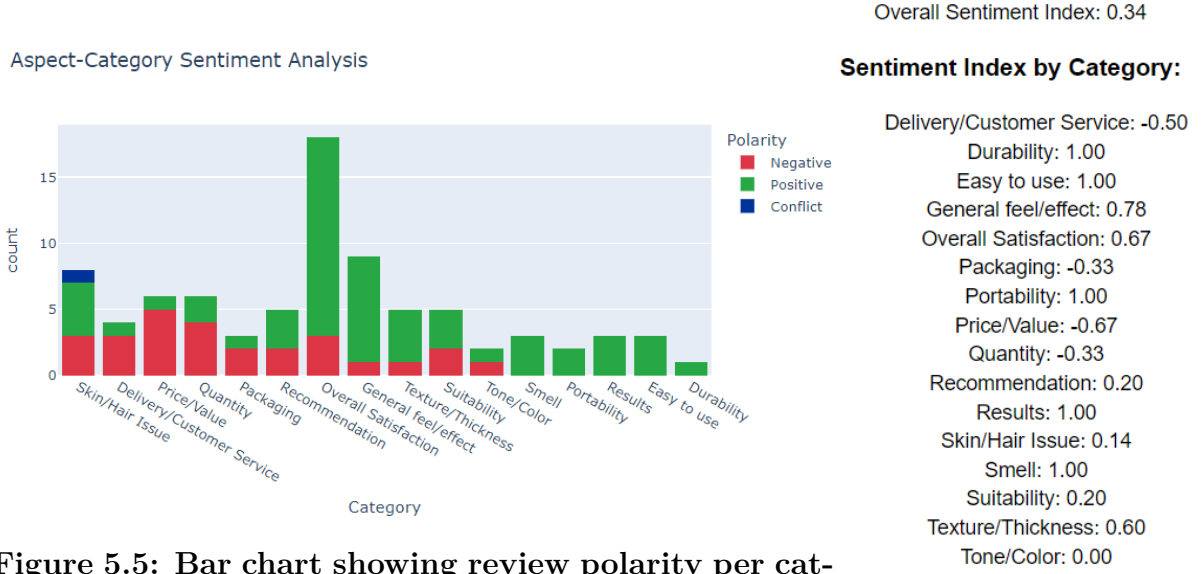


Figure 5.5: Bar chart showing review polarity per category

Figure 5.6: Sentiment Indexes

5.4. Experimental evaluation

The evaluations were conducted on the Beauty dataset shown in section 5.1 and in some publicly available datasets¹: SemEval-Laptop and SemEval-Restaurant in different versions, and MAMS (Multi-Aspect Multi-Sentiment). These datasets were chosen to ensure a comprehensive assessment of the framework’s performance across diverse domains and configurations.

The `Data converter` module of the framework was utilized to transform datasets from XML format into the JSON format compatible with PyACSA.

Table 5.1 provides detailed statistics about the datasets, including the number of training and test samples, the number of aspect categories, and the distribution of sentiment annotations (positive, negative, neutral, and conflict). These statistics highlight the variability and richness of the datasets, which were critical for evaluating the robustness and adaptability of the proposed approach.

¹<https://github.com/l294265421/ACSA/tree/master/datasets>

	Beauty	MAMS	Rest 14 Hard	Rest 141516 Large-Hard	Rest 16	Laptop
# Train	1932	3549	137	270	335	395
# Test	218	400	25	48	90	80
# Categories	20	8	5	8	12	87
# Positive annotations	1124	2415	146	274	1298	1548
# Negative annotations	877	2606	143	259	411	870
# Neutral annotations	104	3858	59	168	78	154
# Conflict annotations	48	-	-	-	52	55

Table 5.1: Datasets used for the experimental evaluation

The diversity in dataset sizes and sentiment distributions across categories ensures that the evaluation captures both domain-specific performance (e.g., Beauty) and generalizability across widely used benchmarks (e.g., MAMS, *SemEval*).

5.5. Experimental settings and results

In the experiment, the following settings are used for the PyACSA Tool: pre-trained `flan-t5-xl` with 3 billion parameters, learning rate of 5×10^{-5} . Epochs and batch size are set to 10 and 6, respectively. A regularization parameter (L_2 Regularization)² that helps prevent overfitting by reducing model weights during training is set to 0.01. The Warmup-Ratio (the ratio of total training steps used for a linear warmup from 0 to `learning_rate`) is set to 0.1.

The PyACSA Tool is compared with the already cited ACSA-Gen [81], one of the state-of-the-art tools in this field, which also leverages the power of a pre-trained generative model with a *Seq2Seq* approach. As previously mentioned, here a text sentence serves as input and the model generates as output a natural language sentence that describes the sentiment and the category.

The base model used is *BART-large-MNLI*³ with these configurations: learning rate of

²<https://paperswithcode.com/method/weight-decay>

³<https://huggingface.co/facebook/bart-large-mnli>

4×10^{-5} , 15 epochs, and batch size of 16. To facilitate the training and use of the model, the authors created templates for the ACSA and ACD tasks. To test the model in the experiment in question, the template “The sentiment polarity of `<given_category>` is `<polarity_type>`”. The generation process involves the encoder transforming the input sentence into a contextual representation, while the decoder uses this representation to generate the subsequent tokens in the output sequence. Each possible output is evaluated through a score based on the probabilities of the generated tokens. It is worth noting that the pre-trained model used in PyACSA is larger than the one used by ACSA-Gen.

The results reported in the table are based on the most common metrics: *Precision*, *Recall*, and *Micro-F1 Score*⁴.

Tool	Metrics	Datasets					
		Beauty	MAMS	Rest14 H	Rest141516 H	Rest16	Lap16
ACSA-Gen	P	0.8109	0.7347	0.78	0.7333	0.8272	0.6567
	R	0.7477	0.7469	0.7358	0.7264	0.6163	0.3229
	F1	0.7780	0.7407	0.7572	0.7298	0.7063	0.4329
PyACSA	P	0.8116	0.7741	0.8	0.8333	0.8316	0.6756
	R	0.8786	0.7913	0.7547	0.8018	0.8069	0.3211
	F1	0.8438	0.7826	0.7766	0.8173	0.8190	0.4353

Table 5.2: Results of the experimental evaluation on different datasets for ACSA task

The results show that PyACSA performs better than ACSA-Gen in all the considered datasets, likely due to the power of the pre-trained model, as the *BART* model has a smaller size (approximately 406 million parameters) compared to the *T5-XL* (3 billion). The larger size of *T5-XL* enables PyACSA to capture more intricate relationships between text and sentiment categories, leading to improved performance across various tasks.

However, it is important to note that while PyACSA excels in the ACSA task on most

⁴Micro F1-Score: computes a global average *F1-score* by summing the True Positives (TP), False Negatives (FN), and False Positives (FP) across all classes and then plugging these values into the F1 equation

datasets, its performance on the Laptop dataset is less impressive. This could be attributed to the complexity of the dataset, as it contains a larger number of categories, which may pose a challenge for models in terms of categorization accuracy and granularity. Despite this, the overall results demonstrate that **PyACSA** performs well in the ACSA task at the text level, except on the Laptop dataset, probably because of the high number of categories it contains.

6 | An Application of ACSA to Consumer Review Analysis to Inform Product Design

This chapter will delve into the potential applications of Aspect Category Sentiment Analysis (ACSA) within business contexts, with a particular focus on two strategic areas: Product Design and Sales Forecasting. It will explore the mechanisms through which ACSA can support innovation in decision-making processes, enhancing the understanding of consumer preferences and anticipating market trends. We report an article which will be submitted to a journal, and the code for the experiments of this study is available at https://github.com/lorisdiquilio/PredictiveModel_ACSA.

Consumer product reviews represent a rich and underutilized resource for manufacturers, offering insights into customer preferences and shedding light on which product features hold the most value for users. By systematically analyzing this data, companies can prioritize the development of products that better align with consumer expectations, ultimately improving their market responsiveness and competitiveness. Previous research has shown that user-generated content, such as reviews, is an influential factor in shaping consumer behavior and driving product innovation [15, 36].

Sales forecasting is a pillar of strategic planning and operational efficiency in numerous industries. Accurate sales predictions enable businesses to optimize inventory management, staffing, and financial planning. Traditional forecasting methods, including time

series models, regression analysis, and moving averages, have been employed extensively. However, advancements in machine learning (ML) have introduced more sophisticated approaches capable of capturing complex patterns in sales data, incorporating diverse variables and nonlinear interactions. Despite these advancements, the integration of qualitative insights such as consumer sentiment derived from reviews—into sales forecasting remains underexplored, creating an opportunity to bridge the gap between customer perception and market outcomes.

Historically, the collection and analysis of product requirements have been conducted through traditional methods such as the Kansei method [95], which involves gathering descriptive words from sources such as interviews, catalogs, and surveys, selecting them through manual or statistical techniques, and finally linking them to user evaluations. While these approaches have helped identify consumer preferences, they present significant limitations, including the labor-intensive nature of the work, the poor quality and quantity of data collected, and limited success rates in new product development. These methods, therefore, are insufficient in addressing the challenges posed by the vast amount of data generated by consumers in the digital age.

Previous studies on data-driven product design have primarily focused on examining the relationship between product characteristics and sales performance. These analyses, detailed in the following sections, provide a foundational understanding of how certain features influence purchasing decisions. Similarly, advances in natural language processing (NLP) and sentiment analysis have enabled researchers to identify key features appreciated by consumers through the examination of product reviews. However, as discussed before, these studies have largely stopped short of linking consumer perceptions directly to economic indicators such as sales or demand.

In this experiment, we propose an integrated approach to address this gap. By combining Aspect Category Sentiment Analysis (ACSA) with sales forecasting techniques, we aim to explore how consumer sentiment about specific product characteristics translates

into market performance. Sales forecasting is a pillar of strategic planning and operational efficiency in numerous industries. Accurate sales predictions enable businesses to optimize inventory management, staffing, and financial planning. Traditional forecasting methods, including time series models, regression analysis, and moving averages, have been employed extensively. However, advancements in machine learning (ML) have introduced more sophisticated approaches capable of capturing complex patterns in sales data, incorporating diverse variables and nonlinear interactions. Despite these advancements, the integration of qualitative insights—such as consumer sentiment derived from reviews—into sales forecasting remains underexplored, creating an opportunity to bridge the gap between customer perception and market outcomes. This approach not only highlights common effects of consumer preferences across different product categories but also identifies those that are unique to particular types of products, providing a more nuanced understanding of customer behavior.

This work demonstrates the potential of combining sentiment analysis with economic forecasting, offering a novel perspective on product design and its relationship to market dynamics. The results of this study serve as a practical guide for manufacturers seeking to leverage consumer information to inform strategic product development, particularly in the early phase of development: requirements analysis. In this analysis, the PyACSA model described in the previous chapter was used to predict topics in customer reviews.

6.1. Data-driven product design

Data-driven product design represents a paradigm that leverages the vast amount of data generated at various stages of a product's life cycle. This data includes, among others, information derived from market research, sales data, and usage data collected from IoT devices. However, for user requirement analysis, online consumer reviews represent a particularly valuable and underutilized resource. Platforms such as *Amazon*, *Booking.com*, and *TrustPilot* offer a wealth of information that can be used to better

understand user needs and preferences.

6.1.1. Product design based on consumer reviews

Online reviews present numerous advantages that make them ideal for user requirement analysis, including the high level of engagement from online buyers, who increasingly use reviews as a reference before making a purchase [37, 110, 124]. This implies that reviews reflect authentic and detailed opinions, often representative of a wide range of experiences, ensuring greater reliability and authenticity due to the direct link with the product through the “Verified Purchase” badge. They provide a broad spectrum of reviews, thus involving a large number of people, and most importantly, they offer real-time access, allowing companies to respond quickly to changing consumer preferences.

Based on the analysis by Feng et al. [52] and Quan et al. [110], data-driven product design, particularly using textual data, can be integrated into decision-making processes to improve product efficiency and quality. One of the main advantages of this approach is the ability to identify latent needs and user requirements, which are often not explicitly expressed through traditional methods, as well as the identification of product defects. Additionally, this approach allows companies to prioritize product features by identifying which characteristics are most relevant in reviews, both positively and negatively, and prioritizing those that enhance overall quality and customer satisfaction.

Another important aspect is the ability to monitor product perception over time and compare it with competing products. Online reviews provide continuous feedback on consumer perception, allowing companies to evaluate the effectiveness of updates and modifications made to products over time.

Finally, immediate access to data facilitates the implementation of a more responsive and iterative product development cycle, reducing the time needed for organizing design knowledge and data, leading to an efficient and dynamic development process capable of

adapting the product to market needs more quickly, from understanding to recommendation.

Therefore, data-driven product design is not just an improvement over traditional methods but a paradigm shift that transforms the entire product development process. By using authentic and reliable data sources such as online reviews, companies can design products that more directly meet the real needs of users, ensuring greater market competitiveness and higher customer satisfaction.

6.1.2. Effect of product review on sales

Building on the analysis of product characteristics and their perceived importance, it is essential to explore how product reviews (consumer perceptions) affect sales. Previous research has investigated a broad range of factors related to review properties and their impact on consumer purchasing behavior.

One critical aspect is the visibility of reviews. The likelihood that a review is seen by potential buyers strongly determines its influence on sales. For example, Alzate et al. [13] demonstrated that reviews deemed more useful have a greater impact on sales than those with lower visibility. This finding underscores the importance of prioritizing detailed and informative reviews in online platforms. Companies can amplify the positive effects of such reviews by improving their visibility, using algorithms that rank reviews by utility or recency.

The sentiment and content of reviews also play a crucial role in driving sales. Zhang and Wu [76] proposed the Joint Sentiment-Topic (JST) model to analyze how numerical ratings and textual sentiment collectively influence sales. Their research found that numerical ratings mediate the effect of textual sentiment, with both direct and indirect impacts on sales. Moreover, sentiment related to specific product aspects is often more predictive of sales than overall sentiment. For instance, positive sentiments about hedon-

nic experiences and hardware features contribute significantly to increased sales, whereas negative sentiments about logistics or interface issues have minimal impact.

Another influential factor is the richness of reviews, including the presence of multimedia elements such as videos or follow-up reviews. Jia and Lui [174] explored this dimension by introducing the concept of “attribute substitution”. Their findings suggest that richer reviews, particularly those with visual or follow-up content, result in higher sales compared to text-only reviews. Consumers perceive visually rich reviews as more reliable and persuasive, often using them to fill gaps in product information.

A study by Chen et al. [35] used eye-tracking to analyze how online reviews influence consumer purchase decisions. The results showed that consumers’ attention, especially that of women, focuses more on negative reviews. There is a correlation between visual attention and purchase intention: those who focus on negative reviews tend not to buy. These findings suggest that sellers should monitor negative reviews and personalize information, taking into account the gender of consumers.

These findings provide evidence that product reviews significantly influence sales by shaping consumer perceptions and decision-making processes. This underscores the strategic importance for businesses to optimize review visibility, encourage sentiment-rich feedback, and monitor consumer engagement with reviews to maximize their impact on sales.

6.2. Methodology

In our approach, we integrate the effects of both aspects to study the impact of variables through sales, while also observing an improvement in their prediction. The methodology presented in this chapter involves several key steps: collecting and preprocessing review data, detecting the most impactful reviews, and integrating and developing models for both sentiment analysis and sales forecasting, respectively. Through the application of

machine learning techniques, such as *Multilayer Perceptron (MLP)* models, and interpretability tools like *SHAP analysis*, we provide a comprehensive framework for linking consumer feedback to actionable business insights.

6.2.1. Data Collection

The dataset was created by selecting 78 products (ASINs) from *Amazon.com*'s *Beauty and Personal Care* category. Products were chosen based on a minimum threshold of 100 customer reviews, ensuring a robust dataset for analysis. *Amazon.com* was selected as the source platform due to its vast collection of English-language reviews and its prominent position in e-commerce.

To allow for more detailed analysis, the products were grouped into sub-categories based subdivision done by *Amazon*, such as *Body Treatment*, *Face Moisturizer*, *Hair Care*, *Hair Styling*, and others. Data collection spanned 206 days, from December 23, 2023, to July 16, 2024, using *SellerSprite*¹, a tool specialized in e-commerce market research. *SellerSprite* provided product-related data by tracking variables such as *price*, *average rating*, *number of ratings*, and *daily sales*. Additionally, for specific days, the *number of reviews* per product was recorded manually.

According to *SellerSprite*, sales data are derived by monitoring daily inventory changes, making them accurate unless disrupted by unusual factors such as changes in *Buy Box sellers*.

The *Buy Box* is a prominent section on *Amazon* that displays critical purchasing information such as price, shipping options, and seller details. When multiple sellers offer the same product, they compete to appear in the *Buy Box*, which is prominently displayed to customers. *Amazon* determines *Buy Box* eligibility based on factors such as pricing, product availability, and seller performance. If the seller displayed in the *Buy Box* changes from one day to the next, inventory tracking may be affected, leading to

¹<https://www.sellersprite.com/>

potential discrepancies in sales data comparisons across days.

For better understanding, an illustration of the *Buy Box* is provided in Figure 6.1.

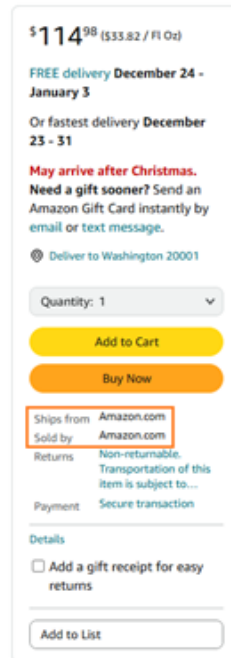


Figure 6.1: Buybox seller of an Amazon product

The dataset collected from SellerSprite consisted of the following columns. *NumReviews* is the only variables added manually:

- **Station:** Indicates the specific Amazon marketplace from which the data was collected, in this case, the US;
- **ASIN:** The Amazon Standard Identification Number, a unique identifier for each product on Amazon;
- **Date:** Represents the date of the specific information;
- **Product Name:** The name of the product as it appears on Amazon;
- **CategoryName:** The specific macro-category to which the product belongs, in this case, Beauty & Personal Care;

- **Sub-Category:** The sub-category of the product, such as Face Cleanser or Body Treatment, classified based on the specific effect of the product by *Amazon*;
- **Product link:** The direct link to the product's page on Amazon;
- **Price:** The price of the product on the specific date;
- **Avg Rating:** The average rating of the product based on customer reviews, on a scale from 1 to 5;
- **NumRatings:** The total number of ratings the product has received;
- **NumReviews (manually added):** The total number of textual reviews left by customers for the product;
- **Units sold:** The number of units of the product sold on the specific date.

6.2.2. Data preprocessing

The collected dataset required preprocessing to handle missing values and ensure consistency for subsequent analysis. Our primary target variable was daily sales (*Units Sold*), but missing data were observed across multiple variables.

Among the columns, *NumReviews*, which was manually added, exhibited the highest proportion of missing values, with approximately 86% of data missing. Other columns showed significantly lower percentages of missing values: *Price* had 2% missing values, *Avg Rating*, *NumRatings*, and *Units Sold* with about 1%.

To address these missing values, we applied a combination of time-based interpolation for variables with a temporal dimension and multivariate imputation by chained equations (MICE) [141] for other cases. This approach ensured that the dataset was complete and consistent for downstream analysis.

Time Interpolation

Temporal interpolation is a well-known method used to fill in missing values within a time series of observations. This technique is particularly useful for variables that exhibit a natural progression over time. Temporal interpolation methods can include linear, nonlinear, and spline interpolation, among others, and are generally applied to ensure continuity and accuracy in data analysis. These methods are essential for imputing missing data in both regularly and irregularly spaced time series [133]. They rely on the time-dependence of the data, using available observations before and after the missing values to estimate the missing points. Specifically, in this experiment, this was done for the cumulative measures:

- *NumRatings*: Missing values for the cumulative number of ratings were interpolated linearly across time for each ASIN, assuming a steady accumulation trend;
- *NumReviews*: Similarly, missing values for the cumulative number of reviews were interpolated using linear methods, preserving temporal continuity.

Linear interpolation is a straightforward and widely used method for imputing missing data in time series. It assumes that changes between consecutive observations are gradual and evenly distributed. This approach is particularly suitable for cumulative measures like ratings or reviews, where each data point logically depends on prior values. By employing time-based interpolation, the dataset retains the chronological structure and ensures that variables with cumulative progression reflect realistic trends over time.

Multivariate Imputation by Chained Equations (MICE)

For variables where temporal relationships are less significant, such as *Price*, *Avg Rating* and *Units sold*, missing values were imputed using the Multivariate Imputation by Chained Equations [141] (MICE) method.

MICE, as outlined by van Buuren and Groothuis-Oudshoorn [141], is a statistically

robust approach that sequentially models each variable with missing values as a function of all other variables in the dataset. The method follows these steps:

1. Initialization: Missing values are initially filled with simple imputation techniques (e.g., mean or median imputation);
2. Sequential Modeling: Each variable with missing values is treated as a dependent variable and modeled using the remaining features as predictors;
3. Iteration: The process repeats iteratively, updating imputed values until convergence.

One key advantage of MICE is that it accounts for the uncertainty of missing values by generating multiple imputations. This makes it particularly for datasets with complex relationships among variables. This method was applied to variables *Price*, *Avg Rating*, and *Units sold*, ensuring that the imputed values reflect the relationships present in the data, rather than introducing biases or simplistic assumptions.

To evaluate the effectiveness of the imputation, a test was conducted on the dataset with missing values in these variables, removing 10% of the data at each iteration (10 iterations) and calculating RMSE (Root Mean Square Error) and MAE (Mean Absolute Error) [32].

The test results in the Table 6.2 shows that BayesianRidge² achieved the best performance overall, which is why it was chosen for imputing the missing data.

Incorporating temporal features

In this phase, additional temporal features were created to capture important time-related patterns that could influence sales trends. Specifically, dummy variables were generated based on the month, weekends, and U.S. holidays, despite the sales data being recorded in Chinese time.

²https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.BayesianRidge.html

- *Month*: a dummy variable for each month was created to account for seasonal variations in sales;
- *Weekend*: a dummy variable was added to indicate whether the data fell on a weekend, as consumer behavior can differ on weekends compared to weekdays;
- *US Holidays*: dummy variables were created for significant US holidays based on consumer behavior and sales patterns.

Holiday	Date
Christmas	2023-12-26
New Year's Day	2024-01-02
Martin Luther King Jr. Day	2024-01-16
Valentine's Day	2024-02-15
Presidents' Day	2024-02-20
St. Patrick's Day	2024-03-18
Easter	2024-04-01
Mother's Day	2024-05-13
Memorial Day	2024-05-28
Father's Day	2024-06-17
Independence Day	2024-07-05

Table 6.2: List of U.S. Holidays Considered for the Analysis

These temporal features were added to the dataset to improve the model's ability to capture the effects of time and consumer behavior on sales.

6.3. Top review detection

This section was developed to identify the 10 most relevant reviews for each product in the dataset on a daily basis, based on the assumption that these reviews have the greatest

influence on consumer decisions. On *Amazon's* main product page, only 10 reviews are displayed, and while the criteria for selecting these reviews are not disclosed by *Amazon*, it is reasonable to assume that these reviews are the ones most frequently viewed by potential buyers. As a result, these reviews are expected to significantly impact purchase decisions, more so than reviews on subsequent pages.

To capture this dynamic influence, a classification model was trained to predict, for each day, which reviews *Amazon* would select to display on the main product page. This model combines manually annotated data with machine learning techniques to track the most influential reviews as they evolve over time. The approach ensures that the reviews with the highest likelihood of affecting consumer behavior are consistently identified.

6.3.1. Data Preparation and Feature Engineering

To implement this task, the process began by annotating a subset of reviews for 42 products that were not included in the main dataset downloaded with the plug-in of *SellerSprite*³. Each review was manually labeled as “Yes” (indicating it was a top review) or “Not” (not among the top reviews). These annotations were designed to reflect the reviews likely displayed on the first page of each product.

Review
“I use this everyday before I step out of the shower. The smell is clean and fresh but not overpowering and fine for men as well as women. I need a stronger moisturizer (Lac Hydrin) in the colder months but I still use this first. I have gone through many bottles of the large pump size and hate to run out. Guests love it too. Put it in your shower you won't regret it!”

Figure 6.2: An example of a customer review in the dataset.

The annotated dataset was enriched with additional features to improve model accuracy:

³<https://chromewebstore.google.com/detail/sellersprite-amazon-resea/lnbmbgocenenhhdjdielgnmeffbnfb>

NumDays, *WordCount*, *AvgHelpfulness* and *HelpfulnessPerRating*.

Feature Name	Definition	Mean	Std. Deviation
NumRatings	Number of ratings received	19455.35	37351.58
NumReviews	Number of reviews published	1916.66	2751.07
Verified Purchase	Verified purchase status (Yes or Not)	-	-
Avg Rating	Average rating of the product	3.45	1.52
NumHelpful	Number of helpful votes	1.90	5.97
NumImages	Number of images in the review	0.13	0.52
NumVideos	Number of videos in the review	0.01	0.10
AvgHelpfulness	Helpfulness score per review count	0.006	0.023
HelpfulnessPerRating	Helpfulness score per rating count	0.001	0.005
NumDays	Days since the review was published	507.20	715.97
WordCount	Number of words in the review text	34.61	38.83
IsTopReviews	Review in the 1st page (Yes or Not)	-	-

Table 6.3: Summary of features including their definitions, means, and standard deviations (for numerical features) considering all variables.

These variables, combined with existing product and review-specific data such as *Verified Purchase*, *Price*, *Avg Rating*, *NumVideos*, and *NumImages* included, provided a robust input for model training.

6.3.2. Classification model

For the experiment, several machine learning methods were tested, including XGBoost, Random Forest, K-Nearest Neighbors, and Logistic Regression. Among these, XGBoost [34] demonstrated the best performance during cross-validation and was selected for this step. XGBoost is considered a highly effective machine learning algorithm, particularly suited for classification tasks due to its efficiency and scalability.

To address the imbalance in the annotated dataset, where the majority of instances were

labeled as “Not”, the Synthetic Minority Oversampling Technique (SMOTE) [33] was applied. SMOTE is a widely used method for mitigating class imbalance by generating synthetic samples for the minority class, ensuring better performance and fairness in classification tasks.

To optimize the model, Bayesian optimization was employed for hyperparameter tuning. This approach utilizes a probabilistic model, typically Gaussian Processes, to explore the hyperparameter space more efficiently than a traditional grid or random search identifying the best combination of hyperparameters in fewer iterations. Key parameters such as learning rate, maximum depth of trees, and the number of boosting rounds were systematically tuned to enhance the model’s performance. This method not only accelerates the optimization process but also reduces the computational cost compared to exhaustive search techniques.

The final trained *XGBoost* model classified reviews into “Yes” or “Not”, enabling the identification of the top 10 reviews displayed on the first page for each product on a daily basis. This dynamic method allowed for tracking changes in consumer-facing content over time, reflecting shifts in review rankings or the addition of new reviews.

The detailed performance of the model and its results will be discussed in the dedicated results section.

6.4. Aspect Category Sentiment Analysis model

Review analysis is a complex process that goes beyond identifying relevant reviews to understanding what consumers like or criticize about a product. To fully understand how reviews influence consumer decisions, it is essential to analyze not only which reviews have an impact, but also what they say about specific aspects of the product, but as highlighted at the beginning of this chapter, product reviews play a crucial role in influencing consumer purchase decisions, which, in turn, directly impact sales.

Aspect Category Sentiment Analysis (ACSA) addresses this need by enabling the extrac-

tion and analysis of consumer opinions on particular aspects of a product. In this study, the dataset and model developed in [112] discussed in Chapter 5 were leveraged to fill this gap and improve the understanding of consumer sentiment.

The core aspect of this section lies in the logic developed to implement the ACSA model. The model was applied to the top ten reviews for each day and each product, resulting in the analysis of over 160,000 lines of data. The processing logic was carefully designed to skip repeated reviews, ensuring that the analysis was not skewed by duplicate content. A critical step in this experiment involved creating code to transform the model predictions into actionable metrics for daily analysis. The predictions from the Aspect Category Sentiment Analysis (ACSA) model were converted into dummy variables to indicate the presence or absence of sentiment across specific categories on a daily basis for each product. The polarities *Neutral* and *Conflict* are considered with 0.

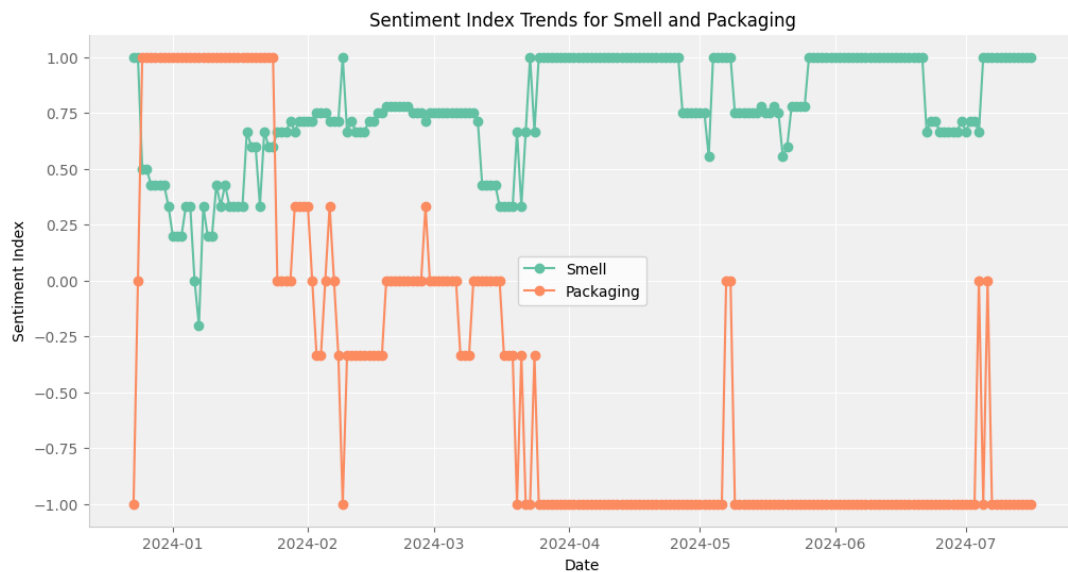


Figure 6.3: Sentiment Index over the time for “philosophy amazing grace firming body emulsion” product

These indices were then aggregated and matched to individual products, enabling us to track category-level sentiment trends over time. This approach offered valuable insights into how consumer perceptions evolved and provided detailed daily monitoring of product performance. Finally, the sentiment indices were integrated into the main dataset

alongside sales data, creating a comprehensive resource for building the predictive sales model.

- D : Set of days in the analysis period.
- P : Set of products in the dataset.
- $R_{d,p}$: Set of 10 reviews for product p on day d .
- C : Set of categories analyzed by the ACSA model.
- $S_{r,c}$: *Sentiment index* for category c in review r , where $S_{r,c} \in [-1, 1]$.

For each day $d \in D$ and each product $p \in P$:

1. Apply the ACSA model to each review $r \in R_{d,p}$ to compute the sentiment index $S_{r,c}$ for each category $c \in C$.
2. Compute the daily average *sentiment index* for each category c :

$$\bar{S}_{d,p,c} = \frac{1}{|R_{d,p}|} \sum_{r \in R_{d,p}} S_{r,c}$$

Since $|R_{d,p}| = 10$, this simplifies to:

$$\bar{S}_{d,p,c} = \frac{1}{10} \sum_{r \in R_{d,p}} S_{r,c}$$

Here, $\bar{S}_{d,p,c}$ represents the normalized daily sentiment intensity for category c , product p , and day d . This index, ranging from -1 to 1, captures the balance between positive and negative sentiments for each category. For example, an index of 0.75 for the category *Smell* would indicate predominantly positive sentiment for that aspect on a given day.

Next, the computed daily sentiment indices $\bar{S}_{d,p,c}$ for all categories $c \in C$ were included as additional variables in the main dataset. This enriched dataset, used to build the sales forecasting model, contains for each row:

- The daily average *sentiment index* $\bar{S}_{d,p,c}$ for each category $c \in C$, normalized and ready for analysis.

- Other relevant features for forecasting, such as historical sales data, pricing, promotions, and external factors.

By incorporating these indices, the dataset provides a comprehensive view of daily product performance, enabling the analysis of how sentiment for specific categories impacts sales trends. This dynamic and enriched dataset formed the foundation for building a robust and interpretable sales forecasting model.

6.5. Sales Forecasting with ACSA

After completing the dataset with all necessary data, the final and most critical phase of the experiment commenced: integrating the Aspect Category Sentiment Analysis (ACSA) into the time series forecasting model and examining its impact.

Product Name	Category
CHRISTIAN DIOR J'Adore Women Eau De Parfume Spray	Parfum
La Roche-Posay Anthelios AOX Daily Antioxidant Serum with SPF	Facial Sunscreens
La Roche-Posay Effaclar Purifying Foaming Gel Cleanser for Oily Skin	Face Cleanser
L'Oreal Men Expert Vita Lift Anti-Wrinkle Gel Moisturiser, 50 ml	Face Moisturizers

Table 6.4: Some products used in the dataset

The ultimate goal of this work is to identify the sentiment characteristics that have the greatest influence on consumers, to leverage these insights for product design. To achieve this, sentiment indices derived from ACSA, which capture consumer opinions on specific product categories over time, were integrated into the forecasting model. These indices were used alongside traditional variables, as described in Section 6.2.

Feature Name	Definition	Mean	Std. Deviation
Price	Price of the product	40.70	36.65
Avg Rating	Average customer rating of the product	4.49	0.19
NumRatings	Number of ratings received	6336.14	12355.25
NumReviews	Number of reviews published	937.13	1784.14
Units sold	Number of units sold	162.63	385.60
Skin/Hair Issue	Skin/Hair Issue sentiment index	0.75	0.42
Overall Satisfaction	Overall satisfaction sentiment index	0.93	0.20
Texture/Thickness	Texture/Thickness sentiment index	0.70	0.46
Tone/Color	Tone/Color sentiment index	0.53	0.55
Results	Results sentiment index	0.74	0.46
General feel/effect	General feel/effect sentiment index	0.68	0.53
Price/Value	Price/Value sentiment index	0.31	0.70
Packaging	Packaging sentiment index	0.00	0.71
Smell	Smell sentiment index	0.52	0.61
Recommendation	Recommendation sentiment index	0.53	0.56
Suitability	Suitability sentiment index	0.47	0.62
Absorption	Absorption sentiment index	0.33	0.53
Delivery/Customer Service	Delivery/Customer Service sentiment index	0.05	0.61
Easy to use	Ease of use sentiment index	0.19	0.57
Ingredients	Ingredients sentiment index	0.18	0.56
Durability	Durability sentiment index	0.42	0.60
Sustainability	Sustainability sentiment index	0.04	0.21
Quantity	Quantity sentiment index	-0.03	0.55
Portability	Portability sentiment index	0.07	0.26
January	Indicator for reviews in January	-	-
February	Indicator for reviews in February	-	-
March	Indicator for reviews in March	-	-
April	Indicator for reviews in April	-	-
May	Indicator for reviews in May	-	-
June	Indicator for reviews in June	-	-
July	Indicator for reviews in July	-	-
August	Indicator for reviews in August	-	-
September	Indicator for reviews in September	-	-
October	Indicator for reviews in October	-	-
November	Indicator for reviews in November	-	-
December	Indicator for reviews in December	-	-

Table 6.5: Summary of all features including definitions, means, and std. deviations (for numerical features).

For better visibility, category variables have not been included in the table. These category indicators are dummy variables similar to the month indicators, with values of 0 (No) and 1 (Yes). The categories include: *Body Sprays*, *Body Treatment*, *Eyes Cream*, *Face Cleanser*, *Face Luminizers*, *Face Mask*, *Face Moisturizer*, *Face Scrubs*, *Face Serum*, *Face Tanner*, *Facial Spray*, *Facial Sunscreens*, *Hair Oil*, *Hair Serum*, *Hair Styling*, and *Parfum*.

Although improving sales forecasting accuracy by adding sentiment variables to traditional predictors is not the primary objective, it serves as an essential step in evaluating the significance of sentiment in consumer behavior. By linking sales data with sentiment trends, the study aims to uncover actionable insights into which aspects of sentiment drive consumer decision-making, ultimately informing more effective product design strategies.

6.5.1. Data Transformation

In this phase, the stationarity of the sales data for each product was assessed using both the Augmented Dickey-Fuller (ADF) test [45] and the KPSS (Kwiatkowski-Phillips-Schmidt-Shin) test [69]. Since the data was found to be non-stationary, differencing techniques were applied to transform it into a stationary form. This step was essential for removing trends and seasonal effects, making the series more suitable for time series modeling.

To further refine the dataset, products with a mean absolute difference [47] in sales values equal to or less than five were excluded. This threshold was established to eliminate products with minimal sales fluctuations, ensuring that only those with sufficient variation were included in the forecasting model. The mean absolute difference was calculated using the formula:

$$\text{MAD} = \frac{1}{n} \sum_{i=1}^n |x_i - \bar{x}|$$

where x_i represents the individual sales values, $\bar{x}$ is the mean of these values, and n is

the number of observations.

Optimal lags for each product were determined by analyzing the autocorrelation patterns and using the Akaike Information Criterion (AIC) [12], the Bayesian Information Criterion (BIC) [121], and the Hannan-Quinn Information Criterion (HQIC) [56]. This analysis revealed that the optimal lag for each product was one, which was then incorporated as a feature in the model to capture temporal dependencies effectively.

Given that a single unified model was developed for all products, dummy variables were created to account for products. The dataset was then prepared for training by splitting it, with the last month designated as the test set to evaluate the model's performance.

Finally, the *Multi-Layer Perceptron (MLP)* model was trained using the prepared data. The model incorporated traditional features, such as lagged sales values, alongside sentiment-driven variables derived from the Aspect Category Sentiment Analysis (ACSA) model. The use of MLP, a versatile and powerful neural network model, allowed for the capture of complex nonlinear relationships within the data. Its ability to handle large datasets and interact effectively with both structured and sentiment-driven variables made it a robust choice for this task.

The integration of sentiment indices into the model aimed to explore the impact of consumer sentiment on sales, providing deeper insights into temporal dependencies and product-specific trends.

6.5.2. Sales forecasting model: Multilayer Perceptron (MLP)

Among the various machine learning techniques, Multilayer Perceptron (MLP) has gained prominence for its ability to model intricate relationships in data. MLP is a type of artificial neural network (ANN) designed for supervised learning tasks. It consists of an input layer, one or more hidden layers, and an output layer, with each layer comprising interconnected neurons. These neurons apply nonlinear activation functions, enabling the model to capture complex, non-linear patterns in the data.

MLPs can be used for both classification and regression tasks. In classification, MLPs are employed to categorize data into distinct classes based on learned patterns. For example, they can be used to classify images, detect spam emails, or predict customer churn. In regression, MLPs predict continuous outcomes, such as forecasting sales, estimating house prices, or predicting stock market trends.

Several studies have highlighted the effectiveness of Multi-Layer Perceptrons (MLPs) in sales forecasting. For instance, Devil H. et al. [176] explored the use of MLPs to forecast sales, comparing them with traditional time-series methods. The results indicated that MLPs provided more accurate sales forecasts, especially for 1 to 6-month horizons, outperforming methods like moving averages. Similarly, Thiesing and Vornberger [135] applied MLPs to forecast weekly sales in a German supermarket, aiming to overcome the limitations of traditional statistical methods in forecasting complex and non-linear time series. The MLP forecasts were compared with two traditional methods used by the supermarket: a “Naive” method (which simply forecasts the last available value) and a statistical moving average. The results showed that MLPs provided superior forecasts, especially for 75% of the products analyzed. More recently, Tiwari et al. [136] compared MLPs and Radial Basis Function Networks (RBFNs) for sales forecasting to determine which model is more effective. Their simulations showed that MLPs outperformed RBFNs, particularly in sales forecasting performance. Additionally, Teixeira et al. [134] studied how promotional data could improve retail sales forecasting using hierarchical models. They compared ARIMA models with MLPs and found that MLPs performed better for longer forecast horizons, proving to be particularly suitable for retail forecasting due to their simplicity and efficiency. Additionally, the study *Forecasting Sales at Fuel Stations Using a Multilayer Perceptron* [68] demonstrates the effectiveness of MLPs in predicting fuel sales, highlighting their superiority over traditional methods. The paper emphasizes the importance of reliable forecasts for managing and optimizing business processes at fuel stations. Accurate predictions help anticipate future trends and demand, enabling better planning, inventory management, and pricing strategies.

These studies highlight that one of the main advantages of MLPs in sales forecasting is their ability to learn complex interactions between features that traditional models often overlook. For example, MLPs can effectively capture the impact of promotions or the relationships between different product categories, offering a more nuanced understanding of sales dynamics. Furthermore, MLPs can be combined with feature importance techniques, such as SHAP values, to identify and analyze the most influential factors driving sales.

6.6. SHAP Analysis

To analyze feature importance, SHAP (Shapley Additive ExPlanations) values were computed (Lundberg and Lee [84]). SHAP is a state-of-the-art model-agnostic method for interpreting machine learning models, which quantifies the contribution of each feature to a specific prediction. This method is particularly useful for understanding the causal relationships between features and their impact on the model's predictions, even when the model itself is complex and nonlinear.

Unlike traditional Shapley values, which are based on an additive approach, SHAP provides local explanations by computing the marginal contribution of each feature for a specific instance. This allows SHAP to offer intuitiveness, consistency, and stability in its generated explanations [118, 144]. These properties make SHAP a valuable tool for interpreting models like Multi-Layer Perceptron (MLP), which are often considered “black-box” due to their complexity.

By applying SHAP to MLPs, we can gain insights into the relative importance of each feature in the prediction process, even when interpreting complex, nonlinear relationships. This helps to provide a clearer understanding of the model's behavior, offering interpretability in cases where simpler models may lack transparency.

Numerous studies have used SHAP to study feature behavior and biases in models with a large number of features extracted from textual data. Velampalli et al. [143] developed

several models for sentiment classification of tweets in the context of a targeted marketing campaign. Representing the text of tweets and emojis in terms of embeddings using pre-trained embedding models like USE and SBERT, the authors examined the SHAP values for these features to provide an overview of their relative contribution to the eventual sentiment polarity identified in a tweet. This application demonstrates how SHAP can be used to analyze complex features, such as embeddings, which capture semantic representations of textual data.

In this analysis, the features were ranked based on their mean SHAP values across the test set. Features at the top of the SHAP plots indicate a higher influence on the model's predictions, while those further down the list have comparatively less impact. A detailed visualization of SHAP values is provided to illustrate the importance of individual features, and these results are discussed in detail in the results section.

6.7. Results and discussion

In this section, all the results from the previous sections will be displayed and discussed, except for the results from section 6.5, as these results are available in section 5.6.

6.7.1. Top reviews detection results

As mentioned in the previous section, for this phase we got the best results from the *XG-Boost classifier* enhanced with SMOTE to address the class imbalance. The performance metrics of the classifier are summarized in the table below:

Metric	Value
Precision	0.86
Recall	0.89
F1-Score	0.88

Table 6.6: Performance Metrics of the Top Reviews Detection Model

The confusion matrix further demonstrates the effectiveness of the model:

	Predicted “No”	Predicted “Yes”
True “No”	1415	33
True “Yes”	22	94

Table 6.7: Confusion Matrix of the Top Reviews Detection model

The model achieved high accuracy with a notable balance between *Precision* and *Recall*, indicating that it is effective at identifying top reviews and minimizing false positives. The *F1-Score* of 0.88 highlights the classifier’s overall strong performance.

Analyzing the feature importance reveals the most influential variables of this model. We observed that the most significant predictor of a review being rated as ‘top’ is whether the review is a verified purchase (*Verified Purchase*). This is followed by the number of reviews (*NumReviews*) for that product, the product rating (*Avg Rating*), and the presence of media such as videos (*NumVideos*). The model suggests that verified purchases, review content, and product ratings are strong indicators of the likelihood that consumers will deem a review valuable.

Other important factors include the number of days (*NumDays*) since the review was posted, the total number of ratings (*NumRatings*) the product has received, and the ratio of helpful votes to the total number of reviews (*AvgHelpfulness*). Additionally, the length of the review text (*WordCount*) and the presence of images (*NumImages*) also play a role, although they are less significant compared to the other factors mentioned.

6.7.2. Sales Forecast with ACSA results

Building upon the methodology outlined in the previous chapter, the sales forecasting model was developed and evaluated. The models were assessed using only traditional variables and with the addition of sentiment variables, both for the overall model and for models created for each category. Below, the model’s results are presented. The MLP

achieved these results (RMSE) across different datasets reported in the following table. The objective of this experiment is to identify the most relevant variables for product design development. However, the *Sales Forecast* model was tested to determine if adding sentiment variables would improve the model, and it did. Starting with a model based solely on non-textual variables (RMSE: 119.22), variables derived from ACSA were added, resulting in an improved RMSE of 115.87. This suggests that adding sentiment variables enhances the model. The results are statistically significant, calculated using the combined Fisher [179](X^2) method. The same approach was applied by dividing the dataset for each category and focusing on the categories with the highest number of products. The results:

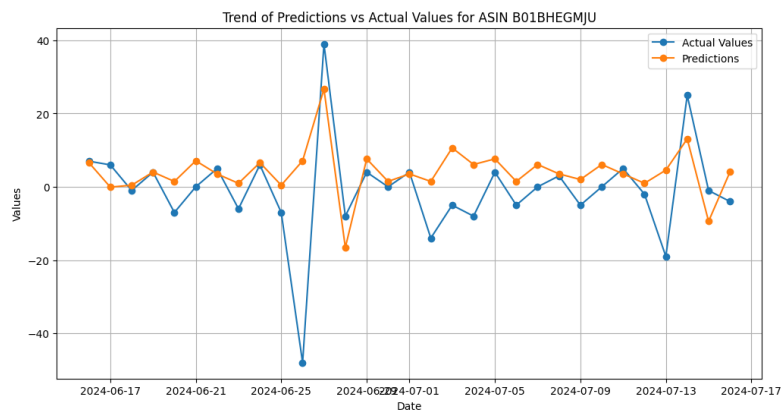
Model	Num Products	Non-Textual	Non-Textual + ACSA	X^2	p-value
Body Treatment	8	76.38	71.82**	30.43	0.0159
Face Cleanser	8	65.03	56.01***	61.08	3.44e-07
Face Moisturizer	14	255.15	227.66***	91.07	1.35e-08
Face Serum	19	144.34	119.81***	176.78	8.69e-20
All Categories	49	119.22	115.87***	462.85	2.55e-38

Table 6.8: Model Performance, Number of Products, and Fisher Test Results

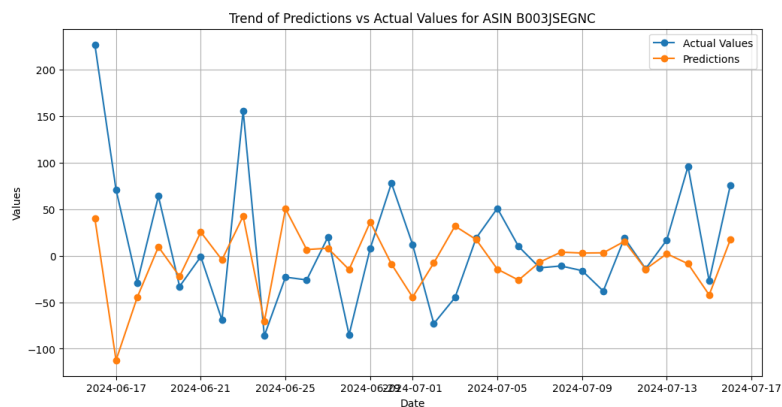
** indicates statistical significance at the 0.05 level;

*** indicates statistical significance at the 0.01 level.

Including sentiment-related variables significantly enhances predictive accuracy across all categories. The integration of sentiment information through the ACSA model leads to statistically significant reductions in RMSE for each product type. This demonstrates that sentiment data adds valuable context, improving the performance of models even when applied to single categories such as *Body Treatment*, *Face Cleanser*, *Face Moisturizer*, and *Face Serum*. The results highlight the robustness of this approach in refining predictions across different domains.



(a) Actual Values compared with the Sales Forecast Values of “Timeless Skin Care Matrixyl” product



(b) Actual Values compared with the Sales Forecast Values of “La Roche-Posay Effaclar Purifying Foaming Gel Cleanser for Oily Skin” product

Figure 6.4: Comparison of Actual Values with Sales Forecast Values for two products

6.7.3. SHAP Analysis results: Predictor impact and Conclusions

As previously discussed, the analysis utilized Shapley Additive Explanations (SHAP) to ensure a comprehensive understanding of the predictors influencing the model’s performance. These initial results pertain to the complete model, encompassing all product categories analyzed together. The SHAP analysis provides insights into how each variable contributes to the model’s predictions at both global and local levels. Positive SHAP values signify a positive contribution to predicted sales, whereas negative values indicate a

detrimental impact. Furthermore, the color gradient in the SHAP visualizations denotes feature values, where higher values (represented in red) correspond to more positive sentiment and lower values (depicted in blue) indicate less favorable sentiment. The findings are illustrated in the subsequent figure. It is important to note that in the graphs below, the visualization of the plots has been restricted within a range on the X-axis to achieve better clarity. However, the impact of the variables is in decreasing order, from top to bottom. In the following graphs, the SHAP beeswarm plots are used to illustrate the impact of variables on the predictive models.



Figure 6.5: SHAP Beeswarm plot for the entire model

This first plot provides an overall analysis of the model, highlighting the key variables across all products. The most influential factor is the expressed rating (*Avg Rating*), which strongly correlates with sales differences (*Units Sold Diff*) as well as whether the product is a *Face Serum* or *Face Cleanser* as we might have imagined. The SHAP value

distribution shows that higher ratings (indicated in red) are consistently associated with positive sales outcomes, underscoring the important role of consumer feedback in driving purchasing decisions.

Since the model considers all products collectively, without distinguishing between specific categories, it is evident that product categorization plays a significant role in shaping sales performance. Products in certain categories benefit from inherent consumer demand trends, making categorization a key driver of sales variation.

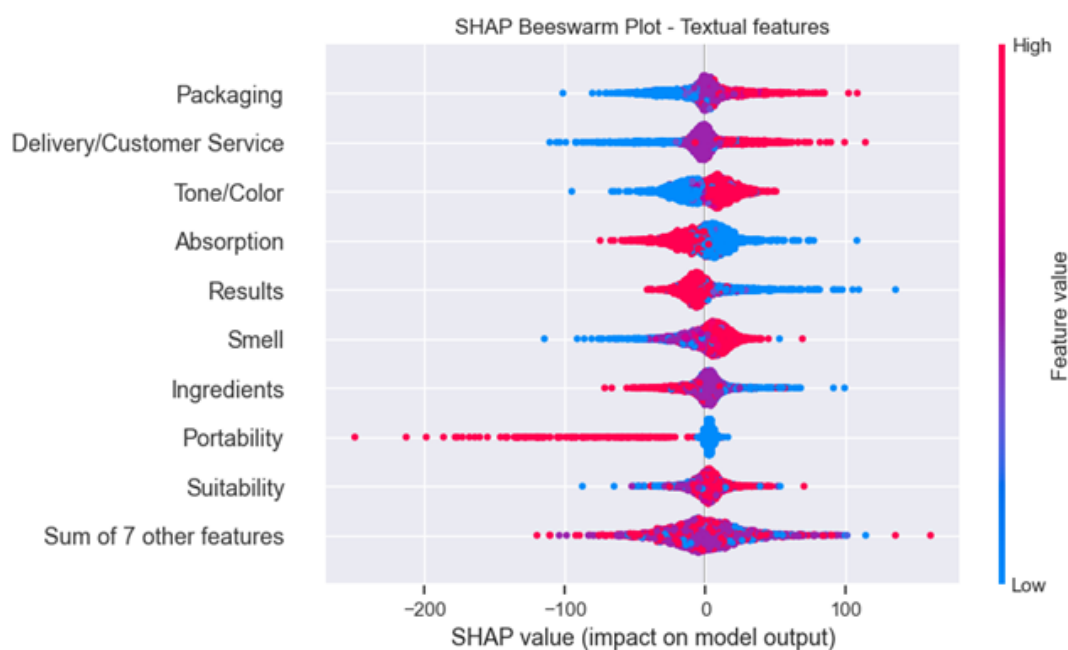


Figure 6.6: SHAP Beeswarm plot for the entire model with only sentiment features

Focusing on sentiment features, the analysis highlights that variables such as *Packaging*, *Delivery/Customer Service* and *Tone/Color* have the greatest impact. This result aligns with the online retail context, where packaging quality and efficient delivery services directly influence customer satisfaction and purchase likelihood. These factors are especially critical in e-commerce, where customers rely heavily on perceived value and service quality due to the lack of in-person evaluation.

Interestingly, some variables, such as *Absorption* and *Results*, exhibit an opposite impact on sales. Although positive values for these features generally indicate favorable per-

ceptions, they tend to correlate with lower sales, and vice versa. This counterintuitive effect may arise because the analysis considers a broad range of products across diverse segments, where the importance of specific attributes may vary significantly by product type and target audience.

To address these nuances, the analysis was further refined by categorizing products and focusing on the categories with the largest number of items: *Body Treatment*, *Face Cleanser*, *Face Moisturizer*, and *Face Serum*. This approach enables a deeper understanding of how individual product attributes influence sales within their respective categories, accounting for the unique preferences and expectations of their target markets.



Figure 6.7: SHAP Beeswarm plot for *Body Treatment* category

Observing the SHAP Beeswarm Plot for *Body Treatment* category is it possible to see how the *Units sold diff lag1* emerges as a primary driver of the model. This indicates that historical sales performance is a significant determinant of future sales predictions. The *Weekend* features exhibits a substantial negative impact when its value is low (because is a dummy variable), suggesting that sales tend to decline during weekends, at

least for this products sample.

Other key features include *Avg Rating* and *Price*, which strongly influence predictions. Higher prices tend to negatively impact sales predictions, as shown by their predominantly negative SHAP values, while higher product ratings have a positive influence. Also *NumReviews* contribute positively, indicating that the number of consumer feedback play critical roles in driving sales outcomes. Between sentiment variables with an highlight impact, there is the *Overall Satisfaction* which shows a positive relationship with expected output, as an indicator with predominantly positive SHAP values when the variable is high (red dots on the graph). In particular, the negative values (blue dots) of this impact negatively the model. This suggests that lower levels of consumer satisfaction contribute significantly to worse outcomes, in the form of decreased sales.

The chart also highlights product-specific factors, such as ASIN *B07GFV9LBX*, which show localized impacts on the model's output, likely because it is a product with a high number of sales. Additionally, time factors, such as *June*, suggest seasonal variations in sales performance.

The *Face Cleanser* category demonstrates both similarities and distinctions compared to the *Body Treatment*. Infact, once again, *Units sold diff lag1* is one of the major driver of the model, underscoring the importance of historical sales data in predicting future outcomes and also here the *Weekend* shows a negative impact on sales predictions, consistent with the findings from the previous category analyzed. Notably, the feature *Easy to use* emerges as particularly influential as a sentiment variable, with higher values correlating positively with the model output. This suggests that ease of use is a critical factor for consumer preference in the *Face Cleanser* category. Additionally, features such as *Suitability*, and *Durability* are significant, highlighting the importance of product attributes that align with consumer needs and expectations. As described in the previous Chapter the feature *Suitability*, which reflects whether a product is compatible with certain skin types (e.g., sensitive, oily, or dry). The SHAP plot shows a stron positive impact when this scores are high, indicating that products perceived as addressing skin

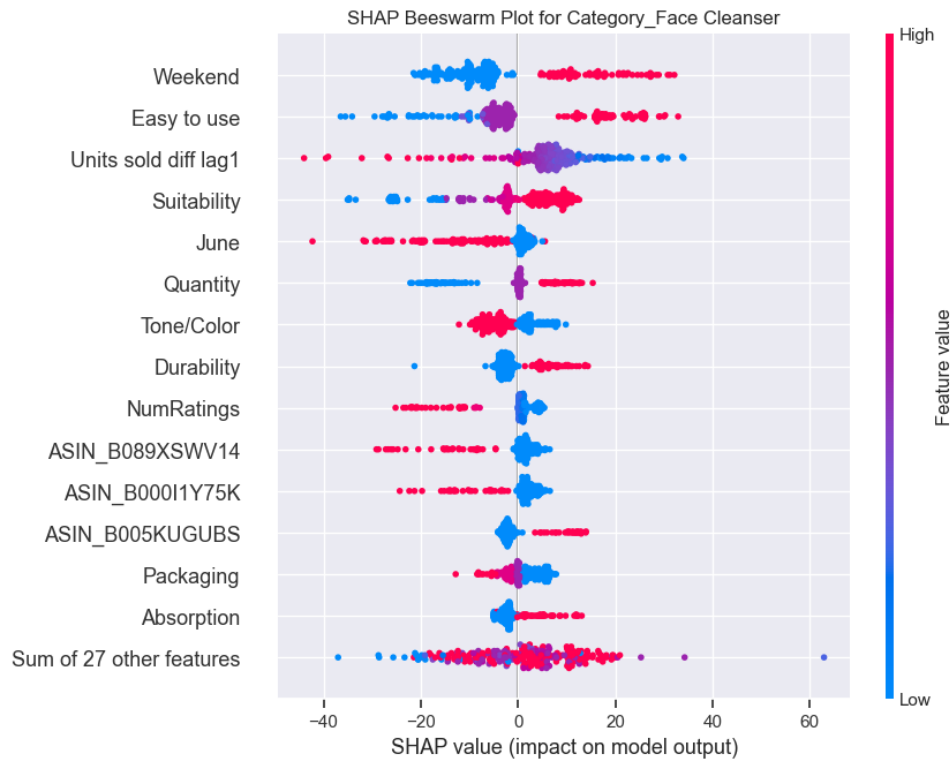


Figure 6.8: SHAP Beeswarm plot for *Face Cleanser* category

compatibility effectively are more likely to succeed. Products that explicitly cater to specific skin types (e.g., sensitive or oily skin) may perform well due to the trust they build with targeted consumers. However, a product’s broad suitability—for example, being designed for “all skin types”—can also drive positive sentiment by appealing to a wider audience. In this case, the positive SHAP values likely capture both scenarios: the specialized products which have a strongly impact with consumers who seek solutions tailored to their unique needs, such as those with sensitive or problematic skin and versatile products which gain popularity because they simplify purchasing decisions and reduce perceived risk, especially for consumers uncertain about their skin type.

Regarding the *Quantity* indicate a significant influence and probably an higher perceived or actual volume may enhance satisfaction for products expected to last a certain period, especially for those used daily, such as face cleansers. Conversely, dissatisfaction with the product’s size (e.g., feeling it is insufficient for its price or intended use) may contribute to negative outcomes. This suggests that brands could ensure that the quantity offered

aligns with consumer expectations for typical usage duration (e.g., “one bottle lasts X weeks”), infact it is correlated with the same impact with an other sentiment variable: *Durability*, especially when it is positive.

It is possible to note an unexpected dynamic, higher sentiment values for *Tone/Color* tend to have a negative impact on the target variable (*Units sold diff*). There are several potential reasons for this. Firstly, products emphasizing tone-related benefits may appeal to a smaller, niche audience rather than addressing the broader needs of consumers. In other words, focusing too much on tone improvement might limit the product’s appeal to a wider group of buyers, which could lead to fluctuations in daily sales. Another possibility is that consumer expectations are not being met. If the claims about tone-related effects are perceived as exaggerated or insufficient, consumers may be disappointed, resulting in lower sales on certain days compared to others. Additionally, as already discussed consumers prioritize other features (*Easy to use* and *Suitability*), over tone-specific benefits. This could mean that products heavily marketed around tone improvement might experience inconsistency in sales, depending on whether these other factors resonate with consumers.

At the same way, in the *Face Serum* plot, it is clear that the main driver is *Units sold diff lag1*. Other time factors such as *Weekend* and *June* month significantly influence the model, likely due to seasonality or consumer behavior. Among product attributes, *Avg Rating* has a positive impact: produts with high ratings tend to sell more. Also, item-specific variables such as ASIN *B0PZF6K9Z* show localized and significant impacts, suggesting that certain products are particularly driving sales negatively for this category. Sentiment variables such as *Tone/Color*, *Skin/Hair Issue*, *Ingredients*, *Easy to use* and *Suitability* (less) show considerable influence to the model. The *Tone/Color* variable seems to represents one of the main success levers for the *Face Serum*. This data reflects the fact that consumers strongly associate serums with the ability to improve skin tone, a quality often perceived as a sign of health, brightness and youth. This positive impact could indicate that consumers see improved tone as a key benefit, capable of satisfying

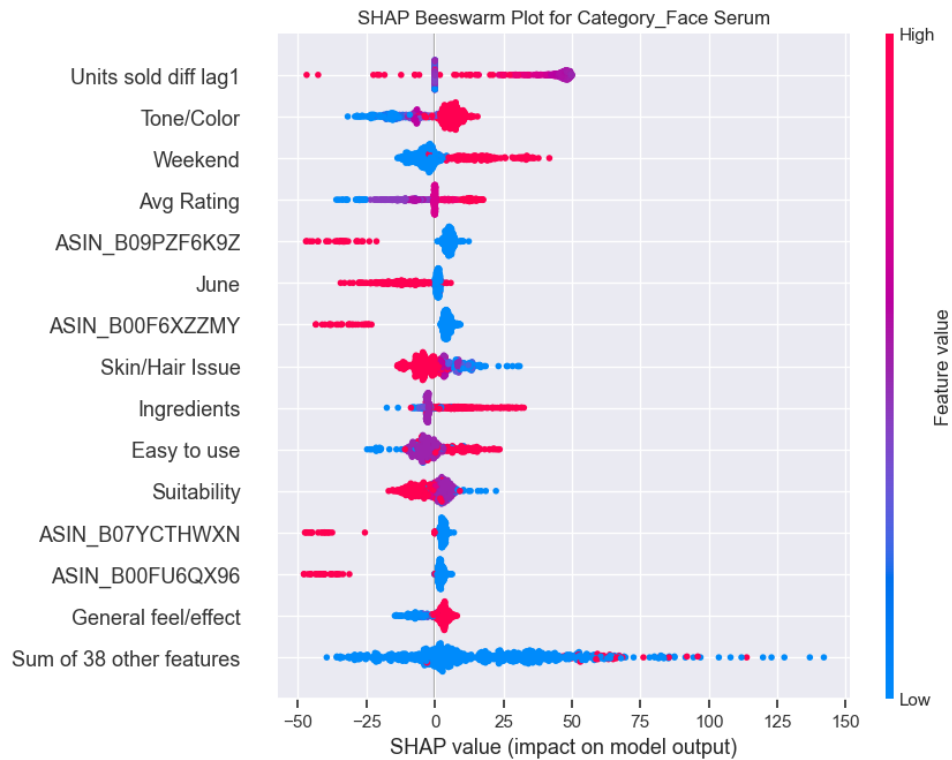


Figure 6.9: SHAP Beeswarm plot for *Face Serum* category

immediate and long-term aesthetic needs. Instead, the *Skin/Hair Issue* variable has a counterintuitive behavior in the graph. Infact, instead of a clearly positive impact when this characteristic is high, is visible that lower values tend to contribute positively to sales, while high values can negatively influence. One possibility is that serums perceived as too focused on specific concerns attract a more niche audience, reducing their appeal to a broader audience which are looking for more generic skin care solutions. This counterintuitive dynamic is also possible to assign to the variable *Suitability*. For example, a serum that is declared “perfect for dry skin” might turn off those with oily or normal skin, even though the product may still be effective for those who left those positive reviews for those products. The *Ingredients* variable also emerges as a crucial factor, especially when the values associated whis this variable are particularly positive. In *Face Serum*, consumers actively look for formulations that include active ingredients known for their specific benefits (especially for the modern consumers), such as hyaluronic acid for hydration, vitamin C and retinol which are associated with tangible benefits. Further, many

consumers are looking for products with natural ingredients, free of parabens, sulfates and artificial fragrances. At the same way, the *Easy to use* variable is also highly positive for this category. High values of this variable indicate that the product is perceived as practical, intuitive, and that integrate easily into routines, characteristics that increase the probability of being chosen and repurchased. Although less influential, the *General Feel/Effect* variable proves important in creating a lasting impression on the consumer, probably because a pleasant experience increases the likelihood that the consumer will continue to use the product and recommend it to others.

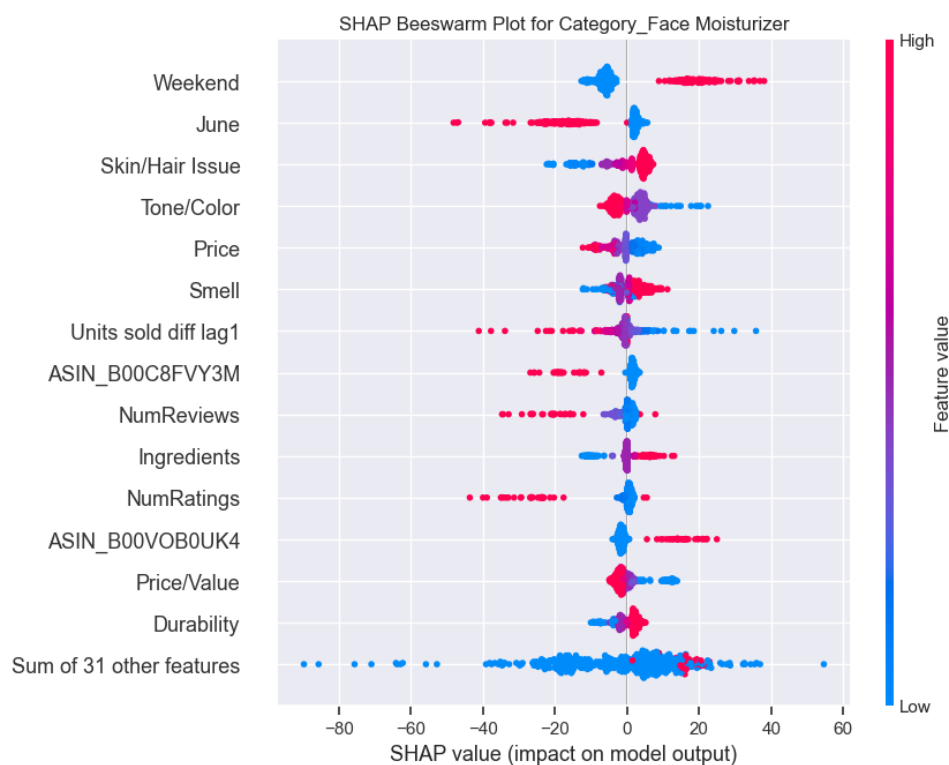


Figure 6.10: SHAP Beeswarm plot for *Face Moisturizer* category

For the *Face Moisturizer* category, there is a significant impact of the temporal variables *Weekend* and *June*, likely indicating that these products tend to be purchased more during weekends and warmer months. However, what we particularly observe is the third variable, which is a sentiment variable: *Skin/Hair Issue*. This reflects the importance of addressing specific skin issues for a moisturizer, such as dryness, irritation, acne, and redness. Here too, *Tone/Color* seems to have an unexpected effect, likely because

consumers consider tone uniformity a secondary benefit compared to addressing specific issues (*Skin/Hair Issue*). Products may not always guarantee both effects, leading to a negative impact on one of the aspects.

The variable *Smell*, on the other hand, seems to have a positive impact, although not predominant. It is evident that olfactory preferences might play an important role in the consumer's decision-making process.

Another variable worth mentioning, although with less intensity compared to others and compared to its effect in the *Face Serum* category, is *Ingredients*. Here too, they have a positive impact when the sentiment value is positive and vice versa.

In conclusion, the experiment, not only allowed for the integration of Sales Forecasting methods with advanced models for Aspect Category Sentiment Analysis but also adopted an approach to evaluate the impact of features within the analyzed markets.

In the e-commerce world, it is increasingly important to give consumers a voice through reviews. With advancements in NLP techniques, it is now possible to capture user opinions more precisely, offering new opportunities to analyze the relationship between consumer perceptions and market performance.

The study aims to improve the approach to product design, focusing particularly on the initial phase of gathering (*user requirements*). From this perspective, the objective is to propose a method to quantify the effect of consumer evaluations of product characteristics on future sales, with particular attention to the *sentiment features* identified in this experiment.

The analysis clearly shows that sentiment features play a significant role in sales, both when considering a general model for all products and when separating by specific categories. This is confirmed by the fact that sales prediction models improve significantly when these variables are included, demonstrating how crucial consumer evaluations of product characteristics are for predicting sales performance.

Furthermore, it emerges that some sentiment variables have a cross-category impact, while others are relevant to specific categories. This provides useful insights for guiding

product design decisions in the analyzed markets. For example, consumers in the *Face Cleanser* and *Face Serum* categories seem to prefer practical products that are easy to integrate into their daily routines, as indicated by the *Easy to use* variable. This suggests that a design focused on ease of use, such as easy-to-apply formulas or intuitive packaging, could enhance brand loyalty.

Variables like *Durability* and *Ingredients* are also relevant across multiple product categories. Clear statements regarding durability and transparency in formulations can help strengthen product positioning.

An interesting aspect concerns the *Tone/Color* variable, which, along with *Suitability* and *Skin/Hair Issue* (to a lesser extent), shows different dynamics between categories. In *Face Serum*, improving skin tone is a key driver, as consumers associate the product with promises of brightness and youth. Conversely, in *Face Cleanser*, skin tone uniformity appears secondary, even having a negative impact compared to other characteristics like the product's suitability for different skin types (*Suitability*).

However, there are sentiment features relevant only to specific categories. For example, *Overall Satisfaction* is particularly significant for *Body Treatments*, indicating that consumers evaluate the overall effectiveness and user experience rather than focusing on specific attributes. In *Face Moisturizers*, the importance of a pleasant sensory experience, as indicated by the *Smell* variable, plays a significant role. Finally, in *Face Serum*, a generally positive application effect (*General Feel/Effect*) seems to contribute, albeit to a lesser extent, to the product's success.

	Mean Baseline		BayesianRidge		SVR		XGBRegressor		KNN (K=5)	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
Price	42.49	23.53	2.93	0.62	41.87	21.22	3.60	0.89	4.70	1.23
Rating	0.19	0.16	0.01	0.00	0.18	0.15	0.01	0.00	0.03	0.01
Ratings	12331.43	6058.84	296.19	36.40	12349.71	4249.66	484.34	82.13	423.80	61.67
Reviews	2026.83	984.39	22.37	3.01	1907.95	616.53	50.52	8.48	125.67	11.56
Units sold	347.72	176.76	82.41	33.45	293.24	101.75	109.03	31.82	73.67	25.05

Table 6.1: Comparison of RMSE and MAE across different regression models for various variables with MICE. Note: Only KNN with $K = 5$ is shown in the table as it performed the best, but results for $K = 1, 3, 7, 10, 15, 20, 25$, and 30 were also considered.

7 | Conclusion and future work

This thesis addressed one of the main challenges in Aspect-Based Sentiment Analysis (ABSA): the difficulty of achieving a complete flow in applying these methodologies. Often, ABSA techniques remain confined to theoretical contexts, focusing on single methodologies or specific steps without materializing into viable tools for real-world decision-making. This research aimed to bridge this gap by developing methodologies and tools to address practical challenges in ABSA and expand its applicability, particularly in the online retail industry.

After a comprehensive overview of Natural Language Processing (NLP) and Deep Learning (DL) fundamentals, the thesis delves into the architecture of Transformer models, highlighting their attention mechanism. It introduces the *Text-To-Text Transfer Transformer (T5)* and its *Flan-T5* variant, explaining their main application domains. These foundational concepts provided the theoretical basis for addressing the core focus of this thesis: sentiment analysis at the aspect level.

The thesis further explored various levels of sentiment analysis application—document, sentence, and aspect levels—describing the evolution from lexicon-based approaches to modern models that leverage neural networks and Transformer architectures. This theoretical framework positioned ABSA within a broader context, shedding light on its inherent complexities and the challenges that remain unresolved.

Chapter three presented a literature review on ABSA, focusing on the field's key tasks and emphasizing the distinctions between single and multiple extractions, as well as the relationships among these processes. It traced the evolution from basic tasks such as Aspect Term Extraction (ATE) or Category Detection (ACD) to more complex frame-

works like pair and triplet extraction, culminating in the most intricate task found in the literature, Aspect Category Opinion Sentiment (ACOS).

Chapter four conducts an applied benchmark for ACOS. The dataset and annotation tool used are detailed, followed by an experimental evaluation comparing different models from the literature. This analysis identified strengths and weaknesses of the evaluated models, providing valuable insights for future improvements. Notably, it was found that *opinion terms*, often composed of multiple words, create confusion for the model and lower performance.

A key contribution of this thesis is presented in chapter five, which details the development of the PyACSA model and a comprehensive framework. This framework includes a conversion module for handling various tools and data formats, along with utilities for manual and semi-automatic data annotation (enabled by the integrated model) and report generation directly from review files. Additionally, a labeled dataset tailored for ACSA in the *Beauty and Personal Care* industry is released, with categories specifically chosen to capture the critical aspects of this field. The chapter concludes with a comparative evaluation of PyACSA against a state-of-the-art tool employing a similar approach, demonstrating the model’s effectiveness across multiple benchmark datasets. It is important to note that loading the model is resource-intensive and requires dedicated computational resources for deployment; moreover, the current interface is domain-specific. Future work should focus on streamlining deployment processes and generalizing the interface to cover multiple domains.

A noteworthy example of the tool’s adaptability has been its application to a classification task in the domain of Argument Mining [103] using the *Flan-T5 Large* pre-trained model. By slightly modifying the tool to classify arguments in a financial dataset (Cornell eRulemaking Corpus [100]), the approach outperformed the performance of $BERT_{Large}$ models both with and without contextual information (in the appendix). This success underscores the potential of the tool to extend beyond sentiment analysis and address challenges in diverse tasks.

The sixth chapter broadens the application horizon by demonstrating how the Aspect-Category Sentiment Analysis (ACSA) task can amplify the voice of consumers and support decision-making processes in product design. An innovative method is proposed to quantify the impact of consumer ratings on future sales by ranking product features based on their influence. The results reveal that not all product features are equally significant; some may even have a negative impact sales if combined suboptimally with others. Furthermore, by integrating sentiment categories into sales forecasting models, an improvement in forecast accuracy was observed, confirming the practical utility of sentiment analysis in e-commerce. Future research should validate these methodologies in contexts beyond those explored in this thesis to assess the scalability and adaptability of the developed tools. Expanding the framework to additional industries will be crucial to ensure its broader applicability and effectiveness.

This research lays a foundation for future developments in ABSA by providing an open and extensible framework. The proposed `PyACSA` model and utilities can be adapted for tasks beyond Aspect-Category Sentiment Analysis, enabling the exploration of more diverse ABSA applications (ACOS, for example). The modular nature of the framework invites extensions, such as the integration of novel tasks, expanded datasets, and other functionalities like model training.

Looking ahead, several promising avenues for future work emerge. Although this thesis has already adopted SHAP to explain model outputs, exploring complementary interpretability techniques such as LIME could be useful for providing additional local, case-specific explanations that enhance understanding of individual predictions. Moreover, while this thesis primarily focused on supervised learning approaches, exploring semi-supervised and unsupervised learning techniques may enhance the generalizability of the models, particularly in settings where labeled data are scarce. Finally, with the rapid evolution of Large Language Models (LLMs) and knowledge-augmented approaches, future research could investigate how these advanced methodologies might further enrich the ABSA process—not only boosting performance but also extending the applicability

of the framework across diverse domains.

In conclusion, this thesis has shown how ABSA, and particularly the ACSA task, can evolve from a predominantly academic endeavor into a practical and strategic tool for businesses. The developed methodologies and tools address critical practical challenges while providing valuable insights into consumer feedback. This research represents a significant step toward broader adoption of ABSA in real-world settings, laying the groundwork for further advancements and applications across diverse domains.

Bibliography

- [1] Absa-quad/data at master · isakzhang/absa-quad. URL <https://github.com/IsakZhang/ABSA-QUAD/tree/master/data>.
- [2] GitHub - zhijing-jin/ARTS_TestSet: This is our ARTS test set, an enriched test set to probe Aspect Robustness of ABSA. URL https://github.com/zhijing-jin/ARTS_TestSet.
- [3] Github - humansignal/label-studio: Label studio is a multi-type data labeling and annotation tool with standardized output format. URL <https://github.com/HumanSignal/label-studio>.
- [4] An introduction to deep learning in natural language processing: Models, techniques, and tools. *Neurocomputing*. URL <https://doi.org/10.1016/j.neucom.2021.05.103>.
- [5] Github - siat-nlp/mams-for-absa: A multi-aspect multi-sentiment dataset for aspect-based sentiment analysis. URL <https://github.com/siat-nlp/MAMS-for-ABSA/tree/master>.
- [6] Github - nustm/acos: The datasets and code of acl 2021 paper “aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions”. URL <https://github.com/NUSTM/ACOS>.
- [7] . URL <https://semeval.github.io/>.
- [8] . URL <https://alt.qcri.org/semeval2014/task4/>.
- [9] . URL <https://alt.qcri.org/semeval2015/task12/>.

- [10] . URL <https://alt.qcri.org/semEval2016/task5/>.
- [11] Fasttext. URL <https://ai.meta.com/tools/fasttext/>.
- [12] H. Akaike. A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723, 12 1974. doi: 10.1109/tac.1974.1100705. URL <https://doi.org/10.1109/tac.1974.1100705>.
- [13] M. Alzate, M. Arce-Urriza, and J. Cebollada. Online reviews and product sales: The role of review visibility. *J. Theor. Appl. Electron. Commer. Res.*, 16(4):638–669, 2021. URL <https://doi.org/10.3390/jtaer16040038>.
- [14] R. Anantha, S. Vakulenko, Z. Tu, S. Longpre, S. Pulman, and S. Chappidi. Open-domain question answering goes conversational via question rewriting. In K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tür, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 520–534. Association for Computational Linguistics, 2021. URL <https://doi.org/10.18653/v1/2021.naacl-main.44>.
- [15] N. Archak, A. Ghose, and P. G. Ipeirotis. Deriving the pricing power of product features by mining consumer reviews. *Manag. Sci.*, 57(8):1485–1509, 2011. URL <https://doi.org/10.1287/mnsc.1110.1370>.
- [16] E. O. Arkhangelskaya and S. I. Nikolenko. Deep learning for natural language processing: A survey. *Journal of Mathematical Sciences*, 273:533–582, 2023. URL <https://doi.org/10.1007/s10958-023-06519-6>.
- [17] S. Asur and B. A. Huberman. Predicting the future with social media. *CoRR*, abs/1003.5699, 2010. URL <http://arxiv.org/abs/1003.5699>.
- [18] S. Baccianella, A. Esuli, and F. Sebastiani. Sentiwordnet 3.0: An enhanced lexical

- resource for sentiment analysis and opinion mining. In N. Calzolari, K. Choukri, B. Maegaard, J. Mariani, J. Odijk, S. Piperidis, M. Rosner, and D. Tapias, editors, *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta*. European Language Resources Association, 2010. URL <http://www.lrec-conf.org/proceedings/lrec2010/summaries/769.html>.
- [19] X. Bao, Z. Wang, X. Jiang, R. Xiao, and S. Li. Aspect-based sentiment analysis with opinion tree generation. In L. D. Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 4044–4050. ijcai.org, 2022. URL <https://doi.org/10.24963/ijcai.2022/561>.
- [20] E. Bassignana, D. Brunato, M. Polignano, and A. Ramponi. Nl4ai 2023: Overview of the seventh workshop on natural language for artificial intelligence (nl4ai. 2023.
- [21] Y. Bengio, P. Y. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Trans. Neural Networks*, 5(2):157–166, 1994. URL <https://doi.org/10.1109/72.279181>.
- [22] O. Bojar, C. Federmann, M. Fishel, Y. Graham, B. Haddow, P. Koehn, and C. Monz. Findings of the 2018 conference on machine translation (WMT18). In *WMT (shared task)*, pages 272–303. Association for Computational Linguistics, 2018.
- [23] G. Bonetta, C. D. Hromei, L. Siciliani, and M. A. Stranisci. Nl4ai 2024: Overview of the eighth workshop on natural language for artificial intelligence (nl4ai 2024). 2024.
- [24] J. Bu, L. Ren, S. Zheng, Y. Yang, J. Wang, F. Zhang, and W. Wu. ASAP: A chinese review dataset towards aspect category sentiment analysis and rating prediction. In K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tür, I. Belt-

- agy, S. Bethard, R. Cotterell, T. Chakraborty, and Y. Zhou, editors, *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 2069–2079. Association for Computational Linguistics, 2021. URL <https://doi.org/10.18653/v1/2021.naacl-main.167>.
- [25] J. Bu, L. Ren, S. Zheng, Y. Yang, J. Wang, F. Zhang, and W. Wu. ASAP: A chinese review dataset towards aspect category sentiment analysis and rating prediction. *CoRR*, abs/2103.06605, 2021. URL <https://arxiv.org/abs/2103.06605>.
- [26] B. Byrne, K. Krishnamoorthi, C. Sankar, A. Neelakantan, B. Goodrich, D. Duckworth, S. Yavuz, A. Dubey, K. Kim, and A. Cedilnik. Taskmaster-1: Toward a realistic and diverse dialog dataset. In K. Inui, J. Jiang, V. Ng, and X. Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4515–4524. Association for Computational Linguistics, 2019. URL <https://doi.org/10.18653/v1/D19-1459>.
- [27] H. Cai, Y. Tu, X. Zhou, J. Yu, and R. Xia. Aspect-category based sentiment analysis with hierarchical graph convolutional network. In D. Scott, N. Bel, and C. Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 833–843. International Committee on Computational Linguistics, 2020. URL <https://doi.org/10.18653/v1/2020.coling-main.72>.
- [28] H. Cai, R. Xia, and J. Yu. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 340–

- 350, 2021. URL <https://doi.org/10.18653/v1/2021.acl-long.29>.
- [29] H. Cai, R. Xia, and J. Yu. Aspect-category-opinion-sentiment quadruple extraction with implicit aspects and opinions. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 340–350. Association for Computational Linguistics, 2021. URL <https://doi.org/10.18653/v1/2021.acl-long.29>.
- [30] H. Cai, N. Song, Z. Wang, Q. Xie, Q. Zhao, K. Li, S. Wu, S. Liu, J. Yu, and R. Xia. MEMD-ABSA: A multi-element multi-domain dataset for aspect-based sentiment analysis. *CoRR*, abs/2306.16956, 2023. URL <https://doi.org/10.48550/arXiv.2306.16956>.
- [31] O. Camburu, B. Shillingford, P. Minervini, T. Lukasiewicz, and P. Blunsom. Make up your mind! adversarial generation of inconsistent natural language explanations. In D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4157–4165. Association for Computational Linguistics, 2020. URL <https://doi.org/10.18653/v1/2020.acl-main.382>.
- [32] T. Chai and R. R. Draxler. Root mean square error (RMSE) or mean absolute error (MAE)? – Arguments against avoiding RMSE in the literature. *Geoscientific model development*, 7(3):1247–1250, 6 2014. doi: 10.5194/gmd-7-1247-2014. URL <https://doi.org/10.5194/gmd-7-1247-2014>.
- [33] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: synthetic minority over-sampling technique. *J. Artif. Intell. Res.*, 16:321–357, 2002. doi: 10.1613/JAIR.953. URL <https://doi.org/10.1613/jair.953>.
- [34] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. *CoRR*,

- abs/1603.02754, 2016. URL <http://arxiv.org/abs/1603.02754>.
- [35] T. Chen, P. Samaranayake, X. Cen, M. Qi, and Y.-C. Lan. The impact of on-line reviews on consumers' purchasing decisions: evidence from an Eye-Tracking study. *Frontiers in Psychology*, 13, 6 2022. URL <https://doi.org/10.3389/fpsyg.2022.865702>.
- [36] Y. Chen and J. Xie. Online consumer review: Word-of-mouth as a new element of marketing communication mix. *Manag. Sci.*, 54(3):477–491, 2008. URL <https://doi.org/10.1287/mnsc.1070.0810>.
- [37] Y. Chen, S. Fay, and Q. Wang. The role of Marketing in Social Media: How Online Consumer Reviews evolve. *Journal of Interactive Marketing*, 25(2):85–94, 3 2011. URL <https://doi.org/10.1016/j.intmar.2011.01.003>.
- [38] Z. Chen and T. Qian. Enhancing aspect term extraction with soft prototypes. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 2107–2117. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.164. URL <https://doi.org/10.18653/v1/2020.emnlp-main.164>.
- [39] N. Choudhary, R. Singh, V. A. Rao, and M. Shrivastava. Twitter corpus of resource-scarce languages for sentiment analysis and multilingual emoji prediction. In E. M. Bender, L. Derczynski, and P. Isabelle, editors, *Proceedings of the 27th International Conference on Computational Linguistics, COLING 2018, Santa Fe, New Mexico, USA, August 20-26, 2018*, pages 1570–1577. Association for Computational Linguistics, 2018.
- [40] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma, A. Webson, S. S. Gu, Z. Dai, M. Suzgun, X. Chen, A. Chowdhery, S. Narang, G. Mishra, A. Yu, V. Y. Zhao, Y. Huang, A. M. Dai,

- H. Yu, S. Petrov, E. H. Chi, J. Dean, J. Devlin, A. Roberts, D. Zhou, Q. V. Le, and J. Wei. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416, 2022. URL <https://doi.org/10.48550/arXiv.2210.11416>.
- [41] K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, C. Hesse, and J. Schulman. Training verifiers to solve math word problems. *CoRR*, abs/2110.14168, 2021. URL <https://arxiv.org/abs/2110.14168>.
- [42] J. Dai, H. Yan, T. Sun, P. Liu, and X. Qiu. Does syntax matter? A strong baseline for aspect-based sentiment analysis with roberta. *CoRR*, abs/2104.04986, 2021. URL <https://arxiv.org/abs/2104.04986>.
- [43] Z. Dai, C. Peng, H. Chen, and Y. Ding. A multi-task incremental learning framework with category name embedding for aspect-category sentiment analysis. *CoRR*, abs/2010.02784, 2020. URL <https://arxiv.org/abs/2010.02784>.
- [44] Z. Dai, A. T. Chaganty, V. Y. Zhao, A. Amini, Q. M. Rashid, M. Green, and K. Guu. Dialog inpainting: Turning documents into dialogs. *CoRR*, abs/2205.09073, 2022. URL <https://doi.org/10.48550/arXiv.2205.09073>.
- [45] D. A. Dickey and W. A. Fuller. Likelihood Ratio Statistics for Autoregressive Time Series with a Unit Root. *Econometrica*, 49(4):1057, 7 1981. doi: 10.2307/1912517. URL <https://doi.org/10.2307/1912517>.
- [46] T. G. Dietterich. Overfitting and undercomputing in machine learning. *ACM Comput. Surv.*, 27(3):326–327, 1995. URL <https://doi.org/10.1145/212094.212114>.
- [47] K. M. Elsayed. Mean absolute deviation: analysis and applications. *International Journal of Business and Statistical Analysis*, 2(2):63–74, 7 2015. URL <https://doi.org/10.12785/ijbsa/020201>.

- [48] Z. Fan, Z. Wu, X. Dai, S. Huang, and J. Chen. Target-oriented opinion words extraction with target-fused neural sequence labeling. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2509–2518. Association for Computational Linguistics, 2019. URL <https://doi.org/10.18653/v1/n19-1259>.
- [49] Z. Fan, Z. Wu, X. Dai, S. Huang, and J. Chen. Target-oriented opinion words extraction with target-fused neural sequence labeling. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2509–2518. Association for Computational Linguistics, 2019. URL <https://doi.org/10.18653/v1/n19-1259>.
- [50] M. Federico, M. Cettolo, L. Bentivogli, M. Paul, and S. Stüker. Overview of the IWSLT 2012 evaluation campaign. In *IWSLT*, pages 12–33. ISCA, 2012.
- [51] H. Feng and F. Shi. Deep learning in natural language processing, edited by li deng and yang liu. singapore: Springer, 2018. ISBN 9789811052088. XVII + 329 pages. *Nat. Lang. Eng.*, 27(3):373–375, 2021. doi: 10.1017/S1351324919000597. URL <https://doi.org/10.1017/S1351324919000597>.
- [52] Y. Feng, Y. Zhao, H. Zheng, Z. Li, and J. Tan. Data-driven product design toward intelligent manufacturing: A review. *International Journal of Advanced Robotic Systems*, 17(2):172988142091125, 3 2020. URL <https://doi.org/10.1177/1729881420911257>.
- [53] L. Gao, Y. Wang, T. Liu, J. Wang, L. Zhang, and J. Liao. Question-driven span labeling model for aspect-opinion pair extraction. In *Thirty-Fifth AAAI*

- Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12875–12883. AAAI Press, 2021. URL <https://doi.org/10.1609/aaai.v35i14.17523>.
- [54] M. Geva, D. Khashabi, E. Segal, T. Khot, D. Roth, and J. Berant. Did aristotle use a laptop? A question answering benchmark with implicit reasoning strategies. *CoRR*, abs/2101.02235, 2021. URL <https://arxiv.org/abs/2101.02235>.
- [55] E. Ghadery, S. Movahedi, M. J. Sabet, H. Faili, and A. Shakery. LICD: A language-independent approach for aspect category detection. In L. Azzopardi, B. Stein, N. Fuhr, P. Mayr, C. Hauff, and D. Hiemstra, editors, *Advances in Information Retrieval - 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14-18, 2019, Proceedings, Part I*, volume 11437 of *Lecture Notes in Computer Science*, pages 575–589. Springer, 2019. URL https://doi.org/10.1007/978-3-030-15712-8_37.
- [56] E. J. Hannan and B. G. Quinn. The determination of the order of an autoregression. *Journal of the Royal Statistical Society Series B (Statistical Methodology)*, 41(2): 190–195, 1 1979. URL <https://doi.org/10.1111/j.2517-6161.1979.tb01072.x>.
- [57] V. Hatzivassiloglou and K. R. McKeown. Predicting the semantic orientation of adjectives. In P. R. Cohen and W. Wahlster, editors, *35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics, Proceedings of the Conference, 7-12 July 1997, Universidad Nacional de Educación a Distancia (UNED), Madrid, Spain*, pages 174–181. Morgan Kaufmann Publishers / ACL, 1997. URL <https://aclanthology.org/P97-1023/>.

- [58] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier. An unsupervised neural attention model for aspect extraction. In R. Barzilay and M. Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 388–397. Association for Computational Linguistics, 2017. URL <https://doi.org/10.18653/v1/P17-1036>.
- [59] R. He, W. S. Lee, H. T. Ng, and D. Dahlmeier. An unsupervised neural attention model for aspect extraction. In R. Barzilay and M. Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 388–397. Association for Computational Linguistics, 2017. URL <https://doi.org/10.18653/v1/P17-1036>.
- [60] M. A. Hearst. *Direction-based text interpretation as an information access refinement*. 1992. URL <https://people.ischool.berkeley.edu/~hearst/papers/tbis-chapter.pdf>.
- [61] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735. URL <https://doi.org/10.1162/neco.1997.9.8.1735>.
- [62] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, 1997.
- [63] M. Hu and B. Liu. Mining and summarizing customer reviews. In W. Kim, R. Kohavi, J. Gehrke, and W. DuMouchel, editors, *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, Washington, USA, August 22-25, 2004*, pages 168–177. ACM, 2004. URL <https://doi.org/10.1145/1014052.1014073>.
- [64] M. Hu, S. Zhao, L. Zhang, K. Cai, Z. Su, R. Cheng, and X. Shen. CAN: constrained

- attention networks for multi-aspect sentiment analysis. *CoRR*, abs/1812.10735, 2018. URL <http://arxiv.org/abs/1812.10735>.
- [65] IBM. What is NLP? URL <https://www.ibm.com/it-it/topics/natural-language-processing>.
- [66] Q. Jiang, L. Chen, R. Xu, X. Ao, and M. Yang. A challenge dataset and effective models for aspect-based sentiment analysis. In K. Inui, J. Jiang, V. Ng, and X. Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 6279–6284. Association for Computational Linguistics, 2019. URL <https://doi.org/10.18653/v1/D19-1654>.
- [67] M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *CoRR*, abs/1705.03551, 2017.
- [68] A. Karoński, M. Hernes, E. Walaszczyk, and A. Rot. *Forecasting sales at fuel stations using a multilayer perceptron*. 1 2024. doi: 10.1007/978-981-97-5934-7\{__}18. URL https://doi.org/10.1007/978-981-97-5934-7_18.
- [69] D. Kwiatkowski, P. C. Phillips, P. Schmidt, and Y. Shin. Testing the null hypothesis of stationarity against the alternative of a unit root. *Journal of Econometrics*, 54 (1-3):159–178, 10 1992. doi: 10.1016/0304-4076(92)90104-y. URL [https://doi.org/10.1016/0304-4076\(92\)90104-y](https://doi.org/10.1016/0304-4076(92)90104-y).
- [70] T. Kwiatkowski, J. Palomaki, O. Redfield, M. Collins, A. P. Parikh, C. Alberti, D. Epstein, I. Polosukhin, J. Devlin, K. Lee, K. Toutanova, L. Jones, M. Kelcey, M. Chang, A. M. Dai, J. Uszkoreit, Q. Le, and S. Petrov. Natural questions: a benchmark for question answering research. *Trans. Assoc. Comput. Linguistics*, 7: 452–466, 2019.

- [71] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *CoRR*, abs/1910.13461, 2019. URL <http://arxiv.org/abs/1910.13461>.
- [72] H. Li. Deep learning for natural language processing: advantages and challenges. *National Science Review*, 5:24–26, 2017.
- [73] K. Li, C. Chen, X. Quan, Q. Ling, and Y. Song. Conditional augmentation for aspect term extraction via masked sequence-to-sequence generation. In D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7056–7066. Association for Computational Linguistics, 2020. URL <https://doi.org/10.18653/v1/2020.acl-main.631>.
- [74] X. Li, L. Bing, P. Li, and W. Lam. A unified model for opinion target extraction and target sentiment prediction. *CoRR*, abs/1811.05082, 2018. URL <http://arxiv.org/abs/1811.05082>.
- [75] X. Li, L. Bing, W. Zhang, and W. Lam. Exploiting BERT for end-to-end aspect-based sentiment analysis. *CoRR*, abs/1910.00883, 2019. URL <http://arxiv.org/abs/1910.00883>.
- [76] X. Li, C. Wu, and F. Mai. The effect of online reviews on product sales: A joint sentiment-topic analysis. *Inf. Manag.*, 56(2):172–184, 2019. URL <https://doi.org/10.1016/j.im.2018.04.007>.
- [77] Y. Li, D. H. Choi, J. Chung, N. Kushman, J. Schrittwieser, R. Leblond, T. Eccles, J. Keeling, F. Gimeno, A. D. Lago, T. Hubert, P. Choy, C. de Masson d’Autume, I. Babuschkin, X. Chen, P. Huang, J. Welbl, S. Gowal, A. Cherepanov, J. Molloy, D. J. Mankowitz, E. S. Robson, P. Kohli, N. de Freitas, K. Kavukcuoglu,

- and O. Vinyals. Competition-level code generation with alphacode. *CoRR*, abs/2203.07814, 2022. URL <https://doi.org/10.48550/arXiv.2203.07814>.
- [78] Y. Liang, F. Meng, J. Zhang, Y. Chen, J. Xu, and J. Zhou. An iterative multi-knowledge transfer network for aspect-based sentiment analysis. In M. Moens, X. Huang, L. Specia, and S. W. Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 1768–1780. Association for Computational Linguistics, 2021. URL <https://doi.org/10.18653/v1/2021.findings-emnlp.152>.
- [79] M. Liao, J. Li, H. Zhang, L. Wang, X. Wu, and K. Wong. Coupling global and local context for unsupervised aspect extraction. In K. Inui, J. Jiang, V. Ng, and X. Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 4578–4588. Association for Computational Linguistics, 2019. URL <https://doi.org/10.18653/v1/D19-1465>.
- [80] B. Liu. *Sentiment analysis and opinion mining*. 2012. doi: 10.1007/978-3-031-02145-9. URL <https://doi.org/10.1007/978-3-031-02145-9>.
- [81] J. Liu, Z. Teng, L. Cui, H. Liu, and Y. Zhang. Solving aspect category sentiment analysis as a text generation task. *CoRR*, abs/2110.07310, 2021. URL <https://arxiv.org/abs/2110.07310>.
- [82] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692, 2019.
- [83] Y. Liu, J. Gu, N. Goyal, X. Li, S. Edunov, M. Ghazvininejad, M. Lewis, and

- L. Zettlemoyer. Multilingual denoising pre-training for neural machine translation. *CoRR*, abs/2001.08210, 2020. URL <https://arxiv.org/abs/2001.08210>.
- [84] S. M. Lundberg and S. Lee. A unified approach to interpreting model predictions. *CoRR*, abs/1705.07874, 2017. URL <http://arxiv.org/abs/1705.07874>.
- [85] H. Luo, L. Ji, T. Li, N. Duan, and D. Jiang. GRACE: gradient harmonized and cascaded labeling for aspect-based sentiment analysis. *CoRR*, abs/2009.10557, 2020. URL <https://arxiv.org/abs/2009.10557>.
- [86] L. Luo, X. Ao, Y. Song, J. Li, X. Yang, Q. He, and D. Yu. Unsupervised neural aspect extraction with sememes. In S. Kraus, editor, *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5123–5129. ijcai.org, 2019. URL <https://doi.org/10.24963/ijcai.2019/712>.
- [87] D. Ma, S. Li, X. Zhang, and H. Wang. Interactive attention networks for aspect-level sentiment classification. *CoRR*, abs/1709.00893, 2017. URL <http://arxiv.org/abs/1709.00893>.
- [88] D. Ma, S. Li, F. Wu, X. Xie, and H. Wang. Exploring sequence-to-sequence learning in aspect term extraction. In A. Korhonen, D. R. Traum, and L. Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 3538–3547. Association for Computational Linguistics, 2019. URL <https://doi.org/10.18653/v1/p19-1344>.
- [89] Y. Ma, H. Peng, T. Khan, E. Cambria, and A. Hussain. Sentic LSTM: a hybrid network for targeted aspect-based sentiment analysis. *Cogn. Comput.*, 10(4):639–650, 2018. URL <https://doi.org/10.1007/s12559-018-9549-x>.
- [90] A. L. Maas, R. E. Daly, P. T. Pham, D. Huang, A. Y. Ng, and C. Potts. Learning

- word vectors for sentiment analysis. In *ACL*, pages 142–150. The Association for Computer Linguistics, 2011.
- [91] Y. Mao, Y. Shen, J. Yang, X. Zhu, and L. Cai. Seq2path: Generating sentiment tuples as paths of a tree. In S. Muresan, P. Nakov, and A. Villavicencio, editors, *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2215–2225. Association for Computational Linguistics, 2022. URL <https://doi.org/10.18653/v1/2022.findings-acl.174>.
- [92] S. Mensah, K. Sun, and N. Aletras. An empirical study on leveraging position embeddings for target-oriented opinion words extraction. In M. Moens, X. Huang, L. Specia, and S. W. Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 9174–9179. Association for Computational Linguistics, 2021. URL <https://doi.org/10.18653/v1/2021.emnlp-main.722>.
- [93] S. M. Mohammad. From once upon a time to happily ever after: Tracking emotions in novels and fairy tales. In K. Zervanou and P. Lendvai, editors, *Proceedings of the 5th ACL Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities, LaTeCH@ACL 2011, 24 June, 2011, Portland, Oregon, USA*, pages 105–114. The Association for Computer Linguistics, 2011. URL <https://aclanthology.org/W11-1514/>.
- [94] S. Movahedi, E. Ghadery, H. Faili, and A. Shakery. Aspect category detection via topic-attention network. *CoRR*, abs/1901.01183, 2019. URL <http://arxiv.org/abs/1901.01183>.
- [95] M. Nagamachi. Kansei engineering as a powerful consumer-oriented technology for product development. *Applied Ergonomics*, 33(3):289–294, 5 2002. URL [https://doi.org/10.1016/s0003-6870\(02\)00019-4](https://doi.org/10.1016/s0003-6870(02)00019-4).

- [96] B. O'Connor, R. Balasubramanyan, B. R. Routledge, and N. A. Smith. From tweets to polls: Linking text sentiment to public opinion time series. In W. W. Cohen and S. Gosling, editors, *Proceedings of the Fourth International Conference on Weblogs and Social Media, ICWSM 2010, Washington, DC, USA, May 23-26, 2010*. The AAAI Press, 2010. URL <http://www.aaai.org/ocs/index.php/ICWSM/ICWSM10/paper/view/1536>.
- [97] K. O'Shea and R. Nash. An introduction to convolutional neural networks. *CoRR*, abs/1511.08458, 2015. URL <http://arxiv.org/abs/1511.08458>.
- [98] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. F. Christiano, J. Leike, and R. Lowe. Training language models to follow instructions with human feedback. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022. URL http://papers.nips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html.
- [99] B. Pang, L. Lee, and S. Vaithyanathan. Thumbs up? sentiment classification using machine learning techniques. *CoRR*, cs.CL/0205070, 2002. URL <https://arxiv.org/abs/cs/0205070>.
- [100] J. Park and C. Cardie. A corpus of erulemaking user comments for measuring evaluability of arguments. In N. Calzolari, K. Choukri, C. Cieri, T. Declerck, S. Goggi, K. Hasida, H. Isahara, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, S. Piperidis, and T. Tokunaga, editors, *Proceedings of the Eleventh International Conference on Language Resources and Evaluation, LREC 2018, Miyazaki, Japan, May 7-12, 2018*. European Language Resources Association (ELRA), 2018. URL

- <http://www.lrec-conf.org/proceedings/lrec2018/summaries/679.html>.
- [101] R. Pascanu, T. Mikolov, and Y. Bengio. On the difficulty of training recurrent neural networks. In *Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013*, volume 28 of *JMLR Workshop and Conference Proceedings*, pages 1310–1318. JMLR.org, 2013. URL <http://proceedings.mlr.press/v28/pascanu13.html>.
- [102] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Z. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. Pytorch: An imperative style, high-performance deep learning library. *CoRR*, abs/1912.01703, 2019. URL <http://arxiv.org/abs/1912.01703>.
- [103] A. Peldszus and M. Stede. From argument diagrams to argumentation mining in texts: A survey. *Int. J. Cogn. Informatics Nat. Intell.*, 7(1):1–31, 2013. URL <https://doi.org/10.4018/jcini.2013010101>.
- [104] H. Peng, L. Xu, L. Bing, F. Huang, W. Lu, and L. Si. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. *CoRR*, abs/1911.01616, 2019. URL <http://arxiv.org/abs/1911.01616>.
- [105] H. Peng, L. Xu, L. Bing, F. Huang, W. Lu, and L. Si. Knowing what, how and why: A near complete solution for aspect-based sentiment analysis. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8600–8607. AAAI Press, 2020. URL <https://doi.org/10.1609/aaai.v34i05.6383>.
- [106] M. T. Pilehvar and J. Camacho-Collados. Wic: the word-in-context dataset for

- evaluating context-sensitive meaning representations. In *NAACL-HLT (1)*, pages 1267–1273. Association for Computational Linguistics, 2019.
- [107] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar. Semeval-2014 task 4: Aspect based sentiment analysis. In P. Nakov and T. Zesch, editors, *Proceedings of the 8th International Workshop on Semantic Evaluation, SemEval@COLING 2014, Dublin, Ireland, August 23-24, 2014*, pages 27–35. The Association for Computer Linguistics, 2014. URL <https://doi.org/10.3115/v1/s14-2004>.
- [108] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar, and I. Androutsopoulos. Semeval-2015 task 12: Aspect based sentiment analysis. In D. M. Cer, D. Jurgens, P. Nakov, and T. Zesch, editors, *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2015, Denver, Colorado, USA, June 4-5, 2015*, pages 486–495. The Association for Computer Linguistics, 2015. URL <https://doi.org/10.18653/v1/s15-2082>.
- [109] M. Pontiki, D. Galanis, H. Papageorgiou, I. Androutsopoulos, S. Manandhar, M. Al-Smadi, M. Al-Ayyoub, Y. Zhao, B. Qin, O. D. Clercq, V. Hoste, M. Apidianaki, X. Tannier, N. V. Loukachevitch, E. V. Kotelnikov, N. Bel, S. M. Jiménez-Zafra, and G. Eryigit. Semeval-2016 task 5: Aspect based sentiment analysis. In S. Bethard, D. M. Cer, M. Carpuat, D. Jurgens, P. Nakov, and T. Zesch, editors, *Proceedings of the 10th International Workshop on Semantic Evaluation, SemEval@NAACL-HLT 2016, San Diego, CA, USA, June 16-17, 2016*, pages 19–30. The Association for Computer Linguistics, 2016. URL <https://doi.org/10.18653/v1/s16-1002>.
- [110] H. Quan, S. Li, C. Zeng, H. Wei, and J. Hu. Big Data and AI-Driven Product Design: A survey. *Applied Sciences*, 13(16):9433, 8 2023. URL <https://doi.org/10.3390/app13169433>.

- [111] L. D. Quilio and F. Fioravanti. Evaluating the aspect-category-opinion-sentiment analysis task on a custom dataset (short paper). In E. Bassignana, D. Brunato, M. Polignano, and A. Ramponi, editors, *Proceedings of the Seventh Workshop on Natural Language for Artificial Intelligence (NL4AI 2023) co-located with 22th International Conference of the Italian Association for Artificial Intelligence (AIxIA 2023), Rome, Italy, November 6th-7th, 2023*, volume 3551 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2023. URL <https://ceur-ws.org/Vol-3551/paper13.pdf>.
- [112] L. D. Quilio and F. Fioravanti. A comprehensive framework for aspect-category sentiment analysis. In G. Bonetta, C. D. Hromei, L. Siciliani, and M. A. Stranisci, editors, *Proceedings of the Eight Workshop on Natural Language for Artificial Intelligence (NL4AI 2024) co-located with 23th International Conference of the Italian Association for Artificial Intelligence (AI*IA 2024), Bolzano, Italy, November 26th-27th, 2024*, volume 3877 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2024. URL <https://ceur-ws.org/Vol-3877/paper13.pdf>.
- [113] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020. URL <https://jmlr.org/papers/v21/20-074.html>.
- [114] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100, 000+ questions for machine comprehension of text. *CoRR*, abs/1606.05250, 2016.
- [115] J. W. Ratcliff, D. Metzener, et al. Pattern matching: The gestalt approach. *Dr. Dobb's Journal*, 13(7):46, 1988.
- [116] S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach (4th Edition)*. Pearson, 2020. URL <http://aima.cs.berkeley.edu/>.
- [117] V. Sanh, A. Webson, C. Raffel, S. H. Bach, L. Sutawika, Z. Alyafeai, A. Chaf-

- fin, A. Stiegler, A. Raja, M. Dey, M. S. Bari, C. Xu, U. Thakker, S. S. Sharma, E. Szczechla, T. Kim, G. Chhablani, N. V. Nayak, D. Datta, J. Chang, M. T. Jiang, H. Wang, M. Manica, S. Shen, Z. X. Yong, H. Pandey, R. Bawden, T. Wang, T. Neeraj, J. Rozen, A. Sharma, A. Santilli, T. Févry, J. A. Fries, R. Teehan, T. L. Scao, S. Biderman, L. Gao, T. Wolf, and A. M. Rush. Multitask prompted training enables zero-shot task generalization. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL <https://openreview.net/forum?id=9Vrb9D0WI4>.
- [118] U. Schlegel, H. Arnout, M. El-Assady, D. Oelke, and D. A. Keim. Towards a rigorous evaluation of XAI methods on time series. *CoRR*, abs/1909.07082, 2019. URL <http://arxiv.org/abs/1909.07082>.
- [119] M. Schmitt, S. Steinheber, K. Schreiber, and B. Roth. Joint aspect and polarity classification for aspect-based sentiment analysis with end-to-end neural networks. *CoRR*, abs/1808.09238, 2018. URL <http://arxiv.org/abs/1808.09238>.
- [120] K. Schouten, O. van der Weijde, F. Frasincar, and R. Dekker. Supervised and unsupervised aspect category detection for sentiment analysis with co-occurrence data. *IEEE Trans. Cybern.*, 48(4):1263–1275, 2018. URL <https://doi.org/10.1109/TCYB.2017.2688801>.
- [121] G. Schwarz. Estimating the dimension of a model. *The Annals of Statistics*, 6(2), 3 1978. URL <https://doi.org/10.1214/aos/1176344136>.
- [122] N. Shazeer and M. Stern. Adafactor: Adaptive learning rates with sublinear memory cost. *CoRR*, abs/1804.04235, 2018. URL <http://arxiv.org/abs/1804.04235>.
- [123] T. Shi, L. Li, P. Wang, and C. K. Reddy. A simple and effective self-supervised contrastive learning framework for aspect detection. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educa-*

- tional Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 13815–13824. AAAI Press, 2021. doi: 10.1609/AAAI.V35I15.17628. URL <https://doi.org/10.1609/aaai.v35i15.17628>.
- [124] A. Smith and A. Smith. 2. Online reviews, 4 2024. URL <https://www.pewresearch.org/internet/2016/12/19/online-reviews>.
- [125] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, pages 1631–1642. ACL, 2013.
- [126] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Sholeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso, A. Kluska, A. Lewkowycz, A. Agarwal, A. Power, A. Ray, A. Warstadt, A. W. Kocurek, A. Safaya, A. Tazarv, A. Xi-ang, A. Parrish, A. Nie, A. Hussain, A. Askeil, A. Dsouza, A. Slone, A. Rahane, A. S. Iyer, A. Andreassen, A. Madotto, A. Santilli, A. Stuhlmüller, A. M. Dai, A. La, A. K. Lampinen, A. Zou, A. Jiang, A. Chen, A. Vuong, A. Gupta, A. Got-tardi, A. Norelli, A. Venkatesh, A. Gholamidavoodi, A. Tabassum, A. Menezes, A. Kirubarajan, A. Mullokandov, A. Sabharwal, A. Herrick, A. Efrat, A. Erdem, A. Karakas, B. R. Roberts, B. S. Loe, B. Zoph, B. Bojanowski, B. Özyurt, B. He-dayatnia, B. Neyshabur, B. Inden, B. Stein, B. Ekmekci, B. Y. Lin, B. Howald, B. Orinion, C. Diao, C. Dour, C. Stinson, C. Argueta, C. F. Ramírez, C. Singh, C. Rathkopf, C. Meng, C. Baral, C. Wu, C. Callison-Burch, C. Waites, C. Voigt, C. D. Manning, C. Potts, C. Ramirez, C. E. Rivera, C. Siro, C. Raffel, C. Ashcraft, C. Garbacea, D. Sileo, D. Garrette, D. Hendrycks, D. Kilman, D. Roth, D. Free-man, D. Khashabi, D. Levy, D. M. González, D. Perszyk, D. Hernandez, D. Chen, D. Ippolito, D. Gilboa, D. Dohan, D. Drakard, D. Jurgens, D. Datta, D. Gan-guli, D. Emelin, D. Kleyko, D. Yuret, D. Chen, D. Tam, D. Hupkes, D. Misra, D. Buzan, D. C. Mollo, D. Yang, D. Lee, D. Schrader, E. Shutova, E. D. Cubuk, E. Segal, E. Hagerman, E. Barnes, E. Donoway, E. Pavlick, E. Rodolà, E. Lam,

E. Chu, E. Tang, E. Erdem, E. Chang, E. A. Chi, E. Dyer, E. J. Jerzak, E. Kim, E. E. Manyasi, E. Zheltonozhskii, F. Xia, F. Siar, F. Martínez-Plumed, F. Happé, F. Chollet, F. Rong, G. Mishra, G. I. Winata, G. de Melo, G. Kruszewski, G. Parascandolo, G. Mariani, G. Wang, G. Jaimovitch-López, G. Betz, G. Gur-Ari, H. Galijasevic, H. Kim, H. Rashkin, H. Hajishirzi, H. Mehta, H. Bogar, H. Shevlin, H. Schütze, H. Yakura, H. Zhang, H. M. Wong, I. Ng, I. Noble, J. Jumelet, J. Geissinger, J. Kernion, J. Hilton, J. Lee, J. F. Fisac, J. B. Simon, J. Koppel, J. Zheng, J. Zou, J. Kocon, J. Thompson, J. Wingfield, J. Kaplan, J. Radom, J. Sohl-Dickstein, J. Phang, J. Wei, J. Yosinski, J. Novikova, J. Bosscher, J. Marsh, J. Kim, J. Taal, J. H. Engel, J. Alabi, J. Xu, J. Song, J. Tang, J. Waweru, J. Burden, J. Miller, J. U. Balis, J. Batchelder, J. Berant, J. Frohberg, J. Rozen, J. Hernández-Orallo, J. Boudeman, J. Guerr, J. Jones, J. B. Tenenbaum, J. S. Rule, J. Chua, K. Kanclerz, K. Livescu, K. Krauth, K. Gopalakrishnan, K. Ignatyeva, K. Markert, K. D. Dhole, K. Gimpel, K. Omondi, K. Mathewson, K. Chiafullo, K. Shkaruta, K. Shridhar, K. McDonell, K. Richardson, L. Reynolds, L. Gao, L. Zhang, L. Dugan, L. Qin, L. C. Ochando, L. Morency, L. Moschella, L. Lam, L. Noble, L. Schmidt, L. He, L. O. Colón, L. Metz, L. K. Senel, M. Bosma, M. Sap, M. ter Hoeve, M. Farooqi, M. Faruqui, M. Mazeika, M. Baturan, M. Marelli, M. Maru, M. J. Ramírez-Quintana, M. Tolkien, M. Giulianelli, M. Lewis, M. Potthast, M. L. Leavitt, M. Hagen, M. Schubert, M. Baitemirova, M. Arnaud, M. McElrath, M. A. Yee, M. Cohen, M. Gu, M. I. Ivanitskiy, M. Starritt, M. Strube, M. Swedrowski, M. Bevilacqua, M. Yasunaga, M. Kale, M. Cain, M. Xu, M. Suzgun, M. Walker, M. Tiwari, M. Bansal, M. Aminnaseri, M. Geva, M. Gheini, M. V. T., N. Peng, N. A. Chi, N. Lee, N. G. Krakover, N. Cameron, N. Roberts, N. Doiron, N. Martinez, N. Nangia, N. Deckers, N. Muennighoff, N. S. Keskar, N. Iyer, N. Constant, N. Fiedel, N. Wen, O. Zhang, O. Agha, O. Elbaghdadi, O. Levy, O. Evans, P. A. M. Casares, P. Doshi, P. Fung, P. P. Liang, P. Vicol, P. Alipoormolabashi, P. Liao, P. Liang, P. Chang, P. Eckersley, P. M. Htut, P. Hwang, P. Milkowski,

- P. Patil, P. Pezeshkpour, P. Oli, Q. Mei, Q. Lyu, Q. Chen, R. Banjade, R. E. Rudolph, R. Gabriel, R. Habacker, R. Risco, R. Milli  re, R. Garg, R. Barnes, R. A. Saurous, R. Arakawa, R. Raymaekers, R. Frank, R. Sikand, R. Novak, R. Sitelew, R. LeBras, R. Liu, R. Jacobs, R. Zhang, R. Salakhutdinov, R. Chi, R. Lee, R. Stovall, R. Teehan, R. Yang, S. Singh, S. M. Mohammad, S. Anand, S. Dillavou, S. Shleifer, S. Wiseman, S. Gruetter, S. R. Bowman, S. S. Schoenholz, S. Han, S. Kwatra, S. A. Rous, S. Ghazarian, S. Ghosh, S. Casey, S. Bischoff, S. Gehrmann, S. Schuster, S. Sadeghi, S. Hamdan, S. Zhou, S. Srivastava, S. Shi, S. Singh, S. Asaadi, S. S. Gu, S. Pachchigar, S. Toshniwal, S. Upadhyay, S. S. Debnath, S. Shakeri, S. Thormeyer, S. Melzi, S. Reddy, S. P. Makini, S. Lee, S. Torene, S. Hatwar, S. Dehaene, S. Divic, S. Ermon, S. Biderman, S. Lin, S. Prasad, S. T. Piantadosi, S. M. Shieber, S. Misherghi, S. Kiritchenko, S. Mishra, T. Linzen, T. Schuster, T. Li, T. Yu, T. Ali, T. Hashimoto, T. Wu, T. Desbordes, T. Rothschild, T. Phan, T. Wang, T. Nkinyili, T. Schick, T. Kornev, T. Tunduny, T. Gerstenberg, T. Chang, T. Neeraj, T. Khot, T. Shultz, U. Shaham, V. Misra, V. Demberg, V. Nyamai, V. Raunak, V. V. Ramasesh, V. U. Prabhu, V. Padmakumar, V. Srikumar, W. Fedus, W. Saunders, W. Zhang, W. Vossen, X. Ren, X. Tong, X. Zhao, X. Wu, X. Shen, Y. Yaghoobzadeh, Y. Lakretz, Y. Song, Y. Bahri, Y. Choi, Y. Yang, Y. Hao, Y. Chen, Y. Belinkov, Y. Hou, Y. Hou, Y. Bai, Z. Seid, Z. Zhao, Z. Wang, Z. J. Wang, Z. Wang, and Z. Wu. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *CoRR*, abs/2206.04615, 2022. doi: 10.48550/ARXIV.2206.04615. URL <https://doi.org/10.48550/arXiv.2206.04615>.
- [127] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.*, 15(1):1929–1958, 2014. URL <https://dl.acm.org/doi/10.5555/2627435.2670313>.

- [128] P. Stenetorp, S. Pyysalo, G. Topic, T. Ohta, S. Ananiadou, and J. Tsujii. brat: a web-based tool for nlp-assisted text annotation. In W. Daelemans, M. Lapata, and L. Màrquez, editors, *EACL 2012, 13th Conference of the European Chapter of the Association for Computational Linguistics, Avignon, France, April 23-27, 2012*, pages 102–107. The Association for Computer Linguistics, 2012. URL <https://aclanthology.org/E12-2021/>.
- [129] C. Sun, L. Huang, and X. Qiu. Utilizing BERT for aspect-based sentiment analysis via constructing auxiliary sentence. *CoRR*, abs/1903.09588, 2019. URL <http://arxiv.org/abs/1903.09588>.
- [130] K. Sun, R. Zhang, S. Mensah, Y. Mao, and X. Liu. Aspect-level sentiment analysis via convolution over dependency tree. In K. Inui, J. Jiang, V. Ng, and X. Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 5678–5687. Association for Computational Linguistics, 2019. URL <https://doi.org/10.18653/v1/D19-1569>.
- [131] C. Sutton and A. McCallum. An introduction to conditional random fields. *Found. Trends Mach. Learn.*, 4(4):267–373, 2012. URL <https://doi.org/10.1561/22000000013>.
- [132] D. Tang, B. Qin, X. Feng, and T. Liu. Effective lstms for target-dependent sentiment classification. In N. Calzolari, Y. Matsumoto, and R. Prasad, editors, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 3298–3307. ACL, 2016. URL <https://aclanthology.org/C16-1311/>.
- [133] R. S. Teegavarapu. *Imputation methods for missing hydrometeorological data estimation*. 1 2024. URL <https://doi.org/10.1007/978-3-031-60946-6>.

- [134] M. Teixeira, J. M. Oliveira, and P. Ramos. Enhancing Hierarchical Sales Forecasting with Promotional Data: A Comparative Study Using ARIMA and Deep Neural Networks. *Machine Learning and Knowledge Extraction*, 6(4):2659–2687, 11 2024. doi: 10.3390/make6040128. URL <https://doi.org/10.3390/make6040128>.
- [135] F. M. Thiesing and O. Vornberger. Forecasting sales using neural networks. In B. Reusch, editor, *Computational Intelligence, Theory and Applications, International Conference, 5th Fuzzy Days, Dortmund, Germany, April 28-30, 1997, Proceedings*, volume 1226 of *Lecture Notes in Computer Science*, pages 321–328. Springer, 1997. URL https://doi.org/10.1007/3-540-62868-1_125.
- [136] R. Tiwari, R. Kumar, R. Gera, and S. Srivastava. On comparing the performances of MLP and RBFN on sales forecasting problem. *International Journal of Information Technology*, 14(1):301–309, 11 2019. doi: 10.1007/s41870-019-00402-x. URL <https://doi.org/10.1007/s41870-019-00402-x>.
- [137] A. Torfi, R. A. Shirvani, Y. Keneshloo, N. Tavaf, and E. A. Fox. Natural language processing advancements by deep learning: A survey. *CoRR*, abs/2003.01200, 2020. URL <https://arxiv.org/abs/2003.01200>.
- [138] S. Tulkens and A. van Cranenburgh. Embarrassingly simple unsupervised aspect extraction. *CoRR*, abs/2004.13580, 2020. URL <https://arxiv.org/abs/2004.13580>.
- [139] P. D. Turney. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. *CoRR*, cs.LG/0212032, 2002. URL <http://arxiv.org/abs/cs/0212032>.
- [140] P. D. Turney and M. L. Littman. Measuring praise and criticism: Inference of semantic orientation from association. *CoRR*, cs.CL/0309034, 2003. URL <http://arxiv.org/abs/cs/0309034>.
- [141] S. van Buuren and K. Groothuis-Oudshoorn. mice: Multivariate imputation by

- chained equations in r. *Journal of Statistical Software*, 45(3):1–67, 2011. doi: 10.18637/jss.v045.i03.
- [142] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL <http://arxiv.org/abs/1706.03762>.
- [143] S. Velampalli, C. Muniyappa, and A. Saxena. Performance evaluation of sentiment analysis on text and emoji data using End-to-End, transfer learning, distributed and explainable AI models. *Journal of Advances in Information Technology*, 13(2), 1 2022. doi: 10.12720/jait.13.2.167-172. URL <https://doi.org/10.12720/jait.13.2.167-172>.
- [144] M. Velmurugan, C. Ouyang, C. Moreira, and R. Sindhgatta. Evaluating stability of post-hoc explanations for business process predictions. In H. Hacid, O. Kao, M. Mecella, N. Moha, and H. Paik, editors, *Service-Oriented Computing - 19th International Conference, ICSOC 2021, Virtual Event, November 22-25, 2021, Proceedings*, volume 13121 of *Lecture Notes in Computer Science*, pages 49–64. Springer, 2021. URL https://doi.org/10.1007/978-3-030-91431-8_4.
- [145] A. P. B. Veyseh, N. Nouri, F. Dernoncourt, D. Dou, and T. H. Nguyen. Introducing syntactic structures into target opinion word extraction with deep learning. *CoRR*, abs/2010.13378, 2020. URL <https://arxiv.org/abs/2010.13378>.
- [146] D. Vo and Y. Zhang. Target-dependent twitter sentiment classification with rich automatic features. In Q. Yang and M. J. Wooldridge, editors, *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, pages 1347–1353. AAAI Press, 2015. URL <http://ijcai.org/Abstract/15/194>.
- [147] H. Wan, Y. Yang, J. Du, Y. Liu, K. Qi, and J. Z. Pan. Target-aspect-sentiment joint detection for aspect-based sentiment analysis. In *The Thirty-Fourth AAAI*

- Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9122–9129. AAAI Press, 2020. URL <https://doi.org/10.1609/aaai.v34i05.6447>.
- [148] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. *CoRR*, abs/1804.07461, 2018.
- [149] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *CoRR*, abs/1905.00537, 2019.
- [150] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. *CoRR*, abs/1905.00537, 2019. URL <http://arxiv.org/abs/1905.00537>.
- [151] Q. Wang, Z. Wen, Q. Zhao, M. Yang, and R. Xu. Progressive self-training with discriminator for aspect term extraction. In M. Moens, X. Huang, L. Specia, and S. W. Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 257–268. Association for Computational Linguistics, 2021. URL <https://doi.org/10.18653/v1/2021.emnlp-main.23>.
- [152] W. Wang, S. J. Pan, D. Dahlmeier, and X. Xiao. Coupled multi-layer attentions for co-extraction of aspect and opinion terms. In S. Singh and S. Markovitch, editors, *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 3316–3322. AAAI Press, 2017. URL <https://doi.org/10.1609/aaai.v31i1.10974>.

- [153] Y. Wang, M. Huang, X. Zhu, and L. Zhao. Attention-based LSTM for aspect-level sentiment classification. In J. Su, X. Carreras, and K. Duh, editors, *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 606–615. The Association for Computational Linguistics, 2016. URL <https://doi.org/10.18653/v1/d16-1058>.
- [154] Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Naik, A. Ashok, A. S. Dhanasekaran, A. Arunkumar, D. Stap, E. Pathak, G. Karamanolakis, H. G. Lai, I. Purohit, I. Mondal, J. Anderson, K. Kuznia, K. Doshi, K. K. Pal, M. Patel, M. Moradshahi, M. Parmar, M. Purohit, N. Varshney, P. R. Kaza, P. Verma, R. S. Puri, R. Karia, S. Doshi, S. K. Sampat, S. Mishra, S. R. A, S. Patro, T. Dixit, and X. Shen. Super-naturalinstructions: Generalization via declarative instructions on 1600+ NLP tasks. In Y. Goldberg, Z. Kozareva, and Y. Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 5085–5109. Association for Computational Linguistics, 2022. URL <https://doi.org/10.18653/v1/2022.emnlp-main.340>.
- [155] M. Wankhade, A. C. S. Rao, and C. Kulkarni. A survey on sentiment analysis methods, applications, and challenges. *Artif. Intell. Rev.*, 55(7):5731–5780, 2022. URL <https://doi.org/10.1007/s10462-022-10144-1>.
- [156] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. Finetuned language models are zero-shot learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net, 2022. URL <https://openreview.net/forum?id=gEZrGCozdqR>.
- [157] J. Wiebe. Identifying subjective characters in narrative. In *13th International*

- Conference on Computational Linguistics, COLING 1990, University of Helsinki, Finland, August 20-25, 1990*, pages 401–406, 1990. URL <https://aclanthology.org/C90-2069/>.
- [158] J. Wiebe. Tracking point of view in narrative. *CoRR*, abs/cmp-lg/9407019, 1994. URL <http://arxiv.org/abs/cmp-lg/9407019>.
- [159] J. Wiebe, R. F. Bruce, and T. P. O’Hara. Development and use of a gold-standard data set for subjectivity classifications. In R. Dale and K. W. Church, editors, *27th Annual Meeting of the Association for Computational Linguistics, University of Maryland, College Park, Maryland, USA, 20-26 June 1999*, pages 246–253. ACL, 1999. URL <https://aclanthology.org/P99-1032/>.
- [160] C. Wu, Q. Xiong, H. Yi, Y. Yu, Q. Zhu, M. Gao, and J. Chen. Multiple-element joint detection for aspect-based sentiment analysis. *Knowl. Based Syst.*, 223:107073, 2021. URL <https://doi.org/10.1016/j.knosys.2021.107073>.
- [161] M. Wu, W. Wang, and S. J. Pan. Deep weighted maxsat for aspect-based opinion extraction. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 5618–5628. Association for Computational Linguistics, 2020. doi: 10.18653/V1/2020.EMNLP-MAIN.453. URL <https://doi.org/10.18653/v1/2020.emnlp-main.453>.
- [162] S. Wu, H. Fei, Y. Ren, D. Ji, and J. Li. Learn from syntax: Improving pair-wise aspect and opinion terms extraction with rich syntactic knowledge. In Z. Zhou, editor, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 3957–3963. ijcai.org, 2021. URL <https://doi.org/10.24963/ijcai.2021/545>.
- [163] Z. Wu and D. C. Ong. Context-guided BERT for targeted aspect-based senti-

- ment analysis. *CoRR*, abs/2010.07523, 2020. URL <https://arxiv.org/abs/2010.07523>.
- [164] Z. Wu, F. Zhao, X. Dai, S. Huang, and J. Chen. Latent opinions transfer network for target-oriented opinion words extraction. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 9298–9305. AAAI Press, 2020. URL <https://doi.org/10.1609/aaai.v34i05.6469>.
- [165] X. Xing, Z. Jin, D. Jin, B. Wang, Q. Zhang, and X. Huang. Tasty burgers, soggy fries: Probing aspect robustness in aspect-based sentiment analysis. In B. Webber, T. Cohn, Y. He, and Y. Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 3594–3605. Association for Computational Linguistics, 2020. URL <https://doi.org/10.18653/v1/2020.emnlp-main.292>.
- [166] H. Xu, B. Liu, L. Shu, and P. S. Yu. Double embeddings and cnn-based sequence labeling for aspect extraction. *CoRR*, abs/1805.04601, 2018. URL <http://arxiv.org/abs/1805.04601>.
- [167] H. Xu, B. Liu, L. Shu, and P. S. Yu. BERT post-training for review reading comprehension and aspect-based sentiment analysis. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 2324–2335. Association for Computational Linguistics, 2019. URL <https://doi.org/10.18653/v1/n19-1242>.
- [168] L. Xu, H. Li, W. Lu, and L. Bing. Position-aware tagging for aspect sentiment

- triplet extraction. *CoRR*, abs/2010.02609, 2020. URL <https://arxiv.org/abs/2010.02609>.
- [169] L. Xu, H. Li, W. Lu, and L. Bing. Position-aware tagging for aspect sentiment triplet extraction. *CoRR*, abs/2010.02609, 2020. URL <https://arxiv.org/abs/2010.02609>.
- [170] L. Xu, Y. K. Chia, and L. Bing. Learning span-level interactions for aspect sentiment triplet extraction. *CoRR*, abs/2107.12214, 2021. URL <https://arxiv.org/abs/2107.12214>.
- [171] H. Yang and K. Li. PyABSA, 2023. URL <https://github.com/yangheng95/PyABSA>.
- [172] H. Yang, C. Zhang, and K. Li. Pyabsa: A modularized framework for reproducible aspect-based sentiment analysis. In I. Frommholz, F. Hopfgartner, M. Lee, M. Oakes, M. Lalmas, M. Zhang, and R. L. T. Santos, editors, *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, pages 5117–5122. ACM, 2023. URL <https://doi.org/10.1145/3583780.3614752>.
- [173] M. Yasunaga and P. Liang. Graph-based, self-supervised program repair from diagnostic feedback. *CoRR*, abs/2005.10636, 2020. URL <https://arxiv.org/abs/2005.10636>.
- [174] H. Yin, S. Zheng, W. Yeoh, and J. Ren. How online review richness impacts sales: An attribute substitution perspective. *J. Assoc. Inf. Sci. Technol.*, 72(7):901–917, 2021. URL <https://doi.org/10.1002/asi.24457>.
- [175] Y. Yin, F. Wei, L. Dong, K. Xu, M. Zhang, and M. Zhou. Unsupervised word and dependency path embeddings for aspect term extraction. In S. Kambhampati, editor, *Proceedings of the Twenty-Fifth International Joint Conference on Artificial*

- Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2979–2985. IJCAI/AAAI Press, 2016. URL <http://www.ijcai.org/Abstract/16/423>.
- [176] D. H. F. Yip, E. L. Hines, and W. W. H. Yu. Application of artificial neural networks in sales forecasting. In *Proceedings of International Conference on Neural Networks (ICNN'97), Houston, TX, USA, June 9-12, 1997*, pages 2121–2124. IEEE, 1997. URL <https://doi.org/10.1109/ICNN.1997.614233>.
- [177] J. Yu, J. Jiang, and R. Xia. Global inference for aspect and opinion terms co-extraction based on multi-task neural networks. *IEEE ACM Trans. Audio Speech Lang. Process.*, 27(1):168–177, 2019. URL <https://doi.org/10.1109/TASLP.2018.2875170>.
- [178] C. Zhang, Q. Li, and D. Song. Aspect-based sentiment classification with aspect-specific graph convolutional networks. *CoRR*, abs/1909.03477, 2019. URL <http://arxiv.org/abs/1909.03477>.
- [179] H. Zhang and Z. Wu. The Generalized Fisher’s Combination and AccurateP-Value Calculation under Dependence. *Biometrics*, 79(2):1159–1172, 2 2022. doi: 10.1111/biom.13634. URL <https://doi.org/10.1111/biom.13634>.
- [180] H. Zhang, Y. Cheah, O. M. Alyasiri, and J. An. Exploring aspect-based sentiment quadruple extraction with implicit aspects, opinions, and chatgpt: a comprehensive survey. *Artif. Intell. Rev.*, 57(2):17, 2024. URL <https://doi.org/10.1007/s10462-023-10633-x>.
- [181] H. Zhang, Y. Cheah, O. M. Alyasiri, and J. An. Exploring aspect-based sentiment quadruple extraction with implicit aspects, opinions, and chatgpt: a comprehensive survey. *Artif. Intell. Rev.*, 57(2):17, 2024. URL <https://doi.org/10.1007/s10462-023-10633-x>.
- [182] W. Zhang, Y. Deng, X. Li, Y. Yuan, L. Bing, and W. Lam. Aspect sentiment quad prediction as paraphrase generation. In M. Moens, X. Huang, L. Specia, and S. W.

- Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 9209–9219. Association for Computational Linguistics, 2021. URL <https://doi.org/10.18653/v1/2021.emnlp-main.726>.
- [183] W. Zhang, Y. Deng, X. Li, Y. Yuan, L. Bing, and W. Lam. Aspect sentiment quad prediction as paraphrase generation. In M. Moens, X. Huang, L. Specia, and S. W. Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 9209–9219. Association for Computational Linguistics, 2021. URL <https://doi.org/10.18653/v1/2021.emnlp-main.726>.
- [184] W. Zhang, X. Li, Y. Deng, L. Bing, and W. Lam. Towards generative aspect-based sentiment analysis. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 2: Short Papers), Virtual Event, August 1-6, 2021*, pages 504–510. Association for Computational Linguistics, 2021. URL <https://doi.org/10.18653/v1/2021.acl-short.64>.
- [185] W. Zhang, X. Li, Y. Deng, L. Bing, and W. Lam. A survey on aspect-based sentiment analysis: Tasks, methods, and challenges. *IEEE Trans. Knowl. Data Eng.*, 35(11):11019–11038, 2023. URL <https://doi.org/10.1109/TKDE.2022.3230975>.
- [186] X. Zhang, J. J. Zhao, and Y. LeCun. Character-level convolutional networks for text classification. *CoRR*, abs/1509.01626, 2015.
- [187] H. Zhao, L. Huang, R. Zhang, Q. Lu, and H. Xue. Spanmlt: A span-based multi-task learning framework for pair-wise aspect and opinion terms extraction. In D. Jurafsky, J. Chai, N. Schluter, and J. R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020*,

- Online, July 5-10, 2020*, pages 3239–3248. Association for Computational Linguistics, 2020. URL <https://doi.org/10.18653/v1/2020.acl-main.296>.
- [188] J. Zhou, H. Yang, Y. He, H. Mou, and J. Yang. A unified one-step solution for aspect sentiment quad prediction. *CoRR*, abs/2306.04152, 2023. URL <https://doi.org/10.48550/arXiv.2306.04152>.
- [189] X. Zhou, X. Wan, and J. Xiao. Representation learning for aspect category detection in online reviews. In B. Bonet and S. Koenig, editors, *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 417–424. AAAI Press, 2015. URL <https://doi.org/10.1609/aaai.v29i1.9194>.

8 | Appendix

Table 8.1: Performance comparison of different models to classify Argumentation in the CDCP Dataset.

Model	Precision	Recall	F1-Score
BERT-Large	0.8395	0.8294	0.8331
BERT-Large with Context	0.8420	0.8333	0.8364
PyACSA Adapted to AM	0.8513	0.8480	0.8487

List of Figures

1.1	Transformers architecture	8
1.2	Task examples with T5	15
3.1	The relations of single and multiple ABSA tasks	41
4.1	Example of annotation template in LabelStudio with hidden categories .	61
5.1	Annotations splitted by categories and polarities	70
5.2	Data converter module	76
5.3	Manual annotation module	77
5.4	Semi-automatic annotation module. In this case the model predicts two tuples: {Absorption, Negative} and {Texture/Thickness, Negative} . . .	78
5.5	Bar chart showing review polarity per category	80
5.6	Sentiment Indexes	80
6.1	Buybox seller of an Amazon product	92
6.2	An example of a customer review in the dataset.	97
6.3	Sentiment Index over the time for “philosophy amazing grace firming body emulsion” product	100
6.4	Comparison of Actual Values with Sales Forecast Values for two products	111
6.5	SHAP Beeswarm plot for the entire model	112
6.6	SHAP Beeswarm plot for the entire model with only sentiment features .	113
6.7	SHAP Beeswarm plot for <i>Body Treatment</i> category	114
6.8	SHAP Beeswarm plot for <i>Face Cleanser</i> category	116

6.9	SHAP Beeswarm plot for <i>Face Serum</i> category	118
6.10	SHAP Beeswarm plot for <i>Face Moisturizer</i> category	119

List of Tables

3.1	Single ABSA Tasks	34
4.1	Number of sentences and annotations in the training and test datasets .	61
4.2	Tool settings used for training	66
4.3	Results of the experiments on the ACOS task with different tools	67
4.4	Results obtained by using the PyABSA tool with the <i>T5-large</i> model on some simpler ABSA tasks	68
5.1	Datasets used for the experimental evaluation	81
5.2	Results of the experimental evaluation on different datasets for ACSA task	82
6.2	List of U.S. Holidays Considered for the Analysis	96
6.3	Summary of features including their definitions, means, and standard de- viations (for numerical features) considering all variables.	98
6.4	Some products used in the dataset	102
6.5	Summary of all features including definitions, means, and std. deviations (for numerical features).	103
6.6	Performance Metrics of the Top Reviews Detection Model	108
6.7	Confusion Matrix of the Top Reviews Detection model	109
6.8	Model Performance, Number of Products, and Fisher Test Results	110
6.1	Comparison of RMSE and MAE across different regression models with MICE	122

8.1	Performance comparison of different models to classify Argumentation in the CDCP Dataset.	161
-----	---	-----

Acknowledgements

I wish to express my deepest gratitude to my supervisors, Professors Fabio Fioravanti, Luca Moscardelli, and Maurizio Parton, for giving me the opportunity to take on this challenge. A special thank you goes to Professor Fabio Fioravanti for his constant availability, patience, and guidance, which have been a fundamental point of reference throughout these years.

I am sincerely grateful to Professor Viktor Pekar for warmly welcoming me to Aston University, a place I had the privilege of calling 'home' during a period of my research. His hospitality and intellectual contribution have deeply enriched this experience.

A special thanks to Fabio Di Memmo, my company supervisor, for welcoming me into AptarGroup with humanity, foresight, and generosity that go beyond a mere professional role. His support, trust, and advice have been a guiding light for me, both in work and personal growth.

I dedicate this work to my family, the pillar of my life, and to myself, for believing and persevering even in the most difficult moments. A thought also goes to all those who believe in themselves, but not too much, because it is in the search for balance between ambition and humility that true strength to move forward is found.

The right path is the one you walk without looking back, with determination and trust, always.

Ad Maiora Semper

