Master Thesis

Multi-Label Topic Classification of Client Feedback in the Governance Domain

Csenge Szabó

a thesis submitted in partial fulfilment of the requirements for the degree of

MA Linguistics
(Text Mining)

Vrije Universiteit Amsterdam

Computational Lexicology and Terminology Lab
Department of Language and Communication
Faculty of Humanities

Supervised by: Dr. Ilia Markov, Sandra Blok
2^{nd} reader: Dr. Isa Maks

Submitted: June 28, 2024

Abstract

The thesis focuses on the multi-label topic classification of written client feedback collected from the governance domain through surveys. Multi-label topic classification involves assigning more than one topic label to a particular text instance from a predefined set of topics. For this purpose, we compare a traditional machine learning classifier (Support Vector Machines) with a more recent transformer-based model (fine-tuned BERT) that currently shows state-of-the-art performance for the majority of Natural Language Processing tasks. Since the topic labels in the dataset are structured into main topics and corresponding subtopics, we experiment with one-step and two-step classification approaches. The former implies the classification of instances for all topic labels at once, while the latter means first predicting main topic labels and then subtopic labels. In order to address the imbalanced nature of the dataset, various data adaptation and data balancing techniques are explored, namely (i) undersampling aimed to reduce the prevalence of overrepresented subtopic classes to the average distribution, and (ii) oversampling aimed to generate synthetic data for underrepresented subtopic classes using a generative large language model, GPT-4. We aim to determine the best approach for multi-label topic classification on the provided dataset using a combination of the aforementioned approaches.

The two-step classification approach, which includes first predicting the main topics and then subtopics, demonstrated enhanced efficiency compared to the one-step approach with both classifiers. The results of the experiments indicate that the fine-tuned BERT model trained on the oversampled dataset achieved superior performance with a macro-averaged F1-score of 0.66. The study highlights the effectiveness of synthetic data in improving classifier performance for underrepresented classes for imbalanced datasets. The research contributes to a better understanding of applying Natural Language Processing techniques for Topic Classification in the governance sector, providing insights into the challenges of handling multi-labeled, imbalanced datasets.

Keywords: Text Mining, Natural Language Processing, Machine Learning, Transfer Learning, Multi-Label Topic Classification, Artificial Intelligence, Transformer, Governance Domain, Customer Experience, Imbalanced Data, Synthetic Data Generation

Declaration of Authorship

I, Csenge Szabó, declare that this thesis, titled *Multi-Label Topic Classification of Client Feedback in the Governance Domain* and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a MA Linguistics degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Date: 27-06-2024

Signed: 

Acknowledgments

First and foremost, I would like to express my gratitude to my supervisors, Dr. Ilia Markov and Sandra Blok, for their support and guidance throughout my research. Their expertise and critical insights have been invaluable for the completion of this work.

I am also grateful to all faculty members of the CLTL Lab at VU Amsterdam for their excellent guidance throughout my Master's program. My gratitude extends to my colleagues at MarketResponse for their support and encouragement throughout my thesis internship.

I would like to express my heartfelt thanks to my peers, Murat Ertaş, Tessel Haagen, Christina Karavida and Selin Açikel, who have supported me not only throughout this thesis but also during the projects we completed together during the Master's degree.

Finally, I thank my friends and family for standing by my side and cheering me on through challenging times. Without your support, I could not have completed this thesis, for which I will always be grateful.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

A large number of companies provide products and services to their customers. Following a successful purchase, a company wants to gain insights into the customer's opinion about its performance, services and overall satisfaction with the experience. Analyzing Customer Experience (CX) is an essential practice for successful enterprises aiming to improve their operations, since it revolves around the client's cognitive, emotional, behavioral, sensorial and social experiences throughout the purchase journey (Lemon and Verhoef, 2016). Implementing CX analysis strategies provides a valuable way to collect customer feedback concerning the company's products and services, which can be used to increase customer satisfaction in the future. In order to gain a deeper understanding of areas needing improvement, the company may decide to implement a CX strategy, for instance by sending surveys to their customers. The first part of such surveys often includes rating scales to measure customer satisfaction regarding the provision of services. While this score is a valuable starting point for evaluating CX, it does not provide customers the opportunity to reflect on their personal experiences and sentiment about the business in detail. In the next parts of the survey, user-generated content is collected through open-ended questions to gather more fine-grained information about the purchase journey. These questions allow the customer to provide detailed feedback about their individual experience and elaborate on points that may not have been previously addressed. Some time has passed, and the company now considers evaluating the incoming responses from the surveys. Considering that a large amount of data has been collected, and the data is partially in an unstructured format, the question arises: How should the company reach this goal? Manual approaches face limitations due to their time-consuming and resource-intensive nature, whereas automated methods offer a solution to extract insights even for large-scale projects.

Extracting this valuable information from text data is a unique challenge due to its unstructured nature, for which Natural Language Processing (NLP) offers a solution. NLP combines knowledge from the fields of Computer Science, Artificial Intelligence and Linguistics to create models that can understand and process human language. In other words, it can be described as a toolkit of computational techniques that can help users with the automatic analysis and representation of language data (Chowdhary, 2020). In the context of CX, NLP offers a variety of techniques for analyzing large volumes of textual data to gain insights into customer sentiment, identify recurring topics, and understand the overall customer journey. The aim of this thesis work is to leverage the power of NLP to address a task related to CX. More specifically, this study compares the performance of two state-of-the-art models, namely a conventional

machine learning classifier, Support Vector Machines (Cortes and Vapnik, 1995), and a transformer-based model, Bidirectional Encoder Representations from Transformers (Devlin et al., 2018), for the task of multi-label topic classification. The comparative experiments will be conducted using the dataset provided by the internship company.

1.1 Problem Description

The objective of the thesis is to solve an NLP task, namely multi-label topic classification within the realm of CX analysis in the governance domain. The data for this study originates from surveys distributed to clients via email following their visit at a major Dutch governmental institution. The research is conducted in collaboration with MarketResponse[1], a Dutch software company specializing in CX analytics in order to bridge the gap between customer feedback and business performance. As CX data often takes the form of text data, the implementation of NLP tools is fundamental.

The collected survey responses belong to the governance domain and are highly homogeneous, since the pieces of feedback reflect customer experiences with the same institution. The first step in extracting insights from the data is the categorization of each feedback item. Specific feedback instances might relate for example to *Employee attitude & behavior*, *Digital possibilities* or *Physical service provision*, or potentially a combination of these categories. This sorting process allows companies to identify the frequently recurring themes and pinpoint areas within their service provision that require improvement. A traditional approach would be the manual categorization of text sequences using a predefined set of categories. Although it seems to be a straightforward solution, this method is both time-consuming and economically inefficient for large datasets. In today's technologically driven environment, companies receive an enormous volume of survey responses on a daily basis, which renders manual categorization impractical. Fortunately, this process can be automated using computational techniques, which is known as the task of Topic Classification in NLP. Topic Classification (TC) is also referred to as Text Classification (Kowsari et al., 2019) or Topic Detection (Garcia and Berton, 2021) in the literature. This NLP task falls under the broader category of classification problems, with the objective of assigning one or more topic labels on the sentence-, paragraph- or document-level (Jurafsky and Martin, 2009). Hence, the objective of this thesis work is to enhance MarketResponse's tools for providing automatic classification of customer feedback instances into predefined categories, that is, topics.

Analyzing the customer feedback data provided for this thesis presents a unique set of challenges that reflect the characteristics of working with real-world CX data. One major difficulty is the inherent noise present in such data, since customer feedback often contains typos, abbreviations, emoticons and colloquial language. Furthermore, CX data is likely to exhibit significant class imbalance. This means that the distribution of class labels is uneven, with some categories having higher frequency in a dataset than others. This imbalance can influence the learning process of classification algorithms, leading to models that favor the majority classes at the expense of accurately identifying the minority classes (Tanha et al., 2020). Another challenge lies in the scarcity of high-quality annotated data for this specific domain, which is fundamental for training reliable models. Moreover, customer feedback has a unique style and structure compared to other genres of text. Feedback instances can vary in length, ranging from

[1] https://marketresponsegroup.com

concise statements with a few words to complex, lengthy sentences. Additionally, the data provided for this work was translated from Dutch to English, resulting in some information inevitably lost during the machine translation process. Finally, the data is asynchronous, meaning there was no real-time interaction between clients and the company while filling out the survey.

The resulting dataset is imbalanced in the sense that it contains a wide range of topics (11 main topics and 31 subtopics), and the number of instances per topic varies from less than 100 to several thousand. This class imbalance necessitates careful consideration when working with TC of user-generated content. A key focus of this study is therefore developing strategies to address this imbalance, which involves experimenting with different data augmentation and distribution balancing techniques to create a more representative training set. While there is existing research on TC, its applicability to this specific domain is underexplored, thus related studies can only be considered with partial eligibility. This research also faces limitations in terms of time and available resources. Nonetheless, the findings of this thesis are expected to provide valuable insights for the task of TC within the governance domain and its application to CX analysis.

1.2 Task Definition: Multi-Label Topic Classification

Various supervised machine learning techniques have been explored for addressing TC. The commonality they share is that annotated data is used to train algorithms to be able to predict the labels of unseen data instances, using the observations learned during training. The classification model is a learner which takes the observed input x_i and a defined set of output labels $Y = \{y_1, y_2, \ldots, y_M\}$ in the training data, and is able to predict the labels $y \in Y$ for instances in the test data. In our use case, the model can predict which of the existing topic labels are applicable for a given text input. When the model encounters the test instance "I'm very satisfied with the employee's professionalism and the speed of work.", it should be able to return the corresponding topic labels, which are *Knowledge & skills of employee*, *Professionalism*, *Handling* and *Speed of Processing*.

Depending on the number of labels that can be assigned to an input instance, TC can take various forms. We distinguish between binary, multi-class and multi-label classification in supervised machine learning. Multi-label topic classification (MLTC) allows for more than one label to be assigned to a data instance simultaneously. For example, the content of a news article may be about economy, politics and society all at once (Herrera et al., 2016). This work focuses on MLTC with the objective of identifying the main topics and subtopics in customer feedback instances using a predefined set of topic labels. MLTC has great significance for various applications, including Email Classification, Document Organization, Sentiment Analysis and Recommendation Systems (Aggarwal and Zhai, 2012; Wang et al., 2011; De Clercq et al., 2020).

1.3 Research Questions

Upon inspection of related literature and consultation with the internship company, the following research questions have been defined:

Research Question: Which approach yields the best performance for multi-label topic classification of client feedback in the governance domain?

Sub-question: How does the performance of classifiers differ between a one-step (main topic labels and subtopic labels combined) and a two-step (first main topic labels, then subtopic labels are classified) classification approach for multi-label topic classification of client feedback in the governance domain?

Sub-question: What is the impact of data adaptation and data distribution balancing techniques on the performance of classifiers for multi-label topic classification of client feedback in the governance domain?

1.4 Approach

This research compares the performance of classification algorithms for the task of MLTC of client feedback in the governance domain. We will compare these models under different experimental setups, utilizing datasets created through data augmentation and data rebalancing techniques. To address the task, a conventional machine learning approach with Support Vector Machines (Cortes and Vapnik, 1995) will be implemented and a transformer-based pre-trained language model, Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018) will be fine-tuned on the provided training data.

The conventional machine learning approach will be explored with different preprocessing steps, with the goal to identify the most effective steps to optimize model performance. A key challenge lies in the limited and imbalanced nature of the dataset, characterized by a large number of classes (42 topics) with varying representation, meaning that some classes have a large volume of data instances while others are underrepresented, with very few instances. To address this challenge, the experiments first aim to discover whether the classification models benefit from a one-step or a two-step classification approach. In the one-step approach, the model assigns all main topic and subtopic labels to feedback instances in a single step. In the two-step approach, the model assigns main topic labels first, followed by subtopic labels for instances belonging to the identified main topics. Hyper-parameter tuning will be conducted with the training and validation set to find the best model settings for both classifiers and classification approaches. The choice of the optimal approach will be decided based on the experiment results on the test set. Furthermore, we will explore the impact of data manipulation on model performance by creating two additional training datasets:

- **Undersampled Dataset**: This dataset has reduced number of instances for the overrepresented subtopic classes to reduce the imbalanced nature of the data.
- **Oversampled Dataset**: This dataset has increased number of instances for the underrepresented subtopic classes by combining the original training data with synthetic data generated by GPT-4.

By evaluating the models with these experimental setups, the aim is to examine the influence of data adaptation and data balancing techniques on achieving reliable results in the governance domain with imbalanced datasets. Testing the models with the newly created datasets can also help us better understand the optimal amount of

training data required to achieve acceptable performance, and the impact of the data manipulation techniques on model performance for underrepresented classes. Figure 1.1 provides an overview of the experiments carried out in this work. Our findings might also be insightful for future researchers investigating MLTC in the governance domain while facing similar challenges regarding data distribution.

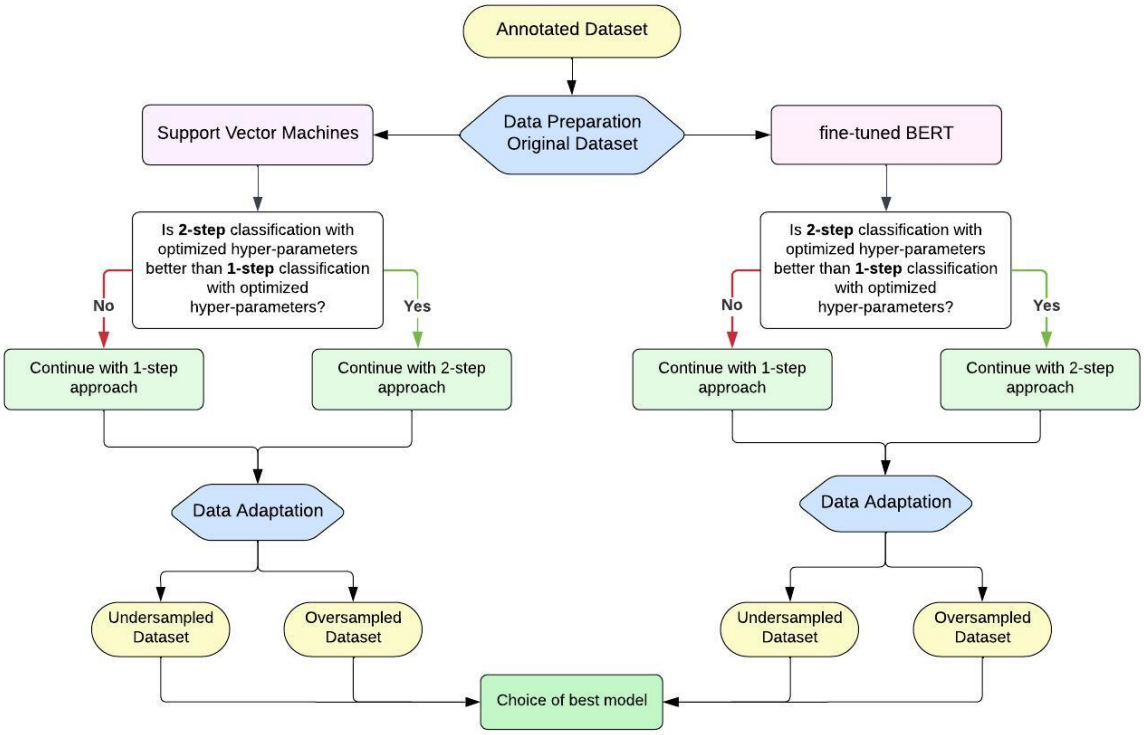

Figure 1.1: Simplified process of multi-label topic classification.

1.5 Thesis Outline

The structure of this thesis is organized as follows. Chapter 2 reviews related research connected to MLTC, focusing on frequently used problem-transformation methods, machine learning and transfer learning approaches, and data balancing techniques. Consequently, Chapter 3 introduces the data, the implemented machine learning and transfer learning approaches and the rationale behind one-step and two-step classification. Chapter 4 presents the results of the experiments with the combination of different models and datasets. It reflects on these findings and provides an error analysis by inspecting the predictions of the best-performing model. Lastly, Chapter 5 provides an overall conclusion for this thesis work, considers the limitations and suggests future directions related to this study.

Chapter 2

Related Work

This chapter provides an overview of the key methodologies and advancements in the field of Topic Classification (TC). Initially, it discusses various approaches utilized in TC, with a particular focus on multi-label topic classification (MLTC), which allows for multiple labels to be assigned to a single data instance. Following this, the chapter introduces specific techniques designed to address class imbalance, a commonly encountered challenge in MLTC. The latter sections of the chapter are dedicated to exploring how different methodologies have been applied within the context of the governance domain.

2.1 Approaches

Topic Classification (TC) has been framed in different ways depending on the number of labels a text instance can have. It can be categorized into three primary types: binary, multi-class and multi-label classification. Binary classification involves categorizing data instances into one of two groups, such as positive or negative. Multi-class classification extends this concept by allowing three or more classes, but each instance is exclusively assigned to only one class. Multi-label topic classification (MLTC) recognizes that instances can belong to multiple classes simultaneously, without exclusive assignment. This is particularly common when handling complex text data, like social media, where a post about a restaurant opening might simultaneously relate to "food," "free time activity" and "local business" topics (Herrera et al., 2016). This section delves deeper into the realm of MLTC, exploring various methodologies used to handle such classification problems, including manually created rule-based systems, feature-based machine learning algorithms and transfer learning methods. Each approach comes with its own advantages and disadvantages, which will be elaborated in the following subsections.

2.1.1 Problem Transformation

Most traditional classification algorithms are only able to process binary or multi-class datasets, meaning they are not inherently applicable to MLTC. Herrera et al. (2016) suggest two strategies to overcome this issue: problem transformation and classifier adaptation. The former modifies the original multi-label dataset into a set of binary or multi-class problems that can be handled by traditional classifiers. The latter implies adapting conventional classification algorithms in a way that they can directly manage

MLTC, enabling them to output multiple labels. This thesis implements the former, primarily due to its versatility and ease of implementation. Problem transformation approaches allow for the utilization of well-established classifiers, which have been widely studied and are likely to be computationally lighter compared to adaptive classifiers. This subsection introduces the prevalent problem transformation approaches, all of which convert the multi-label dataset into binary or multi-class formats compatible with either standard classifiers or classifier ensembles. To complete the classification process, outputs from these classifiers must be aggregated into a comprehensive label set for subsequent model evaluation.

The simplest solution for the multi-class problem is eliminating the multi-label instances from the dataset or selecting a single label for each multi-labeled instance, which can be based on random or heuristic selection (Herrera et al., 2016). Both methods decrease the number of instances and do not offer a real solution to the original problem, therefore, they will not be further discussed. There are several other feasible transformation approaches, the most frequently used ones are described below.

Gonçalves and Quaresma (2003) introduced the **Binary Relevance** technique, which decomposes the multi-label problem into a set of binary problems. On one hand, this method does not change the number of training instances, but it cannot account for the possible correlations between classes. This method can be adjusted using a **chain of classifiers**, where the input for a classifier is composed of the instance features and the output of the previous classifier, accounting for label dependencies (Herrera et al., 2016). Lastly, Boutell et al. (2004) describe a technique often referred to as **Label PowerSet**, which creates a new single label for each multi-label combination that appears in the training data, transforming the task into a multi-class problem for a single classifier. This approach accounts for possible label dependencies because each set of labels composes a new class. As a result, the number of classes significantly increases and certain classes are likely to be underrepresented in the training data. Moreover, multi-label combinations could emerge in the test instances that were not seen by the model in the training data, which can lead to decrease in performance. In recent years, Label PowerSet and Binary Relevance transformations have been prominent solutions for MLTC (Herrera et al., 2016). Table 2.1 summarises the characteristics of the introduced approaches. In this work, the Binary Relevance technique will be used due to its simplicity and common application for multi-label classification (Herrera et al., 2016).

Approach	**Classification type**	**Label Correlation**
Binary Relevance	Binary	Ignored
Classifier Chains	Binary	Considered
Label PowerSet	Multi-class	Considered

Table 2.1: Problem transformation approaches for multi-label topic classification.

2.1.2 Rule-based systems

Previous to the accessibility of machine learning and deep learning algorithms, the majority of studies relied on a selection of hand-crafted rules to address NLP tasks. The prerequisites of establishing a rule-based system include the careful inspection and often the annotation of the dataset. Consequently, word patterns and logical

expressions likely to correspond to certain topic labels are identified using the training data, which are organised into a set of rules. For each test instance, the relevant and possibly overlapping rules are taken into consideration to decide which class the instance belongs to (Aggarwal and Zhai, 2012).

The rule-based approach is often combined with machine learning models or deep learning approaches for text classification, resulting in highly efficient hybrid systems. For instance, Villena Román et al. (2011) applied the k-Nearest-Neighbors algorithm combined with a rule set to filter positive, negative and highly relevant terms for classifying multi-labeled news articles and medical texts. The number of classes in the utilized datasets range from 90 to 1,349, and the results indicate that implementing the rules led to a 7.3% increase in the F1-score, with average precision reaching 0.95. In another study, Li et al. (2021) combined a bi-directional Long Short-term Memory Network (LSTM) with a regular expression-based classifier for multi-class text classification in the medical domain. The system was evaluated using a dataset of user-generated medical queries with 100 categories, reaching 0.89 accuracy and 0.92 F1-score.

Although implementing a rule-based system is computationally efficient and easily interpretable, it requires extensive manual labour and does not lend itself to robustness and generalizability across different domains and languages. The maintenance of rule-based systems also involves the frequent update of the rule set due to new or outdated rules, and in-depth understanding of the domain.

2.1.3 Conventional Machine Learning Approaches

Machine learning methods offer a robust solution for text classification. These methods involve training a model to recognize the associations between text instances and their corresponding labels. This enables the model to predict the labels for new, unseen text data, showcasing the efficiency of machine learning in handling complex NLP tasks. This section introduces the most frequently used feature representation techniques, alongside common classification algorithms for TC.

Feature Extraction and Vectorization

Given that machines can only interpret numerical values, the input text is initially transformed into a structured feature space. After data pre-processing, feature extraction methods are implemented to convert the data into a structured numeric format. Vectorization techniques such as Bag of Words (BoW) and Term Frequency-Inverse Document Frequency (TF-IDF) are commonly utilized for this conversion. These techniques transform each text instance into numerical values that capture its key features relevant to TC (Kowsari et al., 2019).

The simple **Bag of Words (BoW)** feature representation captures the presence or absence of terms in a text, which can help models identify the topic content of a text instance, particularly when combined with other methods like word ***n*-gram** representation (Kowsari et al., 2019). Word n-grams help to reduce the ambiguity in feature representation by capturing sequences of n adjacent words, thus providing a more nuanced understanding of topic-related terms. In their survey, Figueiredo et al. (2011) compare several studies related to text classification using word n-gram features instead of, or in conjunction with BoW, and conclude that the gains tend to be only marginal when incorporating n-grams.

Term frequency (TF), which captures the frequency of terms in a document, can be used to enhance the BoW representation. A more refined alternative, **Term Frequency-Inverse Document Frequency (TF-IDF)**, is commonly implemented for TC as it helps in distinguishing terms that are important in a particular text while filtering out common and less relevant terms (Kowsari et al., 2019). This feature representation was effectively used by Chase et al. (2014) in their study on MLTC of news articles, employing one-versus-all Naive Bayes classifiers with TF-IDF to handle a large dataset of New York Times articles tagged with 9 major topic labels, achieving an average error rate of 13.3%. In this study, they used three feature sets including the full article text, the lead paragraph and the article headline, concluding that the classifiers are fairly accurate even when presented only with the headline.

While neither BoW nor TF-IDF can capture deeper semantic relationships, **word embeddings** address this limitation (Kowsari et al., 2019). Static embedding models, such as Word2Vec (Mikolov et al., 2013) and GloVe (Pennington et al., 2014), represent words as dense vectors in a low-dimensional space and capture the semantic relationships between them. This capability is crucial for enhancing models designed for TC, as it enables a more nuanced understanding of terms and their contextual relevance. Word embeddings will be described in Section 2.1.4.

This section has highlighted how various feature extraction and vectorization techniques play an integral role in TC. Given the effectiveness of TF-IDF for feature representation (Kowsari et al., 2019; Chase et al., 2014; Dadgar et al., 2016), this study will utilize this feature representation technique integrated into a feature-based model – Support Vector Machines (SVMs) –, which will be described in Chapter 3. We will evaluate the model's performance on classifying client feedback instances based on topic categories within the governance domain.

Machine Learning Approaches and Neural Networks

Conventional machine learning classifiers and neural networks have been extensively studied for text classification. This section reviews the most common approaches and highlights studies demonstrating their effectiveness for the task at hand.

Naive Bayes (NB) is a generative probabilistic classifier based on the Bayes' Theorem. This algorithm is referred to as naïve because it assumes complete independence between various feature interactions given the class. The classifier calculates the likelihood that the data point belongs to each class based on prior probabilities, and predicts the one with the highest likelihood (Jurafsky and Martin, 2009). Despite its simplicity, NB has proven powerful in handling text classification problems. For example, Lee et al. (2011) implemented a multinomial NB classifier with TF-IDF feature representation to identify topics in tweets, using a predefined set of 18 classes, achieving an accuracy of 0.56 for the multi-class problem with a small amount of training data. Similarly, Spasic et al. (2012) used a multinomial NB algorithm with pattern-matching rules to classify suicide notes into 15 topics, reaching a 0.53 F1-score.

Logistic Regression is a discriminative classifier that learns which features are most indicative to differenciate between the possible set of classes in the learning phase. Multinomial Logistic Regression, also called softmax regression, can be applied to multi-class classification problems, where the model returns probabilities for the test instance belonging to each class and assigns the label with the highest probability (Kowsari et al., 2019). Shah et al. (2020) conducted a comparative study evaluating Logistic Regression,

Random Forests and k-Nearest-Neighbors with TF-IDF for multi-label classification of news articles into 5 thematic categories, finding that Logistic Regression achieved the best performance with an accuracy of 0.97.

Support Vector Machines (SVMs) are commonly used discriminative algorithms for classification and regression problems. They operate with the objective of finding an optimal hyperplane that maximizes the margin between data instances in feature space. In other words, SVMs are instance-based learners with the aim of adjusting the decision boundary based on the delineations between the classes. Support vectors, which are data instances closest to the decision boundary, are useful to find the optimal separating line in the learning phase (Cortes and Vapnik, 1995). The algorithm can be implemented both for linearly separable and non-linear data by leveraging data transformation into a higher dimensional space with the so-called kernel trick (Herrera et al., 2016). Due to their generalization capabilities and discriminative power, SVMs can be applied in various scenarios, including binary or multi-class classification with imbalanced datasets and large collections of data (Cervantes et al., 2020). SVMs tend to exhibit superior performance for classification tasks when contested with other algorithms (Kowsari et al., 2019; Sen et al., 2020; Aggarwal and Zhai, 2012). As an example, Pranckevičius and Marcinkevičius (2017) contested NB, Random Forest, Decision Tree and Logistic Regression against SVMs for the multi-class classification of Amazon product reviews using 5 classes and BoW with n-gram representation, where SVMs reached a maximum of 0.44 accuracy.

The ***k*-Nearest-Neighbors (*k*NN)** classification algorithm is a popular instance-based method for solving classification and regression problems (Herrera et al., 2016). The intuition behind this algorithm is that data points located close to each other in feature space are likely to belong to the same class. When applied in the field of TC, the algorithm computes the similarity between the new instance and the training instances to classify the instance based on the labels observed in the k nearest instances (Han et al., 2001). For instance, Trstenjak et al. (2014) achieved an accuracy score of 0.92 in classifying news articles into 4 topics using TF-IDF feature represenation and a kNN algorithm. The study highlights the impact of data pre-processing and training data quality on the algorithm's performance for multi-class classification. The results strongly depend on the choice of k value, and its scalability for large datasets may be computationally expensive (Sen et al., 2020).

Tree-based algorithms, such as C4.5 and Random Forest (RF), can also be employed for text classification (Herrera et al., 2016). A decision tree follows a sequence of decisions in a hierarchical structure to split the data into increasingly homogeneous subsets (Aggarwal and Zhai, 2012). Rane and Kumar (2018) compared an improved lightweight RF classifier with Logistic Regression, SVMs, kNN, NB and AdaBoost for classifying airline tweets based on sentiment, finding that RF achieved the highest F1-score of 0.87. Tree-based methods can handle both numerical and categorical data, even with imbalanced datasets. However, they can be susceptible to overfitting, meaning the model performs well on the training data but poorly on unseen data if the trees become too complex or the data is noisy (Kowsari et al., 2019).

In addition to the previously described classifiers, **neural networks** offer powerful capabilities for TC. These networks, inspired by biological neural connections, can understand complex relationships between input text and output labels (Noori, 2021). For example, Jelodar et al. (2020) used a Long Short-term Memory Network (LSTM) for multi-class topic identification in social media posts about COVID-19 using a set

of 25 topics, reaching an accuracy of 0.81. Chen et al. (2017) employed an ensemble of convolutional and recurrent neural networks for MLTC using news stories with nearly 100 topic labels, reaching a 0.71 macro F1-score. Their study highlighted that the efficiency of neural networks strongly depends on the availability of training data, since little amount of data can lead to models that overfit.

While conventional classifiers rely on problem transformation approaches to handle multi-label classification, **adaptive machine learning classifiers** directly accommodate for multi-labeled data. An adaptation of the kNN algorithm, ML-kNN, analyzes the classes among the nearest neighbors of the test instance, and assigns the most probable label set to the test instance (Zhang and Zhou, 2007). Li and Ou (2021) compared ML-kNN against decision trees and traditional kNN for MLTC, using a dataset of research papers from various disciplines, categorized into 6 different topic categories. Their findings demonstrated that ML-kNN achieved superior performance with an average precision of 0.58. Chen et al. (2016) introduced a modified SVMs algorithm called Twin Multi-Label Support Vector Machines (MLTSVMs) for directly addressing multi-label classification problems. This algorithm efficiently finds multiple non-parallel separating hyperplanes to handle multi-label data.

Given the introduced challenges of multi-label classification and the need for robust, scalable and well-documented solutions for classifying client feedback in the governance domain, this study opts for implementing SVMs with Binary Relevance problem transformation. SVMs are chosen for their effectiveness in handling sparse, high-dimensional textual data and their capability to model complex decision boundaries, which are crucial when classifying text into multiple topic categories (Kowsari et al., 2019).

2.1.4 Transfer Learning

Utilizing rule-based systems and conventional machine learning classifiers implies that the systems' performance depends on the selected rules and features during feature engineering. On the contrary, deep learning models leverage low-dimensional dense vector representations instead of relying on handcrafted features (Bogatinovski et al., 2022). However, implementing conventional machine learning methods or neural network models with considerable parameter size comes at a price, which includes the risk of overfitting and poor generalization on new data depending on the size of the available training data, limitations in interpretability due to the "black box" nature of these models, and the potential amplification of bias present in the training data.

Pre-trained embeddings aim to mitigate the explicit need for manual feature engineering and improve generalization capabilities by learning semantic relationships and capturing complex patterns in the training data, which often leads to enhanced performance. However, it is crucial to note that using word embeddings can share some of the challenges mentioned earlier, especially in terms of bias amplification, explainability and cross-domain generalization. We distinguish between first-generation static embeddings, such as the previously mentioned GloVe (Pennington et al., 2014) and Word2Vec (Mikolov et al., 2013), and second-generation contextual word embeddings. In case of static embeddings, each term is represented by a single context-independent vector in a low-dimensional space, which poses a problem since polysemous words, such as *bank, head* and *set*, can have a range of different meanings depending on the context. Second-generation pre-trained language models offer a solution by creating contextualized word representations, in which "word vectors [...] are sensitive to the context in which they appear" (Ethayarajh, 2019, p. 55). These models are trained

on large collections of general domain text in a self-supervised manner, and therefore are able to capture various forms of language representations, including semantic, syntactic and pragmatic information. This approach lends itself to the concept of transfer learning, where knowledge extracted from a pre-trained model can be utilized for downstream tasks, resulting in improved model performance, especially when there is only limited amount of labeled data available for the target task (Hadi and Fard, 2023). For example, Lenc and Král (2017) demonstrated that leveraging word embeddings with Convolutional Neural Networks can be useful for multi-label document classification with large-scale and diverse datasets. More specifically, their study utilizes Czech and English corpora, each comprising nearly 10,000 news articles across 37 and 90 categories, respectively.

In recent years, the emergence of transformer-based language models with advanced architecture resulted in spectacular success thanks to their capability of learning universal language representations from large quantities of text, and transfer this knowledge to various NLP tasks (Kalyan et al., 2021). The key driver of this achievement is the Transformer proposed by Vaswani et al. (2017), a deep-learning model with a stack of encoder and decoder layers, enabling the model to learn complex language representations. The Transformer's core component is its revolutionary attention mechanism, which efficiently captures long-range dependencies between tokens in the input sequence in a parallelized manner. The attention mechanism will be introduced in detail in Chapter 3. Pre-trained transformer models (PTMs) offer a powerful advantage, since they are trained on vast amounts of text and can be fine-tuned for specific NLP tasks in different domains. Leveraging PTMs significantly reduces training time and resource requirements compared to training models from scratch (Hadi and Fard, 2023).

The advantages of transformer-based models can be effectively utilized in TC. Certain auto-encoding models, such as BERT (Devlin et al., 2018), RoBERTa (Liu et al., 2019), DistilBERT (Sanh et al., 2019) and XLNet (Yang et al., 2019) have been previously implemented for this NLP task. Shaheen et al. (2020) experimented with transformer-based models for multi-label classification of documents in the legal domain, reaching a state-of-the-art result of 0.76 micro F1-score by fine-tuning RoBERTa. On the other hand, Yogarajan et al. (2021) highlight in their study on MLTC in the medical domain that traditional neural networks might perform better than transformer-based models for infrequent classes if the dataset is strongly imbalanced, contains long input documents and/or more than 300 classes.

This thesis takes inspiration from the introduced research findings on MLTC. Unlike other studies, the methodology for this work is tailored to the unique challenges of the provided dataset from the governance domain. This study opts for the Binary Relevance method to tackle the inherent multi-label nature of the dataset (Gonçalves and Quaresma, 2003). This decision is grounded in the method's popularity and demonstrated utility in simplifying multi-labeled tasks into a set of binary classification problems (Herrera et al., 2016). We use words in the feedback instances as features and represent them through the TF-IDF weighting scheme due its effectiveness in highlighting term importance across input sequences (Kowsari et al., 2019). SVMs are selected as the primary machine learning classifier for their robustness in managing high-dimensional data (Cervantes et al., 2020), aligning with the characteristics of the dataset for this work. Despite the promising capabilities of adaptive machine learning classifiers, these will not be explored due to time constraints. Additionally, the thesis investigates the power of transfer learning by fine-tuning BERT (Devlin et al., 2018)

for MLTC, with the objective to compare its performance against SVMs. This dual methodology allows us to find the most effective approach for processing multi-labeled client feedback within the governance domain. Due to the scarcity of time and resources, other classification algorithms and transfer learning approaches could be only explored in future work.

2.2 Class Imbalance in Data Distribution

When working with real-world data, researchers often encounter uneven dataset distribution where instances and their corresponding labels classes are non-uniformly distributed across the data space, which can strongly impact the classifiers' learning process. The problem of imbalanced datasets is a frequently discussed challenge in the realm of MLTC (Tahir et al., 2012; Charte et al., 2015; Tarekegn et al., 2021). Additionally, the previously introduced problem transformation approaches, especially Binary Relevance, further exacerbate the levels of imbalance (Herrera et al., 2016). Various strategies have been proposed to handle the challenge of data disparity and enhance the performance of classifiers for the task of MLTC, as described in this section.

These solutions can be categorised into three groups: data resampling, algorithm adaptation and cost-sensitive learning (Herrera et al., 2016). Among these, **data resampling** techniques are independent of the chosen classifier. This approach involves creating a new dataset by undersampling and/or oversampling the original one. Undersampling reduces the number of instances in the training data for the majority classes, aiming to reduce their influence during training. Conversely, oversampling increases the representation of minority classes in the training data by generating new samples. These resampling methods can be implemented either through random or heuristic selection, depending on how instances are removed or added (Tarekegn et al., 2021). Mountassir et al. (2012) propose three distinct undersampling strategies to reduce the size of majority classes in datasets. The first technique, *removing similar* instances, aims to eliminate highly similar data points within the overrepresented classes, since these contribute little to learning new patterns for the classifier. The second method, *remove furthest*, focuses on removing those instances that are most distant from their class centroid, as they are likely to introduce noise and confusion during the classification process. The third strategy, *remove by clustering*, involves applying clustering algorithms to organize the majority class into groups and retaining only those instances that are closest to each cluster's center, thus ensuring an optimal balancing of the dataset.

Algorithm adaptation strategies modify classifiers to directly account for the imbalanced distribution of the dataset during the learning process (Herrera et al., 2016). An example is the approach proposed by Chen et al. (2006), which utilizes a Min-Max Modular network combined with SVMs. This method decomposes the multi-label classification task into a series of binary classification problems, thereby enhancing the classifier's effectiveness in handling data imbalance. Another approach is **cost-sensitive learning**, which employs specific cost metrics that penalize misclassification based on the determined class importance. Unlike traditional models that treat all misclassification equally, cost-sensitive learning assigns greater penalties for errors involving minority class instances. This approach adjusts the weights of instances so they are inversely proportional to the class size, thus placing a higher cost on the misclassification of underrepresented classes, emphasizing their significance in the learning

process (Tarekegn et al., 2021).

In addition to the previously introduced methods, **data augmentation** serves as a valuable strategy to diversify the training data without collecting additional examples. This technique is commonly employed in the field of Computer Vision, for instance by altering the size, shape or color distribution of images to increase the volume of training data and thereby train more resilient models (Maharana et al., 2022). Implementing data augmentation techniques for text data presents unique challenges, as alterations can disrupt the semantic integrity and grammatical structure of text instances. Despite these concerns, data augmentation has proven to be beneficial in the field of NLP. Techniques such as lexical substitution, back-translation, text surface transformation, random noise injection, instance crossover augmentation, syntax-tree manipulation and text element mixing can significantly enhance model performance by diversifying the training data (Chaudhary, 2020). The recent appearance of generative models has introduced novel methods for addressing the data scarcity associated with infrequent classes. Specifically, generative pre-trained transformer models (GPT) can be leveraged to create synthetic data (Yenduri et al., 2023). This approach is particularly effective in increasing the representation of small classes in TC, thus helping to balance datasets and improve model performance (Anaby-Tavor et al., 2019).

To mitigate the issue of class imbalance in the provided dataset, this thesis employed selected data adaptation techniques in order to create additional training sets. Undersampling was applied to reduce the dominance of overrepresented subtopic classes in the dataset, thereby minimizing potential bias towards more common subtopics. Oversampling was used to generate synthetic data for underrepresented subtopic classes using GPT-4 (Achiam et al., 2023) to broaden the models' exposure to less frequent classes. These strategies were implemented to account for the imbalanced nature of the dataset by creating a more balanced representation across various topics. By doing so, the thesis explores the possibility of improved model robustness and generalization capabilities, particularly for subtopics that are underrepresented in the original data.

2.3 Governance Domain

Research explicitly addressing MLTC within the governance domain is scarce, especially studies focusing on user-generated feedback collected through surveys. This research gap highlights a significant opportunity for expanding research in this area. A related study conducted by Mehra et al. (2022) explores the application of a BERT-based model fine-tuned on an Environmental, Social and Governance (ESG) corpus to achieve a high accuracy ESG-related classification tasks. The authors demonstrate that their domain-specific model, pre-trained on ESG-specific text data and further refined through fine-tuning for classification, outperforms the traditional BERT model, reaching a test accuracy of 0.79. Similarly, the research by Nugent et al. (2021) introduces a domain-specific BERT_{RNA} model, a BERT variant pre-trained on a large corpus of finance-, business- and governance-related texts sourced from the Reuters News Archive. This model shows high performance in multi-class and multi-label topic classification tasks, such as ESG controversy detection, when compared to the general BERT model. The implementation of domain-specific pre-training coupled with data augmentation techniques like back-translation has been shown to enhance model performance up to an F1-score of 0.83 when tested on the multi-label UN SDG dataset. The BERT_{RNA} model is not openly accessible, and its resource code is private, which

precludes its application in this thesis. As current knowledge stands, there is no documented study that specifically tackles automatic MLTC of client feedback in the governance domain. This thesis aims to fill this gap and extend the current understanding of TC within the governance domain, hoping to pave the way for future research.

2.4 Concluding Remarks

This chapter has described a variety of strategies for MLTC, ranging from rule-based systems to machine learning techniques and the integration of transfer learning. Additionally, several approaches for handling class imbalance and studies related to the governance domain have been introduced.

In this study, we have opted to employ the Binary Relevance method for its straightforward application in decomposing multi-labeled tasks into simpler binary problems, which has been commonly implemented in various studies (Herrera et al., 2016). SVMs will be compared with a BERT model fine-tuned for the task of MLTC using the provided dataset. The decision to focus on these two models is due to their common application and high performance in TC (Wang and Manning, 2012; Cervantes et al., 2020; Ameer et al., 2023). Other machine learning classifiers and transformer-based models will not be explored due the limitations in time and available computational resources. Further details on the selected machine learning model and the transformer-based model will be explained in Chapter 3. This chapter will include describing the experimental setup, data cleaning steps and hyper-parameter tuning that will be implemented to optimize the models' performance. Moreover, addressing the challenge of imbalanced topic distribution will be key for enhancing the models' performance on the underrepresented subtopic classes, for which two data adaptation methods will be explored. This thesis aims to contribute to the academic understanding of MLTC in the governance domain by experimenting with various methods for categorizing client feedback.

Chapter 3

Methodology

This chapter outlines the methodology implemented for this thesis. It begins with the presentation of the data alongside a data statement, the description of main topics and subtopics, and the utilized data adaptation techniques. It describes the implementation of machine learning and transfer learning techniques for addressing the challenge of multi-label topic classification (MLTC). The chapter also elaborates on the reasons and methods for employing both one-step and two-step classification approaches.

3.1 Data

This study analyzes written client feedback data in English (**ISO 639-1 en**) collected from Dutch governmental institutions using survey forms. These surveys, distributed via email to individuals who visited an office branch, included rating scales and open-ended questions. By analyzing feedback data, governmental institutions aim to gain insights into the offices' operations and identify areas for improving their services.

The collected Dutch text data was annotated using taxonomies with a rule-based system. Following the categorization of feedback instances with Dutch labels, the original data and the labels were machine translated into American English using the Azure AI Translator service[1]. This step was necessary because MarketResponse primarily offers solutions using services that handle English data. To ensure the high quality of translations, a colleague at MarketResponse with bilingual fluency in English and extensive knowledge in NLP reviewed and improved the translations. A significant limitation of this study is that topic labels were not manually reviewed, which will be further elaborated on in Chapter 4 and 5. The length and complexity of the feedback statements vary. In some cases the client feedback statements only contain single words, such as "amazing" or "disappointing", for these mostly the label *No topic found* was assigned. In contrast, some entries can be entire paragraphs where the client elaborates on their experience with the institution, potentially mentioning several topics.

Each main topic has one or more associated subtopics, providing a granular level of detail for analysis, as shown in Figure 3.1. Each data instance has at least one main topic and one subtopic. Some feedback statements with diverse content can have up to three main topics and three subtopics. The dataset consists of 19,529 individual client feedback instances, with the average of 13.6 tokens and 1.36 sentences per text. The complete dataset encompasses a total of 11 main topic labels and 31 subtopic labels.

[1] https://learn.microsoft.com/en-us/azure/ai-services/translator/

The hierarchy among the topics is illustrated in Figure 3.2, while Table 3.1 and 3.2 show the topic distribution in the full dataset.

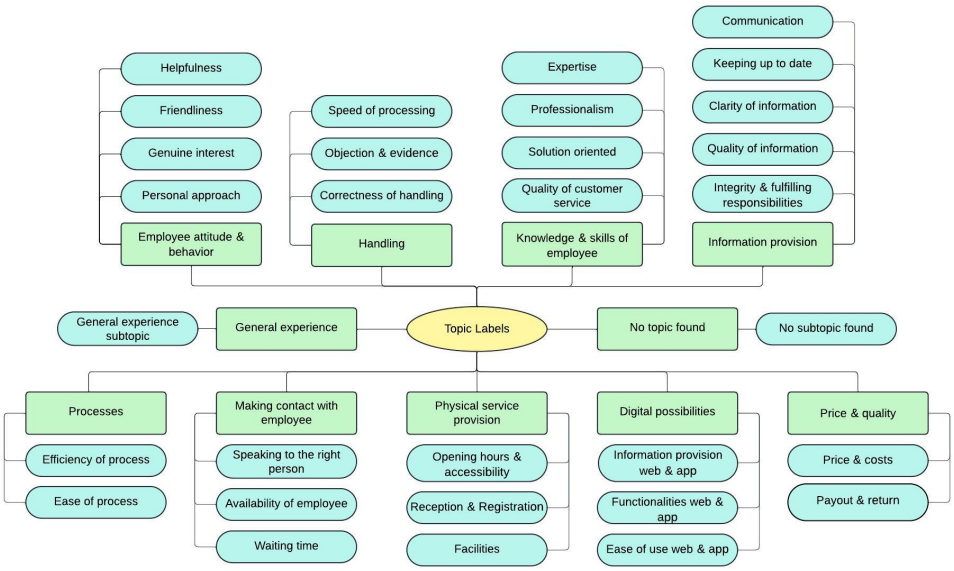

Figure 3.1: Main topic labels (green) and subtopic labels (blue).

Data Statement

Bender and Friedman (2018) introduced data statements in order to mitigate systematic bias and promote transparent scientific experiments by reducing exclusion and bias in NLP. The following parts summarize the characteristics of the dataset, provided by the internship company, that is used for the thesis experiments.

Most of the authors of the feedback statements are likely to be native Dutch speakers or speak Dutch as a second language. Due to the machine translation process from Dutch to American English, some language-specific characteristics may be lost, potentially causing a slight shift in the style of the feedback instances. For instance, the translations may not fully capture language variety, dialects, connotations, denotations, colloquialisms or idiomatic structures. Some feedback statements contain emojis, which were retained in the English version. The vocabulary reflects the domain, with expressions related to the government sector and their provided services. Since clients were asked to answer specific questions related to their experience at the governmental institution, the feedback often revolves around a central topic.

The data collection process was anonymous and voluntary, therefore, there is no information about the writers' demographic background. Since participation in administrative processes at the government offices requires legal age, we assume the survey participants are mostly adults, although the data does not include specific information about age distribution. Similarly, gender and the socioeconomic background of the participants is absent, potentially leading to an imbalanced and unrepresentative sample. Additionally, feedback instances contain privacy-sensitive information like names, addresses and dates, which will be masked with placeholders during data cleaning. The data is strictly confidential and can only be accessed within the company, it should not

be published or distributed elsewhere. The examples used in this work are fabricated illustrations designed solely to reflect the nature of the original data.

ID	Main topic	Number	%
1	Employee attitude & behavior	6,346	25.03
2	Handling	4,786	18.87
3	Information provision	3,082	12.15
4	Knowledge & skills of employee	3,040	11.99
5	No topic found	1,624	6.41
6	Processes	1,570	6.19
7	Making contact with employee	1,510	5.95
8	General experience	1,517	5.98
9	Physical service provision	1,325	5.22
10	Digital possibilities	388	1.53
11	Price & quality	166	0.65
Sum		25,354	100

Table 3.1: Distribution of main topics in the full dataset.

Main Topic Category	ID	Subtopic	Number	%
Employee attitude & behavior	1	Friendliness	4,712	16.73
	2	Helpfulness	1,678	5.96
	3	Genuine interest	839	2.98
	4	Personal approach	195	0.69
Handling	5	Speed of processing	4,666	16.57
	6	Correctness of handling	254	0.90
	7	Objection & evidence	26	0.09
Information provision	8	Clarity of information	1,862	6.61
	9	Quality of information	644	2.29
	10	Communication	597	2.12
	11	Integrity & fulfilling responsibilities	524	1.86
	12	Keeping up to date	148	0.53
Knowledge & skills of employee	13	Solution oriented	1,880	6.68
	14	Expertise	792	2.81
	15	Quality of customer service	479	1.70
	16	Professionalism	457	1.62
No topic found	17	No subtopic found	1,624	5.77
Processes	18	Ease of process	1,046	3.71
	19	Efficiency of process	593	2.11
Making contact with employee	20	Waiting time	1,026	3.64
	21	Availability of employee	328	1.16
	22	Speaking to the right person	230	0.82
General experience	23	General experience subtopic	1,517	5.39
Physical service provision	24	Reception & Registration	887	3.15
	25	Facilities	463	1.64
	26	Opening hours & accessibility	20	0.07
Digital possibilities	27	Ease of use web & app	194	0.69
	28	Functionalities web & app	182	0.65
	29	Information provision web & app	134	0.48
Price & quality	30	Price & costs	161	0.57
	31	Payout & return	5	0.02
Sum			28,163	100

Table 3.2: Distribution of subtopics in the full dataset.

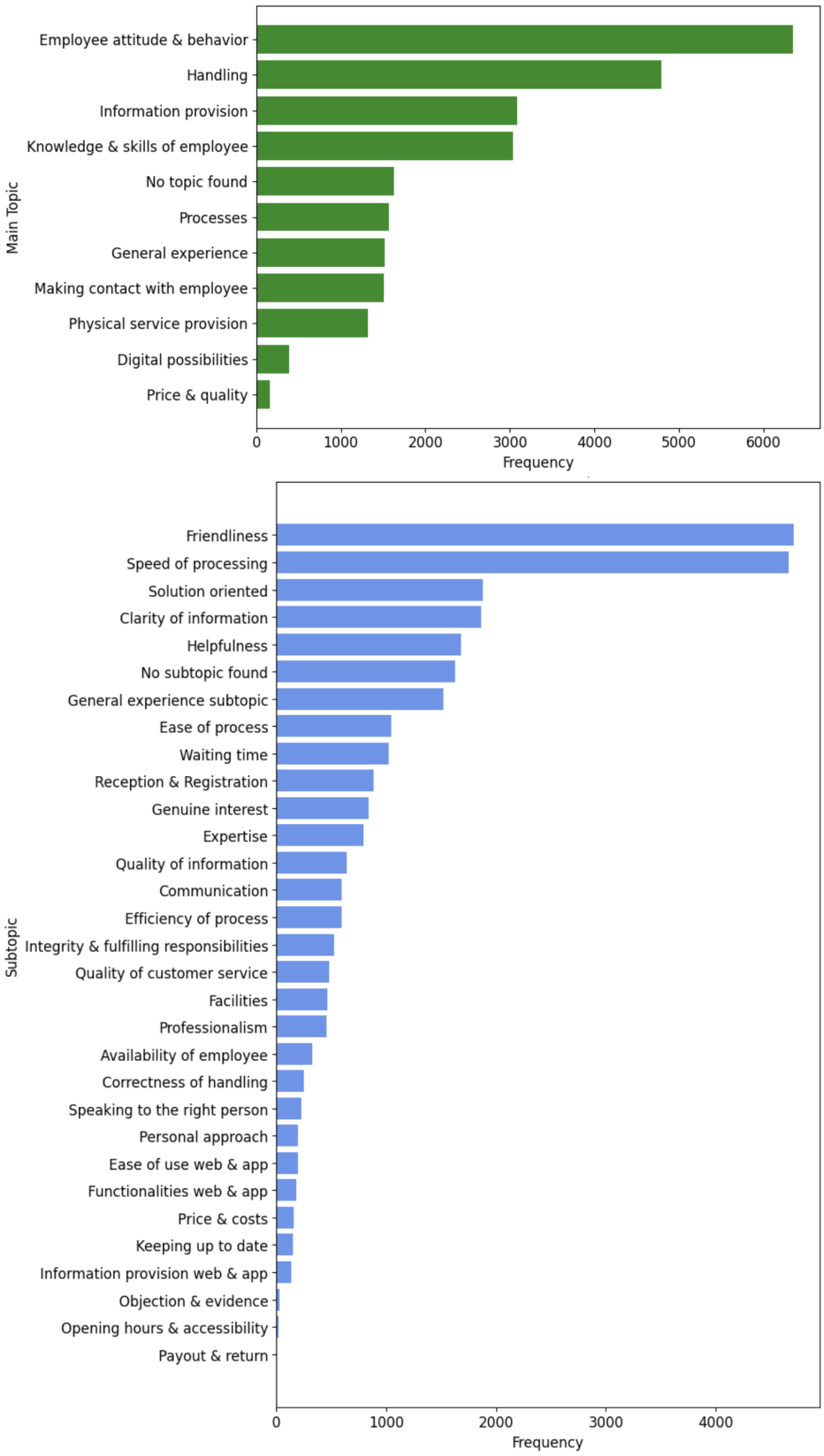

Figure 3.2: Distribution of main topics (top) and subtopics (bottom) in the full dataset.

Content of Main Topics

In order to have an understanding of the similarities and differences between the main topic labels, the following paragraphs aim to summarise the content of the eleven main topics and their corresponding subtopics. The topics are presented in descending order of frequency in the dataset.

Employee attitude & behavior
This is the most prevalent topic in the dataset. Feedback statements belonging to this class describe the clients' experience during their interaction with employees. Specifically, they reveal opinions on employee characteristics like friendliness, helpfulness and personal approach. A positive example feedback instance could be "The employee at the counter was very friendly and patient while answering my questions."

Handling
The topic emerges as the second most frequent theme within the data. This category encompasses client perceptions of the services' efficiency and accuracy in addressing their requests. Feedback labeled with this class often references client satisfaction or dissatisfaction with the processing speed and correctness of inquiry handling. This might include specific examples of both positive experiences, e.g., "The waiting time was quite short." and negative encounters, e.g., "I couldn't get an appointment earlier than 2 weeks!".

Information provision
This is third most frequent class, which delves into client perceptions of how effectively employees communicate and the quality of the information they provide. In other words, feedback of this class encompasses client opinions on the clarity and comprehensiveness of explanations related to their issues. For example, "The passport application process was clearly explained." Additionally, it captures opinions on the accuracy and timeliness of information provided, alongside the perceived integrity, care and responsibility demonstrated by employees during the encounter, such as "Friendly, helpful and supportive staff."

Knowledge & skills of employee
Similar to the previous class, this topic reflects client opinions on employee competence in addressing client concerns. Feedback within this class focuses on employee-related aspects, including professionalism, problem-solving expertise and client-centredness demonstrated during the interactions. It also reveals client satisfaction with the overall quality of the service received. In a feedback statement this might be written like this: "Knowledgeable employee who offered neat solutions to my problem."

No topic found
Approximately 6.4% of the feedback statements received this label, which might be for a variety of reasons. Some instances were simply too short to provide meaningful information, consisting of single words like "flexible" or "staff". Others were overly generic, failing to belong to any established class, such as "All good." or "I'm very satisfied." Moreover, a small portion of the feedback was written in languages other than Dutch or English, which made the categorization impossible. In some cases clients did not wish

to express an opinion in open-ended questions and entered uninformative responses like "don't know", "no opinion".

Processes
This class refers to client experiences with the ease and efficiency of navigating the service. For example, feedback statements in this category elaborate on how clients were treated by the employees, the number of times they were transferred between employees or departments, the quality of phone and email communication, highlighting both difficulties and successes throughout the process. Feedback statements like "The employee at the counter was very friendly and patient while answering my questions." fit into this category.

Making contact with employee
This class captures client experiences regarding the initial contact. Feedback instances of this kind focus on client perspectives on the length of waiting time, the availability of employees, and the efficiency of being directed to the appropriate staff member or department. "The waiting time was a lot shorter than in other branches." – explains the client's satisfaction with the pace of service.

General experience
This main topic label encompasses feedback instances that are informative but do not distinctly belong to any of the other categories. For example, feedback statements might refer to the atmosphere at the institution, the personality of the employee, or express general satisfaction with the service, as in "Well organised operations."

Physical service provision
Additionally, this main category encompasses feedback instances concerning the facilities, the opening hours and accessibility of the buildings, and client experiences with the reception and registration processes. Examples in this category include "The parking area is clean and spacious," "Free coffee and tea is available," and "Upon entering, I was welcomed and assisted immediately."

Digital possibilities
Feedback statements in this class reflect on different aspects of the website and online application of the governmental institution, including the ease of use, functionality and the information available online. Some examples are "The login site doesn't work properly!" and "Making a digital appointment was easy."

Price & quality
Lastly, this is the class with the smallest size, which mainly contains feedback instances concerning the cost of client services, such as "Renewing my passport was very expensive, the price should be decreased." However, there are several instances that are connected to waiting time and arranging appointments, showing inconsistency in the annotations.

This thesis focuses on multi-label topic classification (MLTC) because some feedback statements in the dataset are assigned to multiple main topics and subtopics. Table 3.3 shows that the majority of instances only have two labels, namely one main topic and

one subtopic label. However, a significant portion of instances belong to multiple main- and subtopics, reflecting the multi-labeled nature of the data. The average number of topic labels is 2.74 per feedback instance. The minimum number of labels per instance is 2 and the maximum number of labels is 6 per instance in the dataset.

Number of Labels	Number of Instances	%
2	12,746	65.26
3	1,476	7.56
4	3,593	18.40
5	1,059	5.43
6	655	3.35
Sum	19,529	100

Table 3.3: Label distribution across feedback instances in the full dataset.

3.1.1 Stratified Data Splitting

For this study, the dataset was divided using an 80-10-10 split. Stratified data splitting was implemented to preserve the distribution of data instances within each main topic and subtopic across the different subsets. Consequently, 80% of the dataset comprises the training set, and the remaining 20% is evenly split into validation and test sets. As a result, there are 15,621 instances with 42,824 labels in the training set, 1,954 instances with 5,344 labels in the validation set, and 1,954 instances with 5,349 labels in the test set. The data was split using the iterative-stratification package (Sechidis et al., 2011) and the scikit-learn package (Pedregosa et al., 2011). The overview of the distribution of main topics and subtopics after the splitting can be observed in Table 3.4 and 3.5, as well as in Figure 3.3.

3.1.2 Data Adaptation

Given the substantial class imbalance within the dataset, two data adaptation techniques were employed to adjust the class representation in the training set. We implemented these methods to determine their efficacy for this multi-label classification task, and to assess their impact on the classifiers' performance. Making decisions about these methods involved discussions with representatives from MarketResponse, aiming to align the approaches with their tools and the preferences of the client governmental institution. Although merging certain main topics and subtopics could have been a feasible strategy, it was advised against due to the company's reliance on the existing rule-based tool designed for the 42 topic labels. We have implemented two data sampling techniques: the first focused on undersampling (Kubát and Matwin, 1997) the original training set to reduce the number of instances for overrepresented subtopic classes; while the second centred around oversampling (Chawla et al., 2002) the original training set, utilizing a generative large language model – GPT-4 (Achiam et al., 2023) – to create synthetic data for the underrepresented subtopic classes. These techniques are illustrated in Figure 3.4.

ID	Main topic	Train	%	Valid	%	Test	%
1	Employee attitude & behavior	5,077	25.03	652	25.70	617	24.39
2	Handling	3,829	18.87	480	18.92	477	18.85
3	Information provision	2,466	12.16	303	11.94	313	12.37
4	Knowledge & skills of employee	2,437	12.01	319	12.57	284	11.23
5	No topic found	1,299	6.40	152	5.99	173	6.84
6	Processes	1,256	6.19	149	5.87	165	6.52
7	Making contact with employee	1,208	5.95	158	6.23	144	5.69
8	General experience	1,214	5.98	138	5.44	165	6.52
9	Physical service provision	1,058	5.22	125	4.93	142	5.61
10	Digital possibilities	310	1.53	43	1.69	35	1.38
11	Price & quality	133	0.66	18	0.71	15	0.59
	Sum	20,287	100	2,537	100	2,530	100

Table 3.4: Distribution of main topics in training, validation and test set.

Main Topic Category	ID	Subtopic	Train	%	Val.	%	Test	%
Employee attitude & behavior	1	Friendliness	3,773	16.74	485	17.28	454	16.11
	2	Helpfulness	1,342	5.96	164	5.84	172	6.10
	3	Genuine interest	671	2.98	87	3.10	81	2.87
	4	Personal approach	156	0.69	16	0.57	23	0.82
Handling	5	Speed of processing	3,730	16.55	469	16.71	467	16.57
	6	Correctness of handling	203	0.90	29	1.03	22	0.78
	7	Objection & evidence	21	0.09	1	0.04	4	0.14
Information provision	8	Clarity of information	1,496	6.64	169	6.02	197	6.99
	9	Quality of information	515	2.29	62	2.21	67	2.38
	10	Communication	478	2.12	66	2.35	53	1.88
	11	Integrity & fulfilling responsibilities	419	1.86	51	1.82	54	1.92
	12	Keeping up to date	118	0.52	15	0.53	15	0.53
Knowledge & skills of employee	13	Solution oriented	1,504	6.67	195	6.95	181	6.42
	14	Expertise	634	2.81	83	2.96	75	2.66
	15	Quality of customer service	383	1.70	49	1.75	47	1.67
	16	Professionalism	366	1.62	55	1.96	36	1.28
No topic found	17	No subtopic found	1,299	5.76	152	5.42	173	6.14
Processes	18	Ease of process	837	3.71	103	3.67	106	3.76
	19	Efficiency of process	474	2.10	53	1.89	66	2.34
Making contact with employee	20	Waiting time	820	3.64	107	3.81	99	3.51
	21	Availability of employee	262	1.16	35	1.25	31	1.10
	22	Speaking to the right person	184	0.82	24	0.86	22	0.78
General experience	23	General experience subtopic	1,214	5.39	138	4.92	165	5.85
Physical service provision	24	Reception & Registration	710	3.15	80	2.85	97	3.44
	25	Facilities	370	1.64	44	1.57	49	1.74
	26	Opening hours & accessibility	16	0.07	3	0.11	1	0.04
Digital possibilities	27	Ease of use web & app	155	0.69	21	0.75	18	0.64
	28	Functionalities web & app	147	0.65	18	0.64	17	0.60
	29	Information provision web & app	107	0.47	15	0.53	12	0.43
Price & quality	30	Price & costs	129	0.57	18	0.64	14	0.50
	31	Payout & return	4	0.02	0	0.00	1	0.04
Sum			22,537	100	2,807	100	2,819	100

Table 3.5: Distribution of subtopics in training, validation and test set.

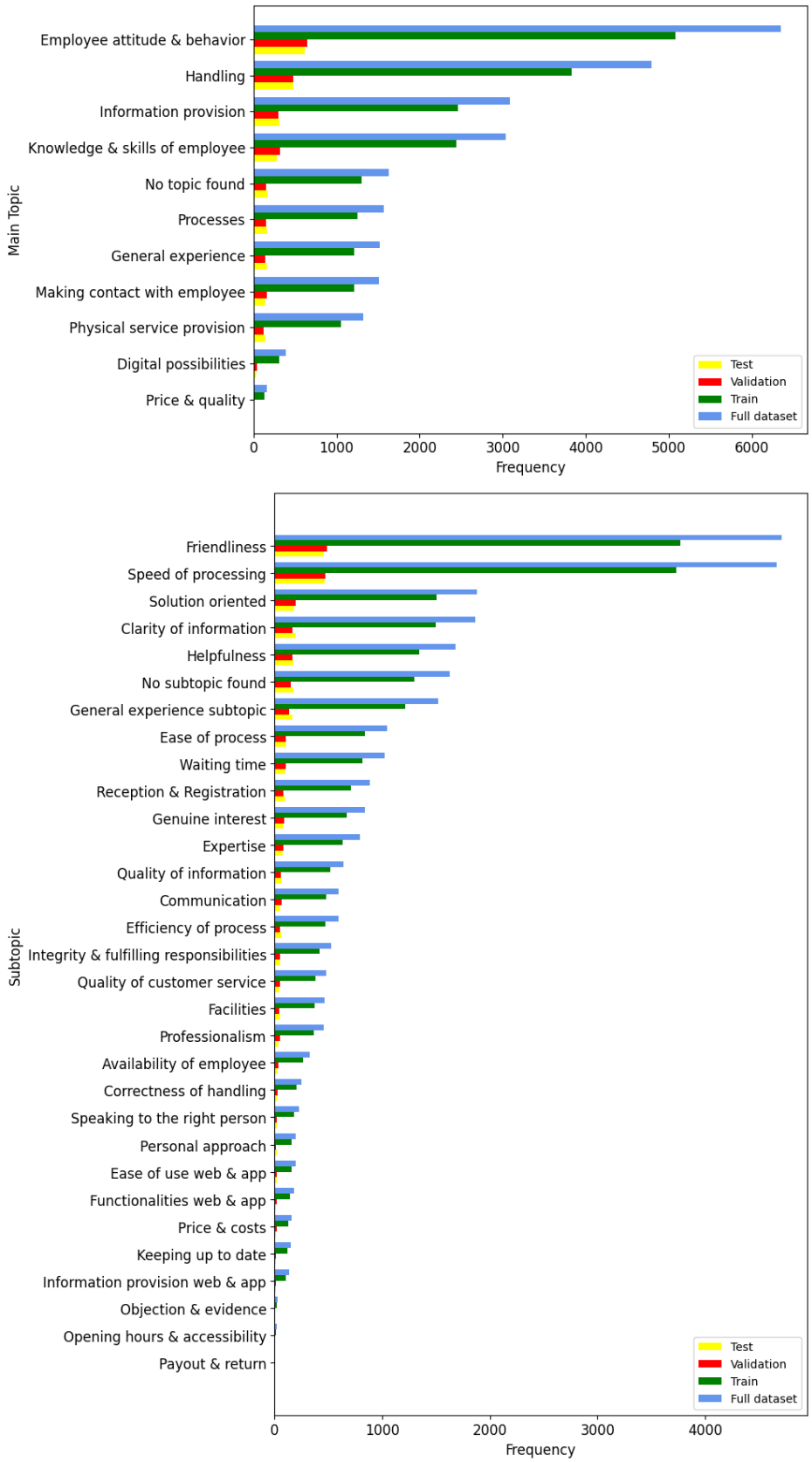

Figure 3.3: Distribution of main topics (top) and subtopics (bottom) in training, validation and test set.

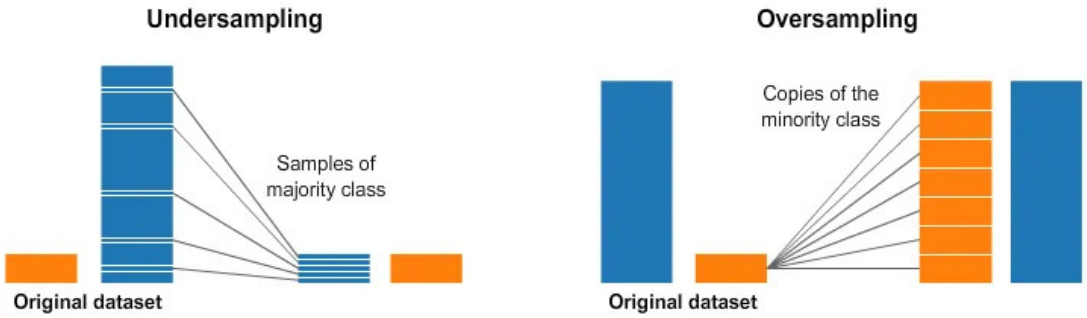

Figure 3.4: Under- and oversampling (Al-Serw, 2021).

Undersampling

Undersampling involves removing data instances from the majority class to balance class distribution. In this work, undersampling targets subtopic classes in the training set that exceed the average representation of instances per class, i.e. more than 727 instances. The identified overrepresented subtopic classes for reduction are *Friendliness*, *Helpfulness*, *Speed of Processing*, *Clarity of information*, *Solution oriented*, *No subtopic found*, *Ease of process*, *Waiting time* and *General experience subtopic*. In order to decrease the imbalanced nature of the dataset, instances from the listed overrepresented classes are randomly removed until they align with the average distribution. A random seed was set to ensure reproducibility of this process. In the first step, the implemented method prioritizes the removal of instances with a single subtopic. However, due to the multi-labeled nature of the data, it is often necessary to also remove instances that contain a combination of subtopic labels. Therefore, in the second step instances containing subtopics other than the target subtopic are removed, while ensuring that instances with underrepresented subtopic labels are protected. The process is iteratively applied until the average distribution is reached for the overrepresented subtopics.

The rationale behind undersampling was to evaluate classifier performance under more balanced class distributions. While the aggregated evaluation scores are not necessarily expected to improve, the results can be insightful for understanding how class representation affects model performance for specific topic categories. A significant disadvantage of this approach is the potential loss of valuable training data, as reducing the representation of majority classes involves discarding informative data instances. The class distribution in the training set after undersampling is shown in Table 3.6 and 3.7.

Oversampling: Synthetic Data Generation

Some subtopic classes are substantially underrepresented due to the imbalanced nature of the original dataset. The limited exposure to these classes can negatively impact model performance on these subtopics. Acquiring additional real-world client feedback is both time-consuming and expensive, therefore generating synthetic data serves as a viable alternative. Synthetic data is artificially created data that closely mimics real-world data, serving as an efficient substitute (Jaderberg et al., 2014). Augmenting the original data with synthetic data allows us to enhance the representation of underrepresented classes, which can potentially increase model robustness and reach better performance for topics with limited number of training data.

ID	Main topic	Train	%	Train U	%
1	Employee attitude & behavior	5,077	25.03	2,064	17.90
2	Handling	3,829	18.87	1,065	9.24
3	Information provision	2,466	12.16	1,697	14.72
4	Knowledge & skills of employee	2,437	12.01	1,660	14.40
5	No topic found	1,299	6.40	727	6.30
6	Processes	1,256	6.19	1,091	9.46
7	Making contact with employee	1,208	5.95	999	8.66
8	General experience	1,214	5.98	727	6.30
9	Physical service provision	1,058	5.22	1,058	9.18
10	Digital possibilities	310	1.53	310	2.69
11	Price & quality	133	0.66	133	1.15
	Sum	20,287	100	11,531	100

Table 3.6: Overview of main topics after undersampling the training data.

Main Topic Category	ID	Subtopic	Train	%	Train U	%
Employee attitude & behavior	1	Friendliness	3,773	16.74	890	6.69
	2	Helpfulness	1,342	5.96	727	5.47
	3	Genuine interest	671	2.98	671	5.05
	4	Personal approach	156	0.69	156	1.17
Handling	5	Speed of processing	3,730	16.55	966	7.27
	6	Correctness of handling	203	0.90	203	1.53
	7	Objection & evidence	21	0.09	21	0.16
Information provision	8	Clarity of information	1,496	6.64	727	5.47
	9	Quality of information	515	2.29	515	3.87
	10	Communication	478	2.12	478	3.60
	11	Integrity & fulfilling responsibilities	419	1.86	419	3.15
	12	Keeping up to date	118	0.52	118	0.89
Knowledge & skills of employee	13	Solution oriented	1,504	6.67	727	5.47
	14	Expertise	634	2.81	634	4.77
	15	Quality of customer service	383	1.70	383	2.88
	16	Professionalism	366	1.62	366	2.75
No topic found	17	No subtopic found	1,299	5.76	727	5.47
Processes	18	Ease of process	837	3.71	672	5.05
	19	Efficiency of process	474	2.10	474	3.56
Making contact with employee	20	Waiting time	820	3.64	611	4.60
	21	Availability of employee	262	1.16	262	1.97
	22	Speaking to the right person	184	0.82	184	1.38
General experience	23	General experience subtopic	1,214	5.39	727	5.47
Physical service provision	24	Reception & Registration	710	3.15	710	5.34
	25	Facilities	370	1.64	370	2.78
	26	Opening hours & accessibility	16	0.07	16	0.11
Digital possibilities	27	Ease of use web & app	155	0.69	155	1.17
	28	Functionalities web & app	147	0.65	147	1.11
	29	Information provision web & app	107	0.47	107	0.80
Price & quality	30	Price & costs	129	0.57	129	0.97
	31	Payout & return	4	0.02	4	0.03
Sum			22,537	100	13,296	100

Table 3.7: Overview of subtopics after undersampling the training data.

Large language models (LLMs) have been frequently used to generate synthetic data for NLP tasks, particularly in contexts where data resources are scarce (Kumar et al., 2020; Ye et al., 2022; Yoo et al., 2021). Data augmentation through synthetic data generated by LLMs has been shown to improve model performance for text classification tasks, like topic classification and humor detection (Li et al., 2023). We distinguish the *zero-shot* technique, where the model generates text or images with the desired labels without receiving any real-world examples, and the *few-shot* technique, which involves providing some examples to guide the model during the data generation process (Li et al., 2023). OpenAI's Generative Pretrained Transformer (GPT) models, including GPT-4, are autoregressive models trained to predict the next token in a sequence (Achiam et al., 2023). Transformer-based models will be described in more detail in Section 3.2.3. The evolution of these models has been notable, with GPT-4 capable of solving complex tasks and producing text that is almost indistinguishable from human-generated content thanks to the model's increased accuracy (Li et al., 2023). While ChatGPT-3.5 is freely available, ChatGPT-4 can currently only be accessed by the public for a monthly subscription fee. Utilizing GPT-4 for synthetic data generation offers several advantages, such as decreasing the time and costs associated with traditional data collection methods, minimizing human effort and obtaining high-quality data. Since GPT-4 can be directed with carefully designed prompts, privacy concerns are mitigated as the output can be generated without including any personal information. However, there are also some concerns of using LLMs, including the potential environmental harm due to the computational demands of prompting these models, biases inherent in the models and the possibility of generating repetitive output.

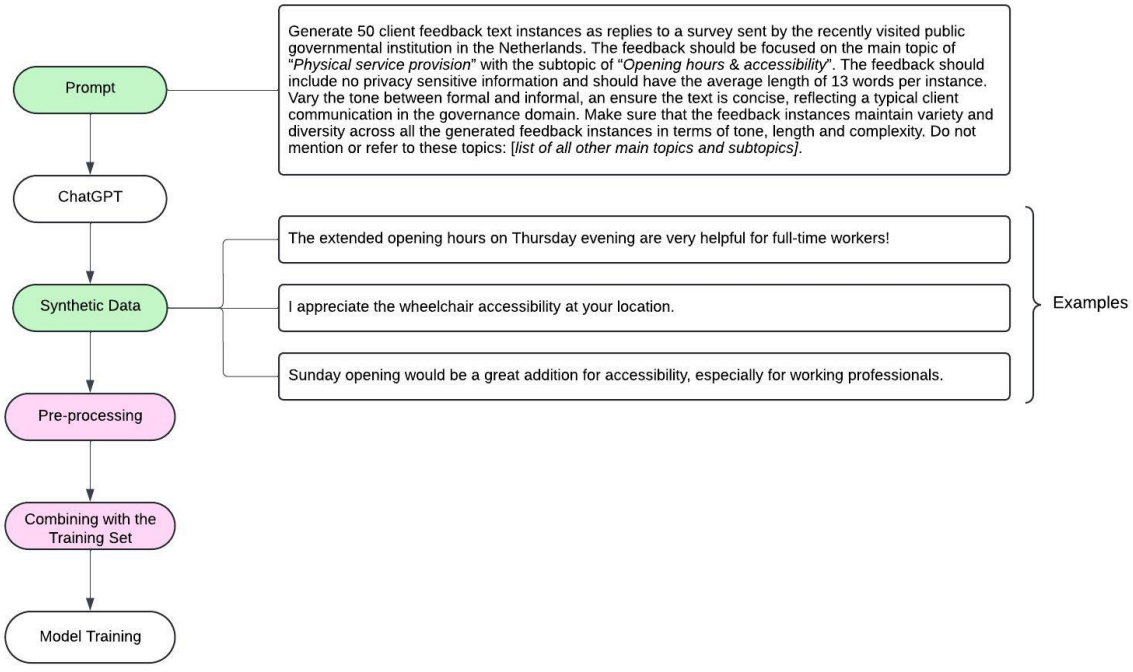

Figure 3.5: Synthetic data generation process.

As previously mentioned, prompting is a key factor in receiving high-quality output from the GPT model. Upon consultations with MarketResponse, we opted for the zero-shot approach due to restrictions on using the existing dataset outside of the company's internal system. The prompt structure of the model is composed of three parts: (1) a context prompt to encourage the model to produce output resembling the target domain, (2) a data generation prompt specifying the style, the corresponding label and the word limit of the text output, and (3) a diversity prompt to ensure the model maintains variety and avoids repetition in the synthetic data output (Li et al., 2023). The utilized prompt and the process of synthetic data generation can be observed in Figure 3.5. Before prompting the model, we identified the underrepresented subtopic labels below the average distribution: *Speaking to the right person*, *Correctness of Handling*, *Functionalities web & app*, *Reception & Registration*, *Quality of information*, *Information provision web & app*, *Availability of employee*, *Price & costs*, *Professionalism*, *Opening hours & accessibility*, *Ease of use web & app*, *Keeping up to date*, *Integrity & fulfilling responsibilities*, *Payout & return*, *Quality of customer service*, *Facilities*, *Objection & evidence*, *Efficiency of process*, *Genuine interest*, *Expertise*, *Personal approach* and *Communication*.

The online interface of GPT-4[2] was used to generate 50 additional feedback entries for each underrepresented subtopic with a hand-crafted prompt and the zero-shot technique. We limited the generation to 50 instances because prompting the model to create more entries led to repetitive outputs and introduced irrelevant subtopics in the text. Since the fourth generation of ChatGPT has long-term memory, i.e. it remembers previous elements of the conversation, we initialized a new dialogue for each iteration. In case of machine learning experiments the synthetic data underwent pre-processing, which will be described in 3.2.2, before it was combined with the original training data and used as input for model training. As a result of oversampling, the number of training examples increased from 15,621 to 16,721, while the validation and test sets were left intact. The detailed overview of the training data after oversampling can be seen in Table 3.8 and 3.9.

ID	Main topic	Train	%	Train O	%
1	Employee attitude & behavior	5,077	25.03	5,177	24.21
2	Handling	3,829	18.87	3,929	18.37
3	Information provision	2,466	12.16	2,666	12.47
4	Knowledge & skills of employee	2,437	12.01	2,587	12.10
5	No topic found	1,299	6.40	1,299	6.07
6	Processes	1,256	6.19	1,306	6.11
7	Making contact with employee	1,208	5.95	1,308	6.12
8	General experience	1,214	5.98	1,214	5.68
9	Physical service provision	1,058	5.22	1,208	5.65
10	Digital possibilities	310	1.53	460	2.15
11	Price & quality	133	0.66	233	1.09
	Sum	20,287	100	21,387	100

Table 3.8: Overview of main topics after oversampling the training data.

[2] https://chat.openai.com/

Main Topic Category	ID	Subtopic	Train	%	Train O	%
Employee attitude & behavior	1	Friendliness	3,773	16.74	3,773	15.96
	2	Helpfulness	1,342	5.96	1,342	5.68
	3	Genuine interest	671	2.98	721	3.05
	4	Personal approach	156	0.69	206	0.87
Handling	5	Speed of processing	3,730	16.55	3,730	15.78
	6	Correctness of handling	203	0.90	253	1.07
	7	Objection & evidence	21	0.09	71	0.30
Information provision	8	Clarity of information	1,496	6.64	1,496	6.33
	9	Quality of information	515	2.29	565	2.39
	10	Communication	478	2.12	528	2.23
	11	Integrity & fulfilling responsibilities	419	1.86	469	1.98
	12	Keeping up to date	118	0.52	168	0.71
Knowledge & skills of employee	13	Solution oriented	1,504	6.67	1,504	6.36
	14	Expertise	634	2.81	684	2.89
	15	Quality of customer service	383	1.70	433	1.83
	16	Professionalism	366	1.62	416	1.76
No topic found	17	No subtopic found	1,299	5.76	1,299	5.50
Processes	18	Ease of process	837	3.71	837	3.54
	19	Efficiency of process	474	2.10	524	2.22
Making contact with employee	20	Waiting time	820	3.64	820	3.47
	21	Availability of employee	262	1.16	312	1.32
	22	Speaking to the right person	184	0.82	234	0.99
General experience	23	General experience subtopic	1,214	5.39	1,214	5.14
Physical service provision	24	Reception & Registration	710	3.15	760	3.22
	25	Facilities	370	1.64	420	1.78
	26	Opening hours & accessibility	16	0.07	66	0.28
Digital possibilities	27	Ease of use web & app	155	0.69	205	0.87
	28	Functionalities web & app	147	0.65	197	0.83
	29	Information provision web & app	107	0.47	157	0.66
Price & quality	30	Price & costs	129	0.57	179	0.76
	31	Payout & return	4	0.02	54	0.23
Sum			22,537	100	23,637	100

Table 3.9: Overview of subtopics after oversampling the training data.

3.2 Multi-Label Topic Classification

The goal of this study is to develop models capable of identifying the topic labels of client feedback instances by comparing them against a predefined list of topics used during training. This study aims to evaluate different classification techniques, assess their effectiveness on an imbalanced dataset and explore various data manipulation strategies to balance this dataset. For the illustrated overview of our experiments, please refer back to Figure 1.1 in Chapter 1. The subsequent sections will detail the methodologies employed in both the machine learning approach using Support Vector Machines (SVMs) (Cortes and Vapnik, 1995) and the transfer learning approach using a fine-tuned Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018).

3.2.1 Data Cleaning

To ensure the data is free from noise and privacy-sensitive information, several data cleaning steps were necessary. This process involved data anonymization and noise filtering. In order to account for privacy sensitive information, names, locations, dates and times, email-addresses and URLs were masked in the data. For named entities removal, focusing on people and locations, SpaCy's Named Entity Recogniser[3] with

[3] https://spacy.io/api/entityrecognizer

a pre-trained English pipeline[4] was applied. The detected names and locations were replaced with the placeholders 'PERSON' and 'LOCATION'. Regular expressions were used to find and replace email-addresses, URLs, dates and times with corresponding placeholder tokens. Masking personal information is essential not only for the purpose of protecting the identity of affected stakeholders, but also for creating more generalizable models. As a result of this step, models do not learn irrelevant connections between frequently occurring named entitites and topic labels in the data. These steps were implemented across the entire dataset to prepare it for both models' input.

3.2.2 Conventional Machine Learning

In this study, we utilized Support Vector Machines (SVMs), a robust supervised machine learning algorithm. SVMs are effective linear classifiers capable of addressing regression, classification, and outlier detection problems (Cortes and Vapnik, 1995). They are particularly suitable for high-dimensional spaces and have been successfully applied to multi-label text classification tasks (de Carvalho and Freitas, 2009).

Before model training, the data must be prepared and transformed into a suitable format for the machine learning algorithm. Common vectorization techniques include TF-IDF, which converts text data into numerical vectors. These vectors capture the relationships between words and their corresponding topic labels, allowing SVMs to accurately predict topics for new text instances. The following sections will describe the algorithm and the specific pre-processing and vectorization techniques used in this study.

Support Vector Machines

SVMs operate by finding a separating hyperplane that serves as a decision boundary dividing data instances into distinct classes within a high-dimensional feature space, as shown in Figure 3.6. The optimal hyperplane is the one that divides the data into distinct classes with the maximum margin, where the margin can be defined as the distance between the nearest data points of each class – known as support vectors – and the decision boundary itself. SVMs aim to maximize this margin while minimizing the classification errors, thereby enhancing the model's ability to predict the correct labels of new, unseen data instances. Given that real-world data is often not perfectly separable, the concept of a soft margin improves the model's flexibility and leads to better generalization. This approach allows some data points to violate the margin constrains and fall on the incorrect side of the hyperplane, which is regulated by parameter C. In scenarios where data points are not linearly separable in the original feature space, the "kernel trick" offers a solution. By applying a kernel function, SVMs can operate in a higher-dimensional feature space, implicitly mapping data points and solving cases where the data cannot be linearly separated. Common kernels include the linear, polynomial, radial basis function (RBF) and sigmoid kernels. The choice of kernel changes the shape of the decision boundary and can significantly influence the performance of the classifier (Gunn et al., 1998).

For this thesis, the linear Support Vector Machine[5] is implemented due to its simple application for binary text classification tasks (Godbole and Sarawagi, 2004). The

[4] https://spacy.io/models/en

[5] https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html

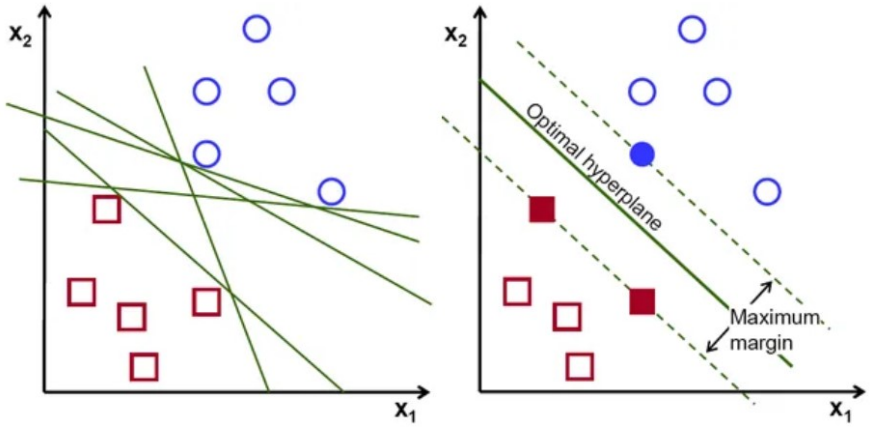

Figure 3.6: Hyperplanes in Support Vector Machines (Gandhi, 2018).

random state was set to 0 for reproducibility. SVMs are particularly suitable for text categorization since they excel in high-dimensional spaces, which is a characteristic of text data. They effectively handle irrelevant features, sparse vector representations and linearly separable classes, which are also typical in text processing. However, SVMs may struggle with very large datasets due to computational demands, and have the tendency to underperform when there is a significant overlap among classes (Joachims, 1998).

Data Pre-processing

Common techniques for preparing text data for TC include tokenization, stop word removal, lemmatization (Kowsari et al., 2019), part-of-speech tagging, spelling correction (Spasic et al., 2012), case transformation, stemming and text normalization (Noori, 2021). Determining which pre-processing steps to implement depends on the characteristics of the task and dataset at hand.

These steps involved applying several text normalization techniques and testing their impact using the validation set. The first step was the **tokenization** of text data, which refers to the division of text into individual words. This step is required for the subsequent vectorization of words, which makes the text data interpretable for machines. The following pre-processing steps were carried out on the tokenized English data. All tokens were converted to their **lowercase** form to ensure consistency and reduce the impact of capitalization variations. English **stop words**, which are high-frequency words with little semantic meaning (e.g., "the", "a"), were removed using a stop list from the NLTK library[6]. This step was extended using a list of **frequent function words** that are not related to specific topics. This list was defined upon careful inspection of the dataset, and consist of 16 words that either contain undefined characters, are contracted verb forms (e.g., "'ll", "'m") or titles (e.g., "Mr.", "Mrs."). Moreover, **punctuation marks** and **digits** were removed for noise reduction. Since nouns and verbs can take a variety of inflectional and derivational suffixes, the base form (lemma) of words was preserved in the data. This process – known as **lemmatization** – involves converting words to their dictionary entry form (Jurafsky and Martin, 2009). As an optional step, code for **spelling correction** was devel-

[6] https://www.nltk.org/

oped but not implemented in this study. The rationale behind spelling correction is to improve model generalizability by handling typographical and grammatical errors. However, considering the data had undergone machine translation and manual correction in previous stages, it contains only a few spelling mistakes, therefore implementing this step was unnecessary for our dataset.

Pre-processing Steps					macro-averaged		
Lowercase	Stop words	Punctuation	Digits	Lemmatize	*precision*	*recall*	*F1*
					0.76	0.46	0.55
✓					0.75	0.46	0.55
✓	✓				0.74	**0.47**	**0.56**
✓	✓	✓			0.74	**0.47**	**0.56**
✓	✓	✓	✓		0.74	**0.47**	**0.56**
✓	✓	✓	✓	✓	**0.76**	0.46	0.55

Table 3.10: Impact of pre-processing steps on model performance with SVMs.

After privacy-sensitive information was masked, an ablation study was conducted using the validation set to empirically determine the most effective pre-processing techniques for the study. During this process, we focused on the incremental addition of steps and monitoring changes in the macro precision, recall and macro F1 scores using a simple linear SVMs classifier with the one-step approach. The observed evaluation metrics – precision, recall and macro F1-score – will be described in Chapter 4. As shown in Table 3.10, most techniques only slightly changed or did not influence the classifier's performance. Interestingly, lemmatization using a SpaCy pipeline adversely affected the macro F1-score, leading to the decision to exclude this step from pre-processing. Finally, we only implemented lowercasing and stop words removal on the dataset for SVMs input in order to maximize performance while minimizing the number of implemented pre-processing steps.

Feature Representation with Bag of Words and TF-IDF

The Bag of Word (BoW) method is commonly utilized for converting text into a numerical format. Introduced in 1954, BoW is constructed as a collection of words that appear in a document, where the order of words is not taken into consideration (Harris, 1954). Input sequences are transformed into vectors, with each vector indicating the presence or absence of each vocabulary element in the text. While BoW offers a straightforward implementation, its major limitation is the loss of grammatical and syntactic context, as it does not consider word order. This approach also fails to capture the nuances in semantics that may differ depending on the context of the word, which can be crucial for tasks like MLTC.

Developed in 1972, the Term Frequency-Inverse Document Frequency (TF-IDF) method modifies the BoW approach by also evaluating the relative importance of each word based on its frequency across the entire text corpus (Sparck Jones, 1972). This technique consists of two components: TF indicates how frequently a word appears within a document, while IDF shows the frequency of a word across all documents in the dataset, thereby offering a measure of how significant a word is.

The formula for calculating TF-IDF is as follows, where tf_{ij} is the term frequency of the i^{th} word in the j^{th} document or text input, and idf_i is the inverse document

frequency of the i^{th} word.

$$w_{ij} = tf_{ij} \times idf_i$$

The outcomes of TF-IDF are sparse and long vectors, where each vector corresponds to a unique word in the corpus, and dimensions correspond to the collection of words in the vocabulary. With this method, function words like "the", "more" and "but" are assigned lower weights due to their high frequency across all the documents. Unlike TF, TF-IDF provides insights into the relative importance of words by considering their distribution across other feedback instances. However, it still does not capture word order or semantic relationships between words, which is a significant limitation for addressing complex NLP tasks like TC.

After pre-processing, we convert the tokenized text instances directly into TF-IDF vectors using the TfidfVectorizer[7] from scikit-learn (Pedregosa et al., 2011). This method combines the steps of tokenization, building a vocabulary, and computing the term frequencies and inverse document frequencies in one step. The TF-IDF transformation refines the initial representations by weighting the word frequencies, thereby adjusting the data representation with insights about the word's significance across all feedback instances in the dataset.

SVMs: Hyper-parameter Tuning

The linear Support Vector Machines algorithm has several parameters, all of which have an impact on the model's performance. Key among these is the soft margin regularization parameter, denoted as C, which controls the trade-off between achieving correct classification of instances and maximizing the decision function's margin. A lower value of C results in a wider margin and allows more margin violations, thus potentially increasing the model's ability to generalize at the risk of more misclassifications. Another parameter is *loss*, which defines whether hinge loss or squared hinge loss is used for the model. Lastly, parameter *tol* is a numeric value determining the tolerance for stopping criteria (Chauhan et al., 2019). In order to optimize the model's parameters, an exhaustive search was conducted using the GridSearchCV[8] tool from the scikit-learn library (Pedregosa et al., 2011). The hyper-parameter search included the following:

- **C**: The values 0.01, 0.1 and 1.0 were tested to determine the best trade-off between maximizing margin width and minimizing classification errors.
- **loss**: Options included 'hinge' and 'squared hinge'. The former is the standard loss setting, while the latter is its squared variant.
- **tol**: The tolerance values 1e-4, 1e-3, 1e-2 and 1e-1 were tested to balance computational efficiency against model accuracy.

The grid search was implemented to evaluate the combinations of these parameters using a 10-fold cross-validation approach with the training data. This method improves the reliability of the parameter optimization process by averaging results across

[7] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html

[8] https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

different subsets of the data, thus reducing the potential for overfitting. The goal of the search was to maximize the macro F1 score, a measure of classification accuracy that considers both precision and recall, which is suitable for our imbalanced multi-label dataset. These metrics will be further explained in Chapter 4. The optimized hyper-parameters were then used to retrain the model and evaluate it using unseen data instances in test set. The described optimization process was applied to both one-step and two-step classification separately, which allowed us to compare differences in performance. The identified hyper-parameter values are C: 1, *loss*: squared hinge, *tol*: 0.0001 for the one-step, and C: 1, *loss*: squared hinge, *tol*: 0.01 for the two-step approach. We used these values to evaluate the model on the test set.

3.2.3 Transfer Learning and Fine-Tuning

Throughout the history of NLP research, traditional machine learning methods have demonstrated notable effectiveness in tasks like text classification. These approaches however, share the same underlying principle that both training and test data inhabit the same feature space and the same distribution. This assumption leads to challenges when distributions vary, as previously trained systems tend to suffer from degraded performance and generally require complete retraining. Obtaining new labeled training data can be expensive, particularly when such data are rare or difficult to acquire (Pan and Yang, 2009). Transfer learning provides a solution to this issue by facilitating the transfer of knowledge from one domain or task (*source*) to another (*target*). This approach allows training and test data to originate from different but related domains and tasks. Transfer learning proves particularly useful when high-quality labeled training data is scarce, yet there is an abundant amount of training data and pre-trained systems available from a related domain (Weiss et al., 2016). In the realm of NLP, sequential transfer learning is especially significant, which refers to tasks being learned in a sequential order. In the pre-training stage the model learns representations of language on a general task or domain, while in the second stage the acquired knowledge is adapted to a new task or dataset (Ruder et al., 2019). Figure 3.7 shows the difference between conventional machine learning and the utility of transfer learning.

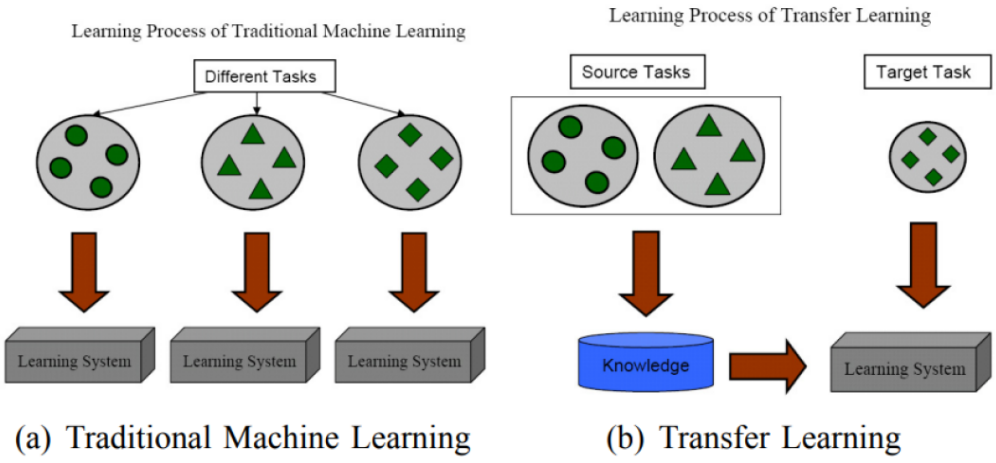

Figure 3.7: Traditional machine learning and transfer learning (Pan and Yang, 2009).

Fine-tuning is a specific form of transfer learning, where certain layers of a pre-trained model, developed for general language tasks, are adjusted for a specialized target task (Howard and Ruder, 2018). In this thesis work, a transformer-based model, initially trained on general tasks, was fine-tuned using the target domain training data for MLTC of feedback statements from the governance domain.

Introduced in 2017, Transformer models have become state-of-the-art for language modeling (Vaswani et al., 2017). They address several limitations of their predecessors, the encoder-decoder Recurrent Neural Networks (RNNs) (Cho et al., 2014). These architectures excel at transforming variable-length sequences into fixed-dimensional vector representations and then decoding them back into variable-length target sequences, which was revolutionary for the field of machine translation. On the other hand, they struggle with capturing long-range dependencies due to the vanishing gradient problem, lack parallel processing capabilities and perform slowly. Additionally, encoder-decoder RNNs are limited to unidirectional processing, meaning they process text input in only one direction. Transformers overcome these challenges using the self-attention mechanism, which allows the model to capture contextual dependencies in both directions of the input sequence, work with parallel processing and operate faster than RNNs (Vaswani et al., 2017). The sections below introduce the architecture of Transformers, then elaborate on the chosen pre-trained language model for this study, Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018).

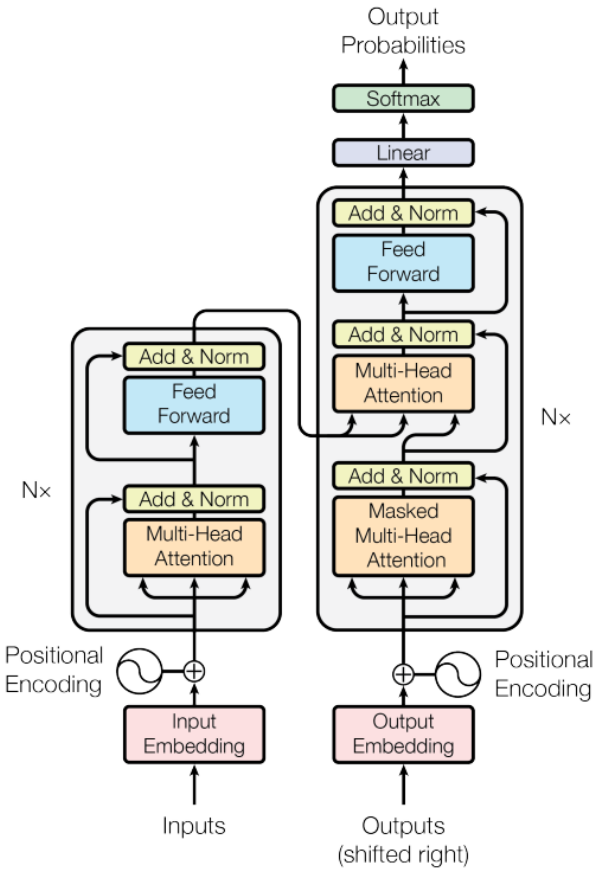

Figure 3.8: Transformer architecture (Vaswani et al., 2017).

The paragraphs below describe the general architecture of the Transformer, which was introduced by Vaswani et al. (2017). The descriptions are based on the original paper and the work of Alammar (2018a). At its core, the Transformer consists of multiple layers arranged into encoder and decoder stacks, which are linked together. Vaswani et al. (2017) use six stacks of encoding and decoding components in the original paper, but this number can vary. Each encoder layer consists of two sub-layers. The inputs first pass through a self-attention layer, which enables the encoder to consider both preceding and succeeding tokens when encoding a specific token. The outputs from the self-attention layer are then passed independently to a feed-forward neural network. Figure 3.8 provides an overview with additional explanations related to the model architecture. It shows that there are other components in the architecture the Transformer relies on, namely the "Add & Norm" layers. *Add* stands for residual connections around the sub-layers responsible for smooth gradient flow through the network by preventing the vanishing gradient problem. The *Norm* part denotes layer normalization, where the vector representations are normalized in each batch in order to control the convergence stability (Voita, 2022).

The following sections explore the model's architecture on a deeper level. Initially, each input word is transformed into a vector using an embedding algorithm at the base layer of the encoder. In order to maintain the order of words in input sentences, the model employs positional encoding. This encoding generates a vector that is added to each input embedding, ensuring that information about the sequence of words is preserved. This combined vector then progresses through two sub-layers within the first encoder: the self-attention layer and the feed-forward neural network. The encoder layers iteratively refine the word representations, with every word undergoing a unique path through the stack of encoders.

As previously introduced, self-attention is the core of the Transformer architecture, enabling the model to encode the current token in relation to all other tokens within the input sequence. This mechanism allows the model to dynamically adjust the representation of a token based on its relationship with other tokens in the sentence (Voita, 2022). Three vectors – query, key and value – play crucial roles in the operation of the self-attention mechanism. The first step is calculating these vectors for each word by multiplying the embedding vectors with three trained matrices. In the second step, a relevance score is calculated between each word and all other words in the sequence, which determines the amount of focus that should be placed on other words in the input. This score is derived from the dot product of the query vector of one word and the key vector of every other word. To stabilize the gradients, these scores are then scaled down by the square root of the dimension of the key vectors. These values are passed through a softmax layer, which normalizes the values to ensure they are positive and sum up to 1. Consequently, each value vector is multiplied by the obtained softmax score with the aim to keep the values of highly relevant words intact and disregard irrelevant words. Lastly, the weighted value vectors are summed up, which produces the output of the self-attention layer for a given word, and the resulting vector can feed into a feed-forward neural network. To optimize computational efficiency, self-attention calculations are performed using matrix operations. This involves constructing Query, Key, and Value matrices by stacking the embeddings and multiplying them with their respective trained weight matrices. The final self-attention outputs are then computed efficiently using these matrices, followed by the softmax operation, as illustrated in Figure 3.9.

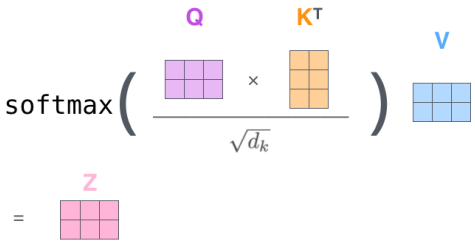

Figure 3.9: Self-attention calculation in matrix form (Alammar, 2018a).

An important feature of the Transformer architecture is the multi-headed attention mechanism, which enhances to model's ability to simultaneously address different aspects of the input sequence. This mechanism operates by expanding the model's focus across multiple positions and introducing varied representational subspaces. Instead of relying on a single set of query, key and value weights, this mechanism divides the attention into multiple independent "heads". Each head performs self-attention calculations using its own set of trained weight matrices, allowing the model to process different dimensions of the input data simultaneously. The data is split across these heads determined by the *Query Size*, which equals the *Embedding Size* divided by the *Number of Heads*. This operation leads to Z matrices that are concatenated and combined with an additional weights matrix before being processed through the feed-forward neural network (Doshi, 2021). Figure 3.10 provides a visual representation of the multi-headed attention mechanism. The dimensions of the data are influenced by certain hyper-parameters, namely the embedding size, the query size, the number of attention heads and the batch size (Doshi, 2021).

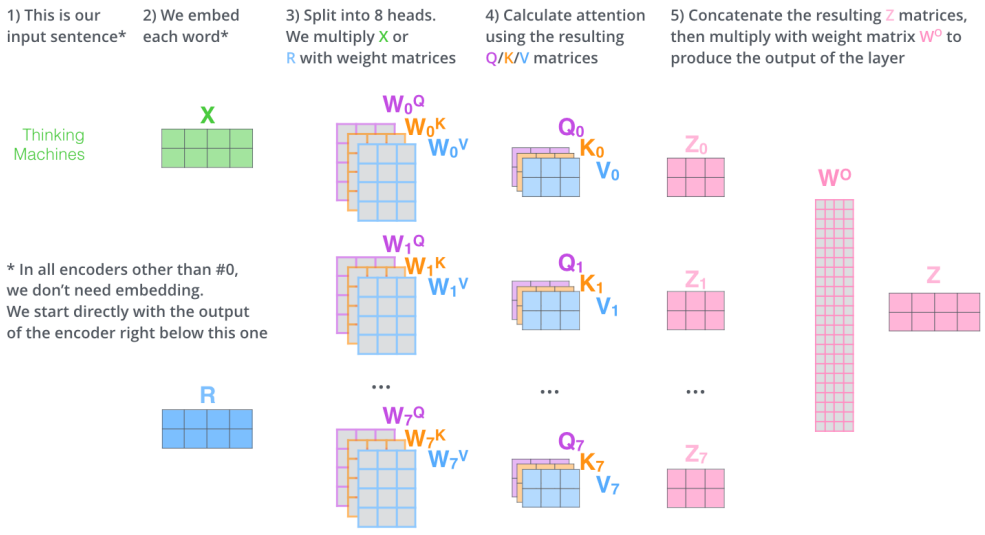

Figure 3.10: Multi-headed self-attention (Alammar, 2018a).

Fine-Tuning a Pre-trained BERT Model

Training a Transformer model from scratch requires a large collection of data, computational resources and a lot of time. Leveraging pre-trained language models (PTM) offers a reasonable alternative, since they can be fine-tuned for downstream NLP tasks – such as topic classification – using a relatively small labeled dataset and less processing power (Hadi and Fard, 2023). For this particular study, it has been decided to fine-tune a Bidirectional Encoder Representations from Transformers (**BERT**) model introduced by Google (Devlin et al., 2018). The model was chosen for this study due to its state-of-the-art performance in various NLP tasks, including question answering and text classification, achieved through bidirectional understanding of context (Devlin et al., 2018). It is particularly adaptable for downstream tasks with limited labeled data, since fine-tuning the model does not require an extensive dataset (Weiss et al., 2016). Moreover, studies have demonstrated the effectiveness of BERT for multi-label document- and text classification (Chang et al., 2020; Adhikari et al., 2019). Since its introduction, BERT has been integrated into various platforms like Hugging Face, which provides pre-trained models for deployment. The following sections describe this particular transformer-based model in detail. Figure 3.11 is the simplified illustration of a BERT model fine-tuned for the MLTC task.

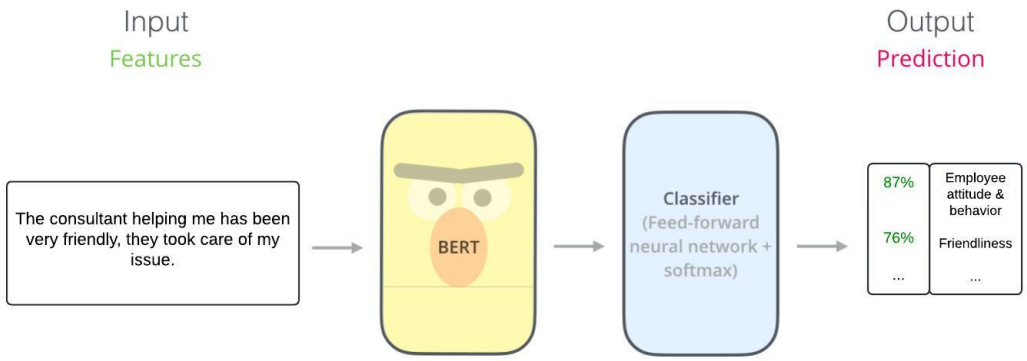

Figure 3.11: BERT for multi-label topic classification, adapted from Alammar (2018b).

Pre-training BERT was carried out in a semi-supervised manner using a large unlabeled dataset comprising the BooksCorpus and the English Wikipedia, totaling 16 GB with 3.3 billion words (Khan, 2021). Unlike the standard encoder-decoder Transformer architecture, the model utilizes a bidirectional encoder to learn contextual representations from text both before and after each word across all layers. The model training involves two primary objectives: Masked Language Model (MLM) and Next Sentence Prediction (NSP). With regards to MLM, random tokens are masked in the input, and the model's task is to predict the masked word by considering its left and right context. Specifically, 80% of the tokens are replaced with a `[MASK]` token, 10% with a random token, and the remaining 10% are left unchanged. The advantage of this training objective is that the model is forced to learn the contextual relationships between words. The NSP task aims to enhance the model's understanding of sentence relationships, which is crucial for downstream tasks such as Question Answering and Natural Language Inference. For BERT, a binary task was implemented where the

model determines whether two sequentially presented sentences are consecutive (50% of the time) or randomly paired (50% of the time) from the corpus. The input format for this task is `[CLS]` ⟨*Sentence A*⟩`[SEP]` ⟨*Sentence B*⟩`[SEP]`. The special `[CLS]` token marks the beginning of the input, while `[SEP]` acts as a special separator token between input sequences (Devlin et al., 2018).

The $\text{BERT}_{\text{BASE}}$ model was chosen for this work since it balances high performance and computational efficiency. This model consists of 12 layers with 110 million parameters, and has 768 hidden units across 12 attention heads. It is configured with a maximum input sequence length of 512 tokens, incorporates dropout for regularization, and uses the Gaussian Error Linear Unit (GELU) activation function, which enhances performance over traditional ReLU and ELU activations (Malte and Ratadiya, 2019). BERT employs the WordPiece tokenization algorithm to handle unknown words more effectively by breaking them down into subwords that are part of the vocabulary (Devlin et al., 2018). The model is publicly available on the Hugging Face platform[9] and is considered state-of-the-art for several NLP tasks (Hadi and Fard, 2023). The model does not differentiate between lowercase and uppercase words in the input, which makes it more computationally efficient compared to the large BERT variant.

For both one-step and two-step classification approaches, some model settings were standardized to facilitate a direct comparison of model performance across the experimental setups. The number of training epochs is set to **5**, the maximum sequence length is **128**, and the dropout rate is **0.1**. A random seed was set to ensure the reproducibility of experiments. Given the multi-labeled nature of the data, Binary Cross-Entropy Loss (BCE) was employed instead of the traditional Cross Entropy loss. BCE allows for the independent calculation of loss for each label, making it well-suited for binary classification tasks within a multi-label context. Additionally, the sigmoid activation function was used to derive output probabilities for each label, ensuring that the probabilities are independent and range between 0 and 1. Labels with probabilities above the defined threshold of 0.5 are classified as positive. The model architecture incorporates a BERT neural network supplemented by a dropout layer for regularization and a linear layer for classification, with the Adam optimizer[10] (Kingma and Ba, 2014) and a StepLR scheduler[11] used for weight adjustments to enhance model performance. The validation set facilitates hyper-parameter tuning, while the test set is reserved for the final model evaluation.

The initial step in processing involves preparing the privacy-masked data for model input, whereby text data is first tokenized using the built-in WordPiece tokenizer of BERT. This tokenizer converts feedback statements into lowercase and transforms them into a uniform sequence length of 128 tokens using padding. Subsequently, topic labels are encoded, and data batches are prepared for processing. The fine-tuning phase begins, where the model learns to map input features to their corresponding topic labels by constructing a vector space representation for each input sequence. During training, the model's objective is to minimize prediction errors and optimize weight parameters to enhance overall performance. In the validation phase, the model assesses each unseen input sequence and calculates the probability of a sequence belonging to each topic, enabling classification across multiple labels.

Due to confidentiality agreements with MarketResponse, the data cannot be shared

[9] https://huggingface.co/google-bert/bert-base-uncased
[10] https://pytorch.org/docs/stable/generated/torch.optim.Adam.html
[11] https://pytorch.org/docs/stable/generated/torch.optim.lr_scheduler.StepLR.html

with external parties, therefore using cloud-based platforms such as Google Colaboratory[12] that provide access to GPU and TPU resources was not possible. Consequently, the experiments were conducted on a local setup featuring an 11th Gen Intel(R) Core(TM) i5-1145G7 @ 2.60GHz with a 16GB RAM.

BERT: Hyper-parameter Tuning

In the course of the research, hyper-parameter tuning was utilized to find the optimal configuration settings for the BERT model, with the help of the grid search approach using the validation set. Hyper-parameter optimization was conducted both with the one-step and two-step classification techniques to evaluate which approach yields better performance outcomes. The following hyper-parameters were tuned in the process:

- **Batch size:** The tested values were 8, 16, 32 and 64 to determine the optimal number of training samples to process before updating the model parameters.
- **Learning rate:** We experimented with 2e-5 and 3e-5 values to determine the rate at which the model learns from the training data.

These hyper-parameter values were chosen based on related literature (Liu and Wang, 2021) and discussions with MarketResponse. Due to constraints in computational resources and time, a more extensive exploration of hyper-parameters was not feasible. Each iteration of the hyper-parameter tuning process required a minimum of 14 hours on a local machine, which significantly limited the scope of experimentation. The optimal set of parameters identified were *batch size*: 8, *learning rate*: 3e-5 for the one-step approach, and *batch size*: 8, *learning rate*: 2e-5 for the two-step approach. We proceeded with the final evaluation phase on the test data with this configuration. Future research could include additional hyper-parameters, such as dropout rate, and a wider spectrum of tested hyper-parameter values.

3.2.4 One-step versus Two-step Classification

Topic labels are organized in a tree-like structure in the dataset, with each main topic having one or more corresponding subtopics. In this thesis work, we explore both one-step and two-step classification methods. The one-step classification method assigns all topic labels to a given text instance simultaneously. Although this approach has the advantage of simplicity, learning to classify instances for 42 topic labels in a single step might be challenging for models due to the complexity of the task.

Conversely, the two-step classification method, referred to as hierarchical classification, decomposes the classification problem into smaller, more manageable tasks. This means the model first predicts classes at the highest hierarchical level, followed by predicting lower-level distinctions within the initially predicted top-level categories. This approach can enhance both accuracy and efficiency by simplifying the classification process into manageable stages (Dumais and Chen, 2000). In our study, the two-step method first identifies the main topics, and then classifies the subtopics within these main categories. For example, if *Processes* is identified as main topic for a feedback instance in the first step, only its associated subtopics – which are *Ease of process* and

[12] https://colab.research.google.com/

Efficiency of Process – will be considered by the model in the second stage. Figure 3.12 illustrates the two-step classification process, where the first level focuses on the main topic prediction and the second level on the corresponding subtopic prediction.

By implementing the two-step approach, we hypothesize that the models will perform with enhanced efficiency due to the reduced number of labels at each stage. However, this method requires more computational resources and increases the total processing time. It should also be noted that the errors predicted in the first step of the two-step classification propagate to the second step, negatively affecting the overall performance. Despite these concerns, the modularity of the two-step approach may result in improved performance outcomes. Both one-step and two-step methods were applied using a conventional machine learning classifier and a transformer-based model to assess the hypothesis. For both approaches, we first transformed the multi-labeled data into a set of binary problems using the Binary Relevance approach, where the presence or absence of each topic is indicated by 1 and 0, respectively (Gonçalves and Quaresma, 2003). The main topics *General experience* and *No topic found* have only one subtopic, thus the predictions for their subtopics were directly derived from the main topic predictions in the second stage of the two-step method.

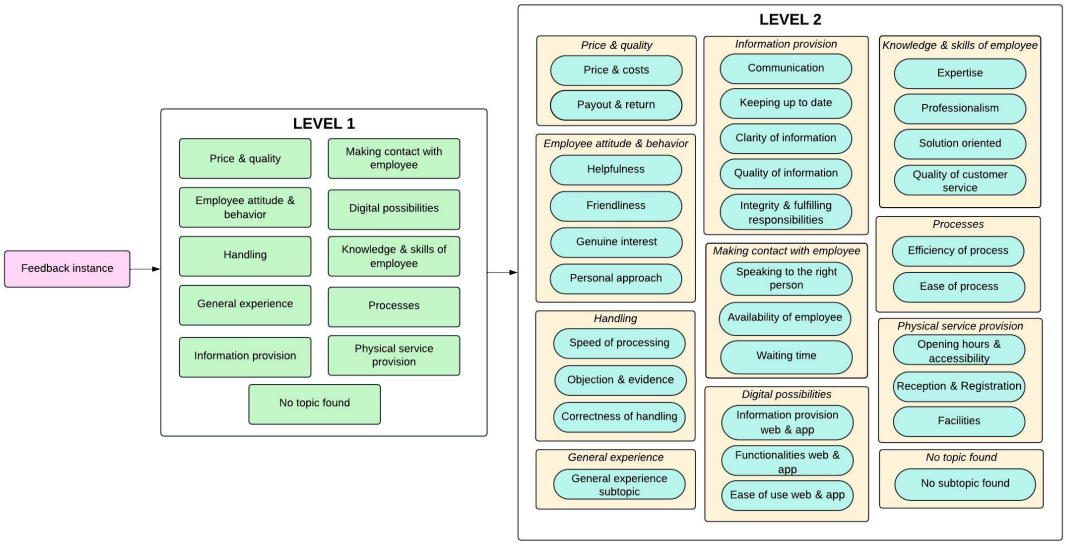

Figure 3.12: Two-step classification approach.

Chapter 4

Results

This chapter details the performance evaluation of the classification approaches in this thesis, employing the aggregated macro-averaged precision, recall and F1-scores. The sections below explain these scores based on the work of Jurafsky and Martin (2009), unless indicated otherwise.

4.1 Evaluation Metrics

In the context of multi-label classification transformed into a set of binary classification problems, the outcomes for each prediction are categorized as True Positives (TP), False Positives (FP), True Negatives (TN) and False Negatives (FN).

Precision is calculated by taking the number of correctly predicted instances, and dividing it by the sum of all TP and FP instances.

$$P = \frac{TP}{TP + FP}$$

Recall is calculated by by taking the number of correctly predicted instances, and dividing it by the sum of all TP and FN instances.

$$R = \frac{TP}{TP + FN}$$

The **F1-score** is the harmonic mean of precision and recall values and serves as a combined measure of these two metrics.

$$F1 = 2 \times \frac{P \times R}{P + R}$$

Additionally, **Hamming loss** is used to measure the proportion of incorrectly predicted labels to the total number of labels and total number of classes. This value ranges from 0 to 1, where a lower score indicates better performance. Hamming loss is relevant for multi-label classification since it captures the average number of label prediction errors per instance, offering a complementary perspective to the metrics introduced above (Tsoumakas and Katakis, 2007). It is calculated as follows:

$$HL = \frac{1}{n \cdot L} \sum_{i=1}^{n} \sum_{j=1}^{L} I(y_{ij} \neq \hat{y}_{ij})$$

In this formula n is the number of instances, and L is the number of labels in the test set. y_{ij} is the true value, while $\hat{y}_{ij}$ is the predicted value of the j-th label for the i-th instance. I is the indicator function, which is 1 if $y_{ij} \neq \hat{y}_{ij}$ and 0 otherwise (Sorower, 2010).

Macro averaging means calculating the performance metrics – precision, recall and F1-score – independently for each class and then averaging these scores. This approach is particularly favourable when dealing with imbalanced datasets, as it assigns equal weight to the performance for each class, regardless of class frequency. The evaluation covers the performance on the test set, which contains 1,954 instances and 5,349 labels, representing 10% of the complete dataset. The sections below present the results for both the original dataset using one-step and two-step classification approaches with hyper-parameter optimization, as well as the results with the undersampled and oversampled training sets. The evaluation focuses on the macro-averaged precision, recall and F1-scores. The classification reports[1] and Hamming loss[2] values were created using the scikit-learn package (Pedregosa et al., 2011).

4.2 Results

The evaluation scores are presented in light of the research questions. First introducing the results of the different classification algorithms with one-step and two-step approaches on the original dataset, then presenting the outcomes after applying two data adaptation techniques, namely undersampling and oversampling on the training set. Since the complete result overview with all 42 classes is large, the full tables can be found in the Appendix. The sections below provide an overview of the most meaningful results, presented with the macro averaged precision (p), recall (r), F1-score (f) and Hamming loss (hl).

4.2.1 Results: Original Dataset

Table 4.1 presents the results of classification algorithms trained on the original training set and tested on the test set. Both the SVMs and the fine-tuned BERT model exhibited higher macro-averaged F1-scores using the two-step approach as compared to the one-step approach, 0.56 and 0.65 respectively. The improvement is marginal for the conventional machine learning model, showing only a 0.01 increase in the macro F1-score, but more substantial for the transformer-based model, with a 0.04 increase in the macro F1-score. One can observe that the fine-tuned BERT model with its advanced architecture outperformed the SVMs model, achieving an F1-score of 0.65. However, it is noteworthy that training the SVMs required less than 10 minutes, whereas fine-tuning the BERT model necessitated a minimum of 14 hours on a local machine setup. Considering computational requirements is essential in an industrial setting, where resources are often limited.

A closer analysis of the SVMs' performance reveals that the results for the main topics were consistent across both one-step and two-step approaches. Nonetheless, there is a noticeable increase in model performance for several underrepresented subtopics with

[1] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.classification_report.html

[2] https://scikit-learn.org/stable/modules/generated/sklearn.metrics.hamming_loss.html

Classifier	**Setup**	*macro-averaged*			
		p	*r*	*f*	*hl*
SVMs	One-step classification + original training set	0.73	0.46	0.55	0.033
SVMs	Two-step classification + original training set	0.71	0.49	0.56	0.034
BERT	One-step classification + original training set	0.70	0.56	0.61	0.026
BERT	Two-step classification + original training set	**0.67**	**0.63**	**0.65**	**0.027**

Table 4.1: Results on the original dataset concerning macro-averaged precision (p), recall (r), F1-score (f) and Hamming loss (hl).

the two-step approach, such as *Professionalism*, *Facilities*, *Personal approach*, *Communication*, *Speaking to the right person*, *Quality of information*, *Quality of customer service*, *Keeping up to date* and *Correctness of handling*. For the fine-tuned BERT model, the implementation of the two-step approach led to an increase in macro recall (up to 0.63) and F1-score (up to 0.65). The improvements are noteworthy for subtopics such as *Availability of employee*, *Reception & Registration* and *Ease of use web & app*, suggesting that the two-step approach facilitates a more nuanced and effective handling of topic categories. It can be concluded that on the original dataset, the two-step approach with a fine-tuned BERT model provides superior performance.

The classifiers were able to predict the majority of topic labels with reasonable performance. Subsequently, we aimed to further increase the models' performance using data adaptation techniques. The experiments continued with the additional training sets created through undersampling and oversampling described in Chapter 3. These experiments exclusively utilized the two-step approach, since it has proven to be more advantageous over the one-step approach on the original dataset. In the next steps we employed the same optimized hyper-parameters previously established based on the original dataset.

4.2.2 Results: Undersampled Dataset

As part of the data adaptation techniques, it has been decided to implement undersampling to reduce the impact of overrepresented subtopic classes in the training set until they reach the average distribution. The best performing classifier concerning this approach remains the fine-tuned BERT model, as shown in Table 4.2.

Classifier	**Setup**	*macro-averaged*			
		p	*r*	*f*	*hl*
SVMs	Two-step classification + original training set (baseline)	0.71	0.49	0.56	0.034
SVMs	Two-step classification + undersampled training set	0.65	0.51	0.56	0.038
BERT	Two-step classification + original training set (baseline)	**0.67**	**0.63**	**0.65**	**0.027**
BERT	Two-step classification + undersampled training set	0.59	0.64	0.61	0.037

Table 4.2: Results on the undersampled dataset concerning macro-averaged precision (p), recall (r), F1-score (f) and Hamming loss (hl).

It is worth noting that in case of SVMs, the model performance only slightly changed when compared to the results on the original training set. There is an observable decrease in the macro precision (down to 0.65) and increase in the macro recall (up to 0.51), resulting in the same 0.56 macro F1-score as with the original training set. When observing the scores for each class individually, we notice a moderate decrease in the F1-scores for the undersampled subtopics, and a moderate increase in the F1-scores

for the underrepresented topics, meaning the performance is more balanced across the subtopics. We may conclude that SVMs perform with a similar success rate when less amount of training data is available, and the data is more balanced across the classes.

As for the fine-tuned BERT model, there is a drop in the macro precision (down to 0.59) and a slight increase in macro recall (up to 0.64), which results in a 0.04 lower macro F1-score compared to the original training set. The Hamming loss increased from 0.027 to 0.037, indicating a higher rate of incorrect label predictions per sample, which is an undesirable outcome for multi-label classification. As for the affected classes, many topics show an increase in recall at the expense of precision. Some topics with low representation, such as *Opening hours & accessibility* and *Payout & return*, showed no improvement and continued to have low F1-scores, highlighting that undersampling alone may not be sufficient to improve classifier performance on rare classes.

With this particular setting, the fine-tuned BERT model with the two-step approach still outperforms the SVMs model, but the gap between their macro F1-scores has decreased. The undersampling technique helps in improving recall by addressing class imbalance, but it also introduces a trade-off with precision, leading to more overall misclassifications, as evidenced by the increase in Hamming loss values.

4.2.3 Results: Oversampled Dataset

We have also explored another data adaptation technique – oversampling – to augment the representation of underrepresented subtopic classes. The rationale for oversampling is that increasing the instance count for infrequent classes can positively impact model performance. We utilized GPT-4, an open-source generative language model, to create synthetic data instances. The synthetic data was merged with the original training data to create an enriched training set. Our findings indicate that the most optimal model configuration is the fine-tuned BERT model employing the two-step classification approach with the oversampled dataset, as shown in Table 4.3.

Classifier	**Setup**	*macro-averaged*			
		p	*r*	*f*	*hl*
SVMs	Two-step classification + original training set (baseline)	0.71	0.49	0.56	0.034
SVMs	Two-step classification + oversampled training set	0.71	0.49	0.57	0.033
BERT	Two-step classification + original training set (baseline)	0.67	0.63	0.65	0.027
BERT	Two-step classification + oversampled training set	**0.69**	**0.66**	**0.66**	**0.027**

Table 4.3: Results on the oversampled dataset concerning macro-averaged precision (p), recall (r), F1-score (f) and Hamming loss (hl).

When comparing the performance metrics of the SVMs model on both the original and oversampled datasets, we observed a marginal improvement in the macro F1-score, which increased to 0.57, alongside a reduction in the Hamming loss to 0.033. The oversampling strategy has generally enhanced model performance, particularly for the underrepresented classes, by improving either recall, precision, or both metrics. This resulted in higher F1-scores for several underrepresented subtopic classes, including *Information provision web & app*, *Quality of information*, *Functionalities web & app* and *Speaking to the right person*. The performance metrics for classes not directly impacted by oversampling remained relatively stable between the two datasets. Regarding the fine-tuned BERT model, an increase is notable in the macro precision (up to 0.69), resulting in a macro F1-score of 0.66 and a Hamming loss value of 0.027. Overall,

oversampling has proven to effectively enhance the BERT model's sensitivity, particularly increasing the recall for initially underrepresented classes such as *Keeping up to date*, *Price & costs*, *Availability of employee*, *Correctness of handling* and *Speaking to the right person*. The detailed overview of results for the best performing classification setup can be observed in Figure 4.4, tables for other models are in the Appendix.

	Topic	Precision	Recall	F1-score	Support
1	Making contact with employee	0.73	0.88	0.80	144
2	Processes	0.83	0.78	0.81	165
3	Digital possibilities	0.82	0.51	0.63	35
4	General experience	0.77	0.71	0.74	165
5	Information provision	0.79	0.83	0.81	313
6	Employee attitude & behavior	0.93	0.90	0.91	617
7	Handling	0.82	0.81	0.81	477
8	No topic found	0.70	0.49	0.57	173
9	Knowledge & skills of employee	0.78	0.75	0.77	284
10	Price & quality	0.38	0.33	0.36	15
11	Physical service provision	0.74	0.79	0.76	142
12	Waiting time	0.81	0.88	0.84	99
13	Speaking to the right person	0.68	0.95	0.79	22
14	Correctness of handling	0.61	0.50	0.55	22
15	Functionalities web & app	0.64	0.53	0.58	17
16	Ease of process	0.75	0.73	0.74	106
17	Reception & Registration	0.74	0.78	0.76	97
18	Friendliness	0.95	0.95	0.95	454
19	Quality of information	0.74	0.72	0.73	67
20	Information provision web & app	0.83	0.42	0.56	12
21	Clarity of information	0.82	0.88	0.85	197
22	Solution oriented	0.75	0.76	0.75	181
23	Availability of employee	0.51	0.74	0.61	31
24	Price & costs	0.38	0.36	0.37	14
25	Speed of processing	0.81	0.81	0.81	467
26	Professionalism	0.88	0.78	0.82	36
27	Opening hours & accessibility	0.14	1.00	0.25	1
28	Ease of use web & app	0.71	0.28	0.40	18
29	Keeping up to date	0.60	0.40	0.48	15
30	Integrity & fulfilling responsibilities	0.70	0.59	0.64	54
31	Payout & return	0.00	0.00	0.00	1
32	No subtopic found	0.70	0.49	0.57	173
33	Quality of customer service	0.82	0.68	0.74	47
34	Facilities	0.73	0.67	0.70	49
35	Objection & evidence	0.33	0.25	0.29	4
36	General experience subtopic	0.77	0.71	0.74	165
37	Efficiency of process	0.85	0.77	0.81	66
38	Genuine interest	0.83	0.83	0.83	81
39	Expertise	0.67	0.63	0.65	75
40	Helpfulness	0.83	0.73	0.78	172
41	Personal approach	0.73	0.35	0.47	23
42	Communication	0.56	0.79	0.66	53
	macro avg	**0.69**	**0.66**	**0.66**	5349
	weighted avg	0.80	0.78	0.79	5349
	hamming loss			0.027	5349

Table 4.4: Results overview on the oversampled dataset using two-step classification with BERT after hyper-parameter optimization.

4.2.4 Results: Concluding Remakrs

Before continuing with a more detailed analysis, some conclusions can be drawn about the optimal model configuration. When observing the results on the original dataset, the fine-tuned BERT model outperformed the conventional SVMs model. These experiments also confirmed that both classifiers benefit from the two-step classification approach. The transformer-based model performed best on the oversampled dataset, reaching an F1-score of 0.66 and a Hamming loss value of 0.027.

The next section provides an in-depth error analysis of the best-performing model. The error analysis aims to discover specific misclassification issues and recurring error patterns, which are essential for refining the model's performance in future research and promote explainability. Understanding these patterns can also help in tailoring future modeling strategies to mitigate these errors, thereby enhancing overall model performance.

4.3 Error Analysis

As stated above, this section aims to analyze the predictions of the most efficient model configuration in more detail. Specifically, we examine the performance of the fine-tuned BERT model that employs the two-step classification approach and was trained on the oversampled training dataset.

The error analysis is composed of a quantitative and qualitative part. In the quantitative section, we inspect the confusion matrices illustrating the alignment between the gold labels and the predicted labels. Due to the multi-labeled nature of the data, there is a separate confusion matrix created for each class. Figure 4.1 displays the confusion matrices for main topic labels and Figure 4.2 for subtopic labels. The examples introduced in this chapter are fabricated to preserve the confidentiality of the data. The analysis is structured by the frequency distribution of the classes in the test set, starting with the most frequent class and progressing to the least frequent one. First, we focus on the main topic, then on the subtopic predictions. Subtopics related to the same main topic are grouped together in the analysis. The qualitative part aims to summarize the underlying reasons for classification errors and provide recommendations for improving the model.

4.3.1 Quantitative Analysis

Analysis of Main Topic Predictions

When observing the precision, recall and F1-scores for the main topics shown in Table 4.4, it is notable that the F1-scores range between 0.36 and 0.91. The model exhibits lower performance for classes with fewer instances, such as *Digital possibilities* and *Price & quality*, compared to classes with higher representation like *Handling* or *Employee attitude & behavior*. Precision is higher than recall for the majority of main topic classes, which is likely to be the result of the imbalanced class distribution. In the following analysis, we aim to gain a better understanding of the strengths and weaknesses of the model by observing the classification errors across different main topic classes. The confusion matrices for main topic labels are shown in Figure 4.1.

Employee attitude & behavior
This is the most frequently occurring label in the test set, with 617 positive instances in the gold data. The model made 105 classification errors, composed of 42 FP and 63 FN instances. The model most frequently confused this main topic with *Knowledge & skills of employee* (18 times), *Handling* (16 times) and *Information provision* (11 times), as the examples illustrate below.

- "The workers were friendly and you were treated properly." – misclassified as *Knowledge & skills of employee*
- "It's great to have people available who are willing to discuss and assist you." – misclassified as *Knowledge & skills of employee* and *General experience*
- "I don't understand why I need to make an appointment if I'm only going to be helped 30 minutes later." – misclassified as *Handling*
- "I asked for certain information, which they searched for and provided to me accurately." – misclassified as *Information provision*

Handling
As the second most frequent main topic label in the test set, this class has 477 positive instances in the gold test data. The model made 87 FP and 93 FN predictions, totalling 180 misclassified instances. Most commonly the model confused this topic with *Making contact with employee* (25 times), *Processes* (19 times) and *Information provision* (16 times), see the examples below.

- "I had to wait 50 minutes before my turn. Is there a chance to improve the tool that predicts the length of all appointments?" – misclassified as *Making contact with employee*
- "I was expecting them to contact me on DATE but I'm still waiting for that call." – misclassified as *Making contact with employee*
- "We received quick and professional assistance, and the woman who helped us even went the extra mile to ensure we didn't need to return for a second visit." – misclassified as *Processes*
- "The information I requested was accurately researched and provided." – misclassified as *Information provision*

Information provision
This topic ranks as the third most frequent main topic label in the test set, having 313 positive examples in the test data. The model incorrectly classified 121 instances, including 67 FP and 54 FN. It most often mixed up this topic with *Handling* (11 times), *Knowledge & skills of employee* (9 times) and *Physical service provision* (8 times), as illustrated in the examples below.

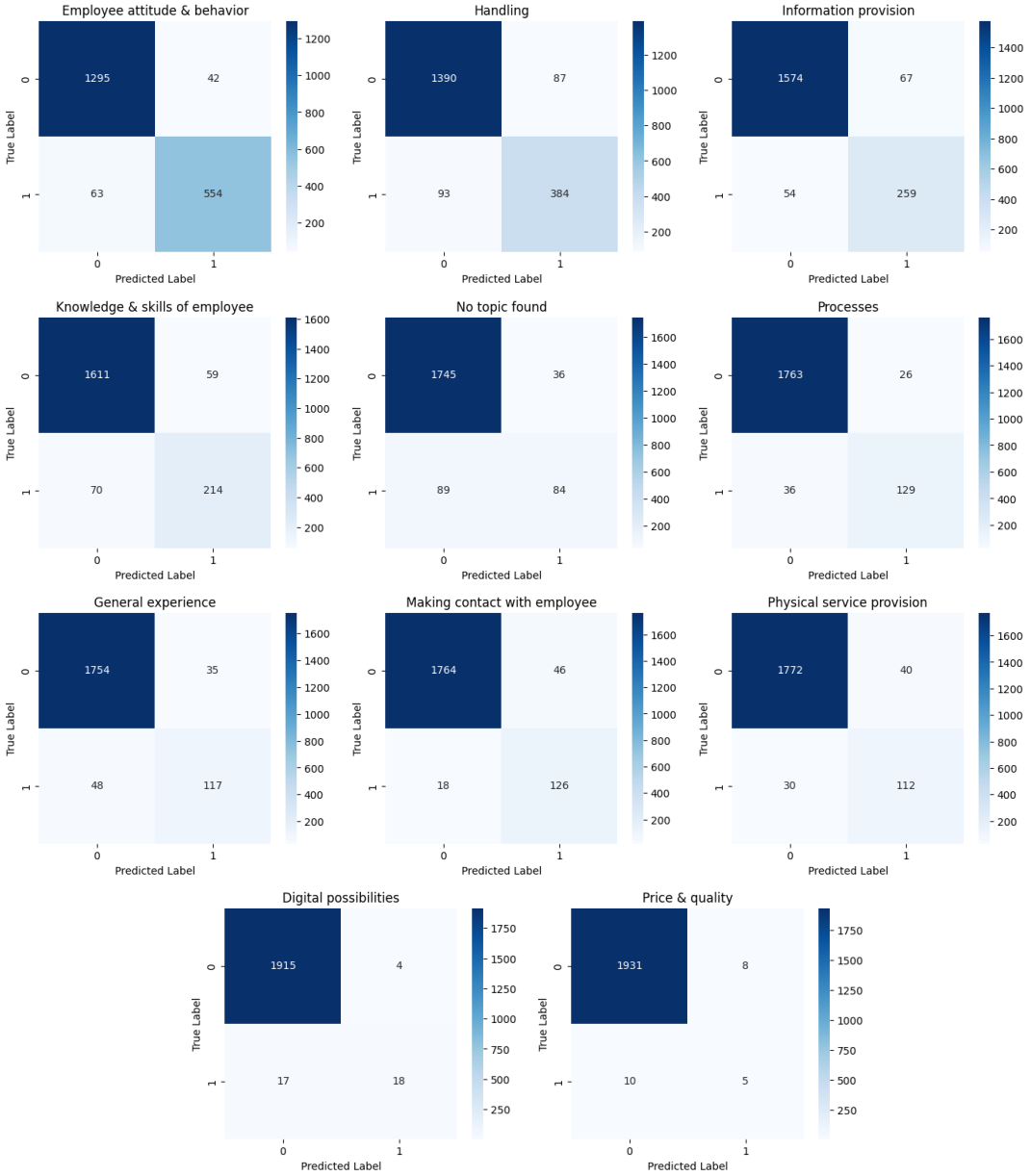

Figure 4.1: Confusion matrices for main topics using the fine-tuned BERT model with two-step classification on the oversampled dataset.

- "There was about a half-hour gap between my appointment time and when I was actually assisted." – misclassified as *Handling*
- "I wasn't assisted and had to email another department, which hasn't responded yet." – misclassified as *Knowledge & skills of employee*
- "I received the most suitable answer for my needs." – misclassified as *Knowledge & skills of employee*
- "I was told to mail the forms instead of submitting them in person at the office, which has delayed my registration." – misclassified as *Physical service provision*

Knowledge & skills of employee
This main topic appears 284 times as positive examples in the test data. The model misclassified 129 instances, with 59 FP and 70 FN. The model frequently confused this topic with *Employee attitude & behavior* (26 times), *Handling* (19 times) and *Processes* (10 times), as shown in the examples below.

- "I am from LOCATION where customer service is often poor. Here, you are treated with respect and without any judgement. It's truly excellent." – misclassified as *Employee attitude & behavior*
- "The lady provided a clear and thorough explanation, taking great care." – misclassified as *Employee attitude & behavior*
- "The workers are helpful, and the service works fast." – misclassified as *Handling* and *Employee attitude & behavior*
- "Extremely accommodating and empathetic, eager to meet the needs and goes the extra mile to serve customers." – misclassified as *Processes* and *Employee attitude & behavior*

No topic found
This main topic is present 173 times as positive in the gold data. The model incorrectly classified the topic in 129 cases, consisting of 36 FP and 89 FN. The model often mistakenly identified this topic as *Information provision* (19 times), *Handling* (14 times) and *Physical service provision* (12 times), as demonstrated by the examples below.

- "Terms and choices were clarified." – misclassified as *Information provision*
- "I had to talk to too many workers on the phone." – misclassified as *Information provision*
- "Adopt a paperless system. If someone sets an appointment, they should be attended to at the scheduled time. We ended up waiting for 40 minutes in line before an employee could see us. A week after our appointment, we received a notification that my husband was still not registered!" – misclassified as *Handling* and *Information provision*
- "A clearer signage indicating the locations of the counter rooms and the meeting rooms would be helpful." – misclassified as *Physical service provision*

Processes
The main topic has 165 positive instances in the test set. The model misclassified a total of 36 instances, including 26 FP and 36 FN examples. The most common incorrect classification patterns of the model are *Handling* (13 times), *Making contact with employees* (8 times) and *Knowledge & skills of employee* (5 times), as illustrated below.

- "The entire process was very smooth." – misclassified as *Handling*
- "At last, an employee who was willing to thoroughly investigate my question. Before this, I had called four times for the same issue, always being told to wait or call back." – misclassified as *Making contact with employee*
- "Easily reachable." – misclassified as *Making contact with employee*
- "I've visited the office of LOCATION three times now, and the service has been excellent each time!" – misclassified as *Knowledge & skills of employee*

General experience
The main topic has 165 positive examples in the test data. The model made 83 classification errors, composed of 35 FP and 48 FN instances. Most commonly the model confused this topic with *No topic found* (14 times), *Knowledge & skills of employee* (9 times) and *Employee attitude & behavior* (6 times).

- "Everything was fine." – misclassified as *No topic found*
- "It was fine, I have been through worse." – misclassified as *No topic found*
- "I was happy when an employee came to help me out with getting a waiting number." – misclassified as *Knowledge & skills of employee*
- "The lady fully understood the issue and patiently responded, making sure I understood her instructions correctly." – misclassified as *Employee attitude & behavior*

Making contact with employee
This particular topic has 144 positive instances in the gold data. There is a total of 64 misclassifications by the system, including 46 FP and 18 FN instances. The model had the tendency to confuse this topic with *Handling* (8 times), *Knowledge & skills of employee* (3 times) and *Information provision* (2 times), as shown by the examples below.

- "We got in contact quickly and I received accurate information." – misclassified as *Handling*
- "Display more availability in the calendar; many future dates are blocked. Consider opening the calendar for at least three months. It took me a month to get an appointment scheduled, and even phone assistance was of no help." – misclassified as *Handling*

Physical service provision
This main topic has 142 positive occurrences in the test data. The model misclassified 70 instances, resulting in 40 FP and 30 FN instances. Most commonly the model confused this labels with *Making contact with employee* (12 times), *Information provision* (9 times) and *Handling* (7 times), see the examples below.

- "Welcoming and calm guidance, clear explanations, and helpful answers." – misclassified as *Information provision*

- "Warm welcome, clear explanation about the QR code. The lady at the desk was also very kind and provided great assistance." – misclassified as *Information provision*

- "My passport photo was rejected for a driver's license, yet the same photo was accepted for my passport six months earlier. Isn't that odd?" – misclassified as *Handling*

- "It should be more clearly marked what waiting rooms A and B are. The woman at the counter was friendly, but she spoke so quickly that I had to ask her to repeat herself." – misclassified as *Handling*

Digital possibilities
This class appears 35 times in the test data. The model made 21 classification mistakes, including 4 FP and 17 FN cases. The model most frequently confused this topic label with *Information provision* (6 times), *Handling* (5 times) and *No topic found* (2 times), illustrated below.

- "The employee was able to explain many things. She only didn't know how to answer questions about the app." – misclassified as *Information provision*

- "It's often hard to find available times at the city center office. I had to frequently check online for openings." – misclassified as *Handling*

- "The website provides a lot of information and allows for quick appointment scheduling, taking only a few minutes." – misclassified as *Handling*

- "It all depends if you 'click' with someone or not." – misclassified as *No topic found*

Price & quality
This is the least frequent main topic in the test data, with 15 instances. The model misclassified it 18 times, including 8 FP and 10 FN cases. Most often the instances were confused with *Making contact with employee* (4 times), *Processes* (2 times) and *Handling* (2 times), as illustrated below.

- "I had to wait for 50 minutes to renew my passport. Could you adjust the tool used for estimating the waiting time?" – misclassified as *Making contact with employee*

Analysis of Subtopic Predictions

In the following paragraphs we inspect the recurring misclassification patterns for subtopic labels. As detailed in Table 4.4, the F1-scores for subtopics differ on a large

spectrum, ranging from 0 to 0.95. Mirroring the trend observed with main topics, the model demonstrates robust classification capabilities for subtopics that are well-represented in the training set, such as *Friendliness* and *Speed of processing*. Conversely, it displays notable deficiencies in handling underrepresented classes like *Payout & return* and *Opening hours & accessibility*. The balance between precision and recall varies across subtopic labels, displaying no consistent trend in whether precision or recall predominates. The subsequent analysis aims to categorize these classification errors to uncover underlying patterns and possible explanations for misclassifications. The confusion matrices for subtopics can be observed in Figure 4.2. This section does not discuss the subtopics for *No topic found* and *General experience*, since for these main topics the subtopic prediction is directly derived from the main topic prediction.

Subtopics within Employee attitude & behavior
The subtopics *Friendliness*, *Helpfulness*, *Genuine interest* and *Personal approach* are categorized under this main topic. For *Friendliness*, there were 23 FP and 23 FN. For *Helpfulness*, the model registered 25 FP and 46 FN, with *Genuine interest* had 14 FP and 14 FN and *Personal approach* had 3 FP and 15 FN classification errors. The main tendency of the model is to confuse these subtopics with each other. For instance, there are 22 instances where the model incorrectly predicted only *Friendliness* when both *Friendliness* and *Helpfulness* were indicated in the gold data. The following examples highlight typical cases of misclassification.

- "I was satisfied with the assistance I received; the employees were both friendly and helpful." – assigned *Friendliness* instead of *Friendliness* and *Helpfulness*

- "Excellent, timely and polite staff." – assigned *Speed of processing* instead of *Friendliness* and *Speed of processing*

- "I received personal assistance and they thought along with me." – assigned *Genuine interest* instead of *Solution oriented*, *Personal approach* and *Genuine interest*

Subtopics within Handling
The subtopics *Speed of processing*, *Correctness of handling* and *Objection & evidence* are part of this main topic. For *Speed of processing*, there are 87 FP and 88 FN instances, 7 FP and 11 FN for *Correctness of handling* and 2 FP and 3 FN for *Objection & evidence*. The model shows the trend of confusing *Speed of processing* with *Ease of process* (16 times) and *Waiting time* (14 times). The subtopic *Correctness of handling* is sometimes confused with *Friendliness* (3 times). The examples below illustrate these misclassification patterns.

- "It was easy to apply for the document and the postman delivered it fast." – assigned *Ease of process* instead of *Speed of processing*

- "You shouldn't make people wait in the lobby for such a long time" – assigned *Waiting time* instead of *Speed of processing*

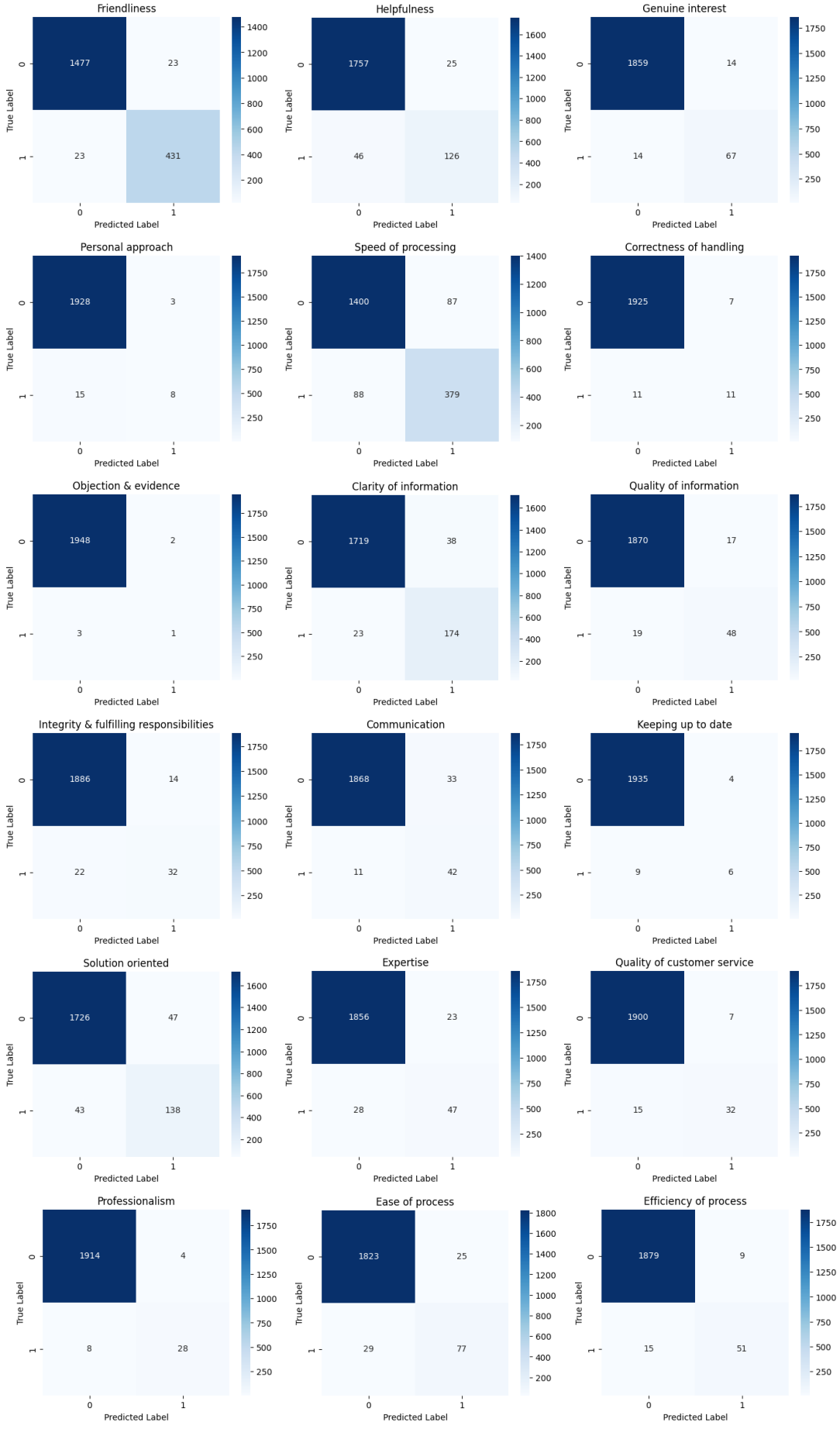

Figure 4.2: Confusion matrices for subtopics using the fine-tuned BERT model with two-step classification on the oversampled dataset (part I).

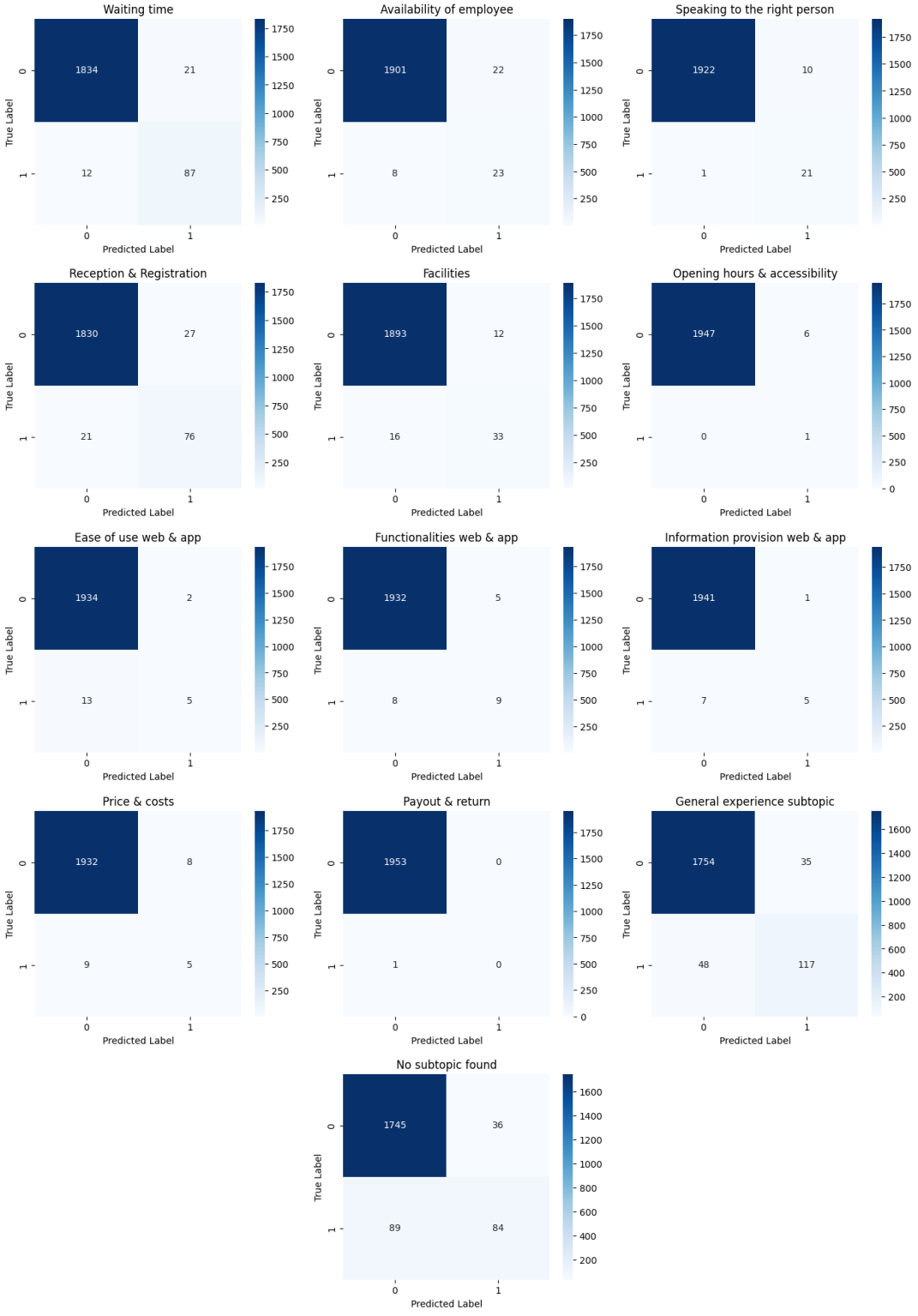

Figure 4.2: Confusion matrices for subtopics using the fine-tuned BERT model with two-step classification on the oversampled dataset (part II).

- "The employee was very open and friendly throughout the whole process." – assigned *Friendliness* instead of *Friendliness*, *Correctness of handling*, *Speed of processing*

Subtopics within Information provision
The subtopics *Clarity of information*, *Quality of information*, *Integrity & fulfilling responsibilities*, *Communication* and *Keeping up to date* belong to this main topic. For the label *Clarity of information* the model predicted 38 FP and 23 FN instances, 17 FP and 19 FN for *Quality of information*, 14 FP and 22 FN for *Integrity & fulfilling responsibilities*, 33 FP and 11 FN for *Communication*, and 4 FP and 9 FN for *Keeping up to date*. A common patter is confusing the subtopic labels belonging to this main topic with one another, for example the model mistook *Quality of information* for *Clarity of information* 4 times. The examples below illustrate the model's behavior for these subtopics.

- "I don't understand why it's an obligation to translate all the documents for the application. Why can't you accept my Spanish birth certificate?" – assigned *Communication* instead of *Quality of information* and *Clarity of information*

- "Only half an hour passed between the appointment and the time I was given help." – assigned *Speed of processing* instead of *Integrity & fulfilling responsibilities*

- "Nobody has helped me so far. I sent an email to another department and got no reply." – assigned *Solution oriented* instead of *Communication*

Subtopics within Knowledge & skills of employee
This main topics has four associated subtopics: *Solution oriented*, *Expertise*, *Quality of customer service* and *Professionalism*. The model made 47 FP and 43 FN predictions for *Solution oriented*, 23 FP and 28 FN predictions for *Expertise*, 7 FP and 15 FN for *Quality of customer service*, and 4 FP and 8 FN for *Professionalism*. When observing the misclassifications, it is notable that the model often predicts the overrepresented *Speed of processing* instead of the smaller *Expertise* (10 time) or *Solution oriented* (10 times). The model also tends to confuse *Professionalism* with *Expertise*, as the examples show below.

- "I have been provided professional help." – assigned *Expertise* instead of *Professionalism*

- "The staff member quickly understood what I want and helped me fast." – assigned *Speed of Processing* instead of *Speed of processing* and *Expertise*

- "The staff was helpful and kind, there was no queue, the case was closed fast." – assigned *Speed of processing* and *Helpfulness* instead of *Solution oriented*

Subtopics within Processes
The subtopics *Ease of process* and *Efficiency of process* belong to this main topic.

The model made 25 FP and 29 FN misclassifications for *Ease of process*, and 9 FP and 15 FN misclassifications for *Efficiency of process*. The model has the tendency to confuse these subtopics with the better represented *Speed of processing*. This happened 10 times for *Ease of process* and 5 times for *Efficiency of process*, as the examples illustrate below.

- "I could only reach the central office after calling three times" – assigned *Speed of processing* instead of *Efficiency of process*

- "It would be better if you extended the opening hours and hired more staff. You should consider employing part-time students, making the process faster." – assigned *Speed of processing* instead of *Ease of process*, *Availability of employee*, *Expertise*

Subtopics within Making contact with employee
This main topic has three associated subtopics, these are *Waiting time*, *Availability of employee* and *Speaking to the right person*. We can find 21 FP and 12 FN instances for *Waiting time*, 22 FP and 8 FN for *Availability of employee*, and 10 FP and 1 FN for *Speaking to the right person*. The observed misclassifications suggest that the model often mistakes *Waiting time* and *Availability of employee* for *Speed of processing* (6 and 3 times respectively). The examples below illustrate this trend.

- "It was my turn without having to wait for long" – assigned *Speed of processing* instead of *Waiting time* and *Speed of processing*

- "You should indicate more availability options in your calendar." – assigned *Speed of processing* instead of *Availability of employee*

Subtopics within Physical service provision
This main topic has three corresponding subtopics, namely *Reception & Registration*, *Facilities* and *Opening hours & accessibility*. The model had 27 FP and 21 FN predictions for *Reception & Registration*, 12 FP and 17 FN predictions for *Facilities*, and 6 FP predictions for *Opening hours & accessibility*. The model often confused *Reception & Registration* with *Speaking to the right person* (8 times), *Facilities* with *Availability of employee* (5 times), see the examples below.

- "If you come with a less frequently asked question, it is sometimes difficult to reacht the right department." – assigned *Speaking to the right person* instead of *Speaking to the right person* and *Availability of employee*

- "The location is easily accessible for people with a wheelchair." – assigned *Facilities* instead of *Facilities* and *Availability of employee*

Subtopics within Digital possibilities
The second smallest main topics has three associated subtopics: *Ease of use web & app*, *Functionalities web & app* and *Information provision web & app*. There are 2 FP and 13 FN predictions for *Ease of use web & app*, 5 FP and 8 FN predictions for

Functionalities web & app, and 1 FP and 7 FN predictions for *Information provision web & app*. The model tends to confuse these subtopic labels with one another, as the examples demonstrate below.

- "There should be a digital calendar to facilitate appointment scheduling." – assigned *Functionalities web & app* instead of *Functionalities web & app* and *Ease of use web & app*

- "The explanations for passport photo requirements are different online than in person." – assigned *Functionalities web & app* instead of *Information provision web & app*

Subtopics within Price & quality
The two subtopics *Price & costs* and *Payout & return* belong to this main topic. There are 8 FP and 9 FN predictions for *Price & costs*, and 1 FN for *Price & costs*. Since both topics are strongly underrepresented in the dataset, the model most often missed to assign a positive label to them, as the example shows below.

- "The waiting time was 40 minutes before my turn. Can you use a time estimation tool for all appointments?" – assigned *Waiting time* instead of *Waiting time*, *Price & costs*, *Speed of processing*

4.3.2 Qualitative Analysis

The quantitative error analysis of the best-performing classifier configuration provided insights into the major misclassification patterns. First, we inspected the common classification errors between main topics, then between subtopics. The paragraphs below provide an overview of the identified issues and corresponding recommendations to address them.

Quality of Annotations

A significant problem we identified is the reliance on a rule-based system for annotating the data provided by the company for model training and evaluation. This system assigns labels based on the presence of specific keywords; "online" or "click" trigger the assignment of the *Digital possibilities* topic, "friendly" corresponds to the label *Friendliness*, and the word "quickly" is associated with *Speed of processing*. While the rule-based system provides appropriate labels for the majority of instances, there are several cases where annotations are incorrect. This is mainly because the system fails to capture context and implicit meaning. For example, the feedback "We clicked with the staff member right away" refers to *Employee attitude & behavior* rather than *Digital possibilities*. The rule-based system also struggles to correctly label long feedback instances mentioning a wide range of topics, since it is limited to only assign a maximum of 6 labels per instance.

Inconsistency in the annotations is another concern. The test set contains instances with similar content but different labels. For example, clients often mention that "everything went smoothly" when describing the experience of visiting the governmental

institution. Such feedback statements sometimes received the label *Speed of process*, while in other cases the label *Ease of process*, or a combination of both. When referring to the professional attitude of an employee, we often find the labels *Professionalism* and *Expertise* inconsistently used for annotation.

	Topic	# FP	# Verified FP	# FN	# Verified FN	% Error
1	Making contact with employee	46	24	18	14	40.62
2	Processes	26	10	36	27	40.32
3	Digital possibilities	4	1	17	13	33.33
4	General experience	35	28	48	27	33.72
5	Information provision	67	49	54	37	28.93
6	Employee attitude & behavior	42	25	63	48	30.48
7	Handling	87	53	93	74	29.44
8	No topic found	36	24	89	28	58.4
9	Knowledge & skills of employee	59	42	70	61	20.16
10	Price & quality	8	5	10	3	55.56
11	Physical service provision	40	25	30	13	45.71
12	Waiting time	21	7	12	10	48.48
13	Speaking to the right person	10	4	1	1	54.55
14	Correctness of handling	7	4	11	10	22.22
15	Functionalities web & app	5	3	8	7	23.08
16	Ease of process	25	10	29	22	40.74
17	Reception & Registration	27	14	21	12	45.83
18	Friendliness	23	10	23	14	47.83
19	Quality of information	17	7	19	13	44.44
20	Information provision web & app	1	0	7	5	37.50
21	Clarity of information	38	19	23	14	45.90
22	Solution oriented	47	45	43	37	8.89
23	Availability of employee	22	15	8	5	33.33
24	Price & costs	8	5	9	3	52.94
25	Speed of processing	87	38	88	58	45.14
26	Professionalism	4	2	8	6	33.33
27	Opening hours & accessibility	6	4	0	0	33.33
28	Ease of use web & app	2	0	13	9	40.0
29	Keeping up to date	4	2	9	5	46.15
30	Integrity & fulfilling responsibilities	14	11	22	7	50.0
31	Payout & return	0	0	1	1	0
32	No subtopic found	36	24	89	28	58.4
33	Quality of customer service	7	3	15	10	40.91
34	Facilities	12	3	16	8	60.71
35	Objection & evidence	2	1	3	2	40.0
36	General experience subtopic	35	28	48	27	33.73
37	Efficiency of process	9	5	15	12	29.17
38	Genuine interest	14	8	14	7	46.43
39	Expertise	23	13	28	20	35.29
40	Helpfulness	25	5	46	39	38.03
41	Personal approach	3	3	15	11	22.22
42	Communication	33	24	11	8	27.27
	Sum	**1,017**	**603**	**1,183**	**756**	**38.23**

Table 4.5: Rate of annotation errors by the rule-based system for a sample of the test set. Columns from left to right: Topic (2), Number of False Positives (3), Number of Verified False Positives (4), Number of False Negatives (5), Number of Verified False Negatives (6), Percentage of Annotation Errors (7).

In order have a better understanding of the quality of annotations, we inspected the misclassified test instances, i.e. feedback statements with labels that belong to the category of FP (1,017 labels) or FN (756 labels). Table 4.5 shows the number of labels that were confirmed to be false positives and the ones confirmed to be false negatives based on human revisions. It is apparent that annotations produced by the rule-based system are frequently incorrect and that there are several cases where the fine-tuned BERT model made correct predictions while the rule-based system did not. The rate of erroneous annotations for the inspected sample is 38.23%, ranging between 0% and 60.71% for the individual topic categories.

Using the output of a rule-based system as training data can result in low-quality annotations as this approach fails to capture the subtleties of meaning that may vary depending on the context. Inaccurate annotations negatively impact the training process, which causes the models to learn incorrect patterns. Incorporating human revision in the annotation process could enhance the quality of the training data, thus enabling the models to learn more reliable and truthful patterns. Hiring a team of trained annotators, defining clear annotation guidelines and calculating inter-annotator agreement should be the foundation for curating a reliable dataset.

Overlapping Topics

Another prominent issue is the overlap between the content of certain closely related main topics and subtopics. For example, *Expertise* and *Professionalism* or *Speed of processing* and *Efficiency of process* are semantically close, making it difficulties not only for models but also for human annotators to distinguish them. A practical solution to this problem could be merging closely related topics. This strategy could reduce the number of topic labels and simplify the classification task, potentially leading to an increase in model performance. Even though the company and the client confirmed that they do not intend to merge the current labels, such experiments could be insightful in future work.

Translation and Text Quality

Errors also stem from the quality of feedback instances, particularly since the original Dutch data underwent machine translation. Translating the feedback data via automated tools can lead to information loss, resulting in sentences that are difficult to interpret or classify, even for humans. An example is the instance “Got no solution, depends on the system” annotated with the subtopic label *Solution oriented* due to the occurrence of the keyword “solution”. The translation lacks clarity and might be misleading the model during training. The dataset also contains a small portion of instances written in languages other than English, which can be confusing for the models. Although a colleague at MarketResponse has revised the translations, further proofreading could provide clearer training examples for the models.

Binary Problem Transformation

The transformation of the multi-label classification problem into a set binary problems is a straightforward approach but has a key disadvantage. This approach prevents the model from learning label correlations and understanding the common co-

occurrence of certain classes. For instance, *Clarity of information* and *Quality of information*, or *Friendliness* and *Helpfulness* often appear together, but the model fails to capture this relationship under the current setup. Revisiting the problem transformation approach to account for label correlations could improve the prediction accuracy for complex multi-labeled instances.

Concluding Remarks

A significant limitation of this thesis stems from the nature of the dataset containing silver annotations provided by MarketResponse. The primary focus of this research was to explore the generalization capabilities of machine learning and transfer learning approaches within the context of multi-label topic classification. The data, which was annotated using a rule-based system, facilitated model training and evaluation to examine the capabilities of the different classification methods given the constraints. The methodology of the thesis aligns with the company's interest in assessing the performance of the trained models using the available annotations.

Although re-annotating the dataset was considered to achieve a higher level of data quality, it was not possible within the given time constraints and without established annotation guidelines. The decision to proceed with the existing annotations was made together with the company to ensure that the project was completed within the defined timeline. Despite these constraints, this chapter provides insights into the strengths and weaknesses of the applied machine learning and transfer techniques. It also highlights the importance of using high-quality, human-annotated data. Future work could benefit from incorporating human revision into the annotation process to refine the performance and reliability of the models.

Chapter 5

Conclusion and Discussion

5.1 Concluding Remarks

This study focuses on the supervised task of multi-label topic classification of client feedback within the governance domain, aiming to improve the tools of MarketResponse for assessing customers' experience with a major Dutch governmental institution. Our objective was to compare the performance of state-of-the-art Natural Language Processing algorithms on an imbalanced dataset with a broad range of topic classes.

The methodology for this research encompasses a variety of classification approaches, including traditional machine learning and transfer learning. The dataset provided by the internship company was first anonymized by masking privacy-sensitive information such as names and dates. As for the machine learning approach, we utilized the Support Vector Machines (SVMs) classifier with TF-IDF feature representation and employed two text normalization techniques: lowercasing and stop words removal. For the transfer learning approach, the transformer-based BERT model was chosen, trained with 5 epochs, a maximum sequence length of 128, and the Adam optimizer with StepLR scheduler. We implemented both models with one-step and two-step classification approaches to identify the most optimal model configuration. These experiments included hyper-parameter optimization for both models with each experimental setup. For SVMs, the hyper-parameter tuning involved the parameters *C*, *loss* and *tol*; for BERT, it involved experimenting with different values for *batch size* and *learning rate*.

Additionally, due to the high degree of imbalance in the dataset, various data adaptation and data balancing techniques were tested to explore their impact on the classifiers' performance and gain an understanding of the optimal amount of training data per class. Undersampling did not enhance the classifiers' overall performance but yielded better outcomes for some underrepresented classes. Conversely, utilizing an oversampled dataset by combining the original training set with synthetic data generated by GPT-4 demonstrated a positive effect on both classifiers, underscoring the advantage of increased training data for small classes. The two-step approach, which separates the prediction process into main topics and subsequent subtopics, has proven to be beneficial for both SVMs and BERT. Our findings reveal that BERT outperformed SVMs, achieving a macro-averaged F1-score of 0.66 and a Hamming Loss of 0.027, indicating its superiority in handling this complex multi-label classification task.

5.1.1 Research Questions

The previous section outlined the methodology and main research findings. The paragraphs below aim to provide more detailed answers to the research questions of this thesis.

Research Question: Which approach yields the best performance for multi-label topic classification of client feedback in the governance domain?

We have identified that the fine-tuned BERT model trained on the oversampled dataset with a two-step classification approach and the model configuration *maximum sequence length* 128, *training epochs* 5, *batch size* 8 and *learning rate* 2e-5 exhibits superior performance, reaching a macro-averaged F1-score of 0.66 and Hamming Loss of 0.027. These results highlight the effectiveness of the transformer-based BERT model thanks to its advanced attention mechanism, enabling it to outperform the conventional SVMs classifier. The SVMs, proving to be less powerful for this task with a macro-averaged F1-score of 0.57 on the oversampled training set, offer an advantage in terms of training efficiency. It should be noted that while BERT delivers higher performance results, the feature-based model requires significantly less time and computational resources to train. Training time is an important factor in an industrial setting, thus the choice between models may depend on the balance between performance gains and available computational resources.

Sub-question: How does the performance of classifiers differ between a one-step (main topic labels and subtopic labels combined) and a two-step (first main topic labels, then subtopic labels are classified) classification approach for multi-label topic classification of client feedback in the governance domain?

The classes are hierarchically structured in the provided dataset, since each main topic has one or multiple related subtopics. In this study, we explored both one-step and two-step classification methods in relation to multi-label topic classification. The one-step approach implements the model to predict the presence or absence of all 42 topic labels simultaneously for each feedback instance. On the other hand, the two-step approach refines the process by first predicting the 11 main topic classes and subsequently predicting the 31 corresponding subtopics based on the initial predictions. The latter approach has demonstrated to be advantageous for both classifiers, since it simplifies the classification task by reducing the number of labels the model must consider in each step. Our results underscore this, showing that the two-step approach improved the macro-averaged F1-score by 0.01 for SVMs and by 0.04 for the fine-tuned BERT model. Although the two-step approach increases computational demands and extends processing time, it has proven to be beneficial for handling the complexities of multi-label classification.

Sub-question: What is the impact of data adaptation and data distribution balancing techniques on the performance of classifiers for multi-label topic classification of client feedback in the governance domain?

Two modified datasets have been created to address this sub-question. The undersampled dataset aims to mitigate the influence of overrepresented subtopics by reducing

their representation to the average across the training set. Conversely, the oversampled dataset increases the number of samples for underrepresented subtopics through synthetic data generation using GPT-4. The rationale behind the oversampled dataset is the enhancement of the model's exposure to these rare classes.

The results on the undersampled dataset demonstrated that the SVMs maintained their overall macro-averaged F1-score of 0.56, while the balance between macro precision and recall improved when compared with the original dataset. This suggests that SVMs can operate effectively even with reduced data quantity if the distribution across the classes is more balanced. Notably, there was a moderate improvement in performance metrics for small classes and a slight decrease for large classes. In contrast, the impact of undersampling on the fine-tuned BERT model was less favorable. The macro-averaged F1-score decreased by 0.04, and there was a significant increase in the Hamming Loss from 0.027 to 0.037. This suggests that while undersampling helps balance the dataset, it may reduce the overall amount of data too much, which negatively impacts the performance of BERT. The increase in Hamming Loss indicates a rise in the rate of incorrect label predictions per sample, suggesting that undersampling may not be the most effective strategy for enhancing performance on less frequent classes.

As for the oversampled dataset, we noticed an increase in both models' performance, particularly with the fine-tuned BERT model. Training the models on an enriched dataset containing synthetic instances for underrepresented classes led to an increase in macro-averaged F1-scores. This is especially apparent in the performance of the fine-tuned BERT model, where there was an increase in the macro-averaged precision by 0.02, recall by 0.03 and F1-score by 0.01. The SVMs also benefited from the additional synthetic data, resulting in a 0.01 increase in the macro-averaged F1-score. This confirms the hypothesis that increasing the training volume for infrequent classes can contribute to better model sensitivity and overall accuracy.

Overall, both approaches aimed to address class imbalance and the results suggest that oversampling is particularly beneficial for models with complex architecture like BERT, reaching a macro-averaged precision of 0.69, recall of 0.66 and F1-score of 0.66. The findings highlight the importance of choosing the right data balancing strategy based on the specific characteristics and requirements of the classification model and task at hand.

5.2 Limitations

This research was subject to several constraints that influenced its scope and outcomes. One primary limitation was the time constraint, which restricted the depth of the conducted experiments. Additionally, due to confidentiality agreements with MarketResponse, the use of third-party computational resources with limited access to GPUs, such as Google Colab, was prohibited. Consequently, all experiments were restricted to a local machine, which impacted the speed of model training.

A significant limitation arose from the annotations in the utilized data. The dataset is the output of a rule-based system using taxonomies without human verification, which led to many annotation inconsistencies and errors. Relying on a dataset with incorrect annotations directly impacts the training process and performance of the supervised models. Due to time constraints, defining clear annotation guidelines and employing trained annotators for the annotation process was not feasible. Furthermore, the classification systems developed in this work can only categorize feedback statements using

a predefined list of 11 main topics and 31 subtopics, which poses a limitation in the adaptability to incorporate new topics without retraining. An alternative approach could be the implementation of unsupervised Topic Modeling techniques to address this issue.

The quality of the text data itself presents challenges. The Dutch dataset was machine-translated into English, resulting in the loss of certain linguistic characteristics and spelling errors. This factor could influence the models' understanding and handling of the text data. Employing human translators or proofreaders might help in preserving the integrity and subtleties of the original data. Training models using the original Dutch data could also yield more accurate classification results by retaining the cultural and linguistic nuances lost in translation.

Using GPT-4 for synthetic data generation to address class imbalance also introduces some concerns. While leveraging the generative language model is a cost-efficient method to increase the amount of training data for underrepresented classes, the environmental impact of prompting the model and the model's inherent biases are issues that must be acknowledged. The synthetic data created through zero-shot prompting does not perfectly align with the quality and style of the original dataset. Additionally, the quantity of synthetic data was limited to 1,100 instances – 50 instances per underrepresented subtopic class –, which is proportionately small in contrast with the original dataset comprising 19,529 instances. Increasing the amount of synthetic data with alternative methods could help us achieve the desired balancing effect and lead to further increase in model performance.

5.3 Future Work

Future research directions could extend the scope of this study in several directions, given the previously outlined limitations. The primary step could involve defining detailed annotation guidelines and engaging trained crowd workers or experts in the annotation process. This would help minimize the annotation inconsistencies observed in the current dataset and provide a higher-quality resource for model training and evaluation. Further exploration into other Natural Language Processing tasks, such as sentiment analysis or keyword extraction, could help the company gain a better understanding of the customers' experience with the governmental institution. This would provide deeper insights into the clients' sentiment, expectations and concerns.

Additionally, the experiments could be adapted to directly address the Dutch dataset without translating it, for example by fine-tuning a Dutch pre-trained transformer-based model, BERTje[1] (De Vries et al., 2019), or adapting the project to a multilingual context using a model like the multilingual BERT[2], which was pre-trained on multilingual datasets. This would allow for the direct application of the Dutch dataset, preserving the linguistic nuances lost in translation. The scope of the implemented machine learning and transfer learning approaches could also be expanded. While this thesis only employed SVMs and a fine-tuned BERT model, future projects could experiment with other algorithms, such as Logistic Regression or Naive Bayes, and explore different transformer-based architectures like RoBERTa (Liu et al., 2019). Moreover, adapting the classification strategy to directly handle the multi-label dataset with adaptive classifiers or experimenting with different problem transformation methods could

[1] https://huggingface.co/GroNLP/bert-base-dutch-cased
[2] https://huggingface.co/google-bert/bert-base-multilingual-uncased

offer insights into their respective effectiveness.

Regarding the one-step classification approach, a modification could be considered where the models are trained to predict only the subtopics for each instance, and the main topic predictions are inferred from the subtopic predictions. This could decrease the number of topic labels from 42 to 31, potentially leading to improved classification accuracy. Additionally, merging similar subtopic classes or excluding those with less than 50 instances could simplify the classification task and enhance model performance.

In terms of data manipulation techniques, future research could implement simultaneous undersampling and oversampling to create a more balanced training set. Exploring different data augmentation methods, such as back-translation, lexical substitution or text element mixing, could further enhance the robustness of the models (Chaudhary, 2020). Adjustments to the prediction threshold used by the classifiers could also be tested to observe if lower or higher values yield performance improvements. Exploring alternative features for SVMs, such as incorporating pre-trained word embeddings or leveraging morphosyntactic features like part-of-speech tags, could provide more detailed text representation. For the fine-tuned BERT model, expanding the hyper-parameter tuning to include additional parameters, such as dropout rate, and increasing the training epochs could further optimize the model's performance.

Lastly, MarketResponse could link the results of multi-label topic classification and sentiment analysis with other available CX metrics – such as Key Performance Indicators (KPIs) or Customer Satisfaction Score (CSAT) – to obtain a more holistic view of the customer's journey. This integration could help the Dutch governmental institution better understand and address the needs of its clientele, ultimately leading to improved service provision and higher level of customer satisfaction.

Appendix A

Results with Conventional Machine Learning

A.1 Original Training Set

A.1.1 One-step Approach

	Topic	Precision	Recall	F1-score	Support
1	Making contact with employee	0.78	0.60	0.68	144
2	Processes	0.77	0.62	0.69	165
3	Digital possibilities	0.78	0.40	0.53	35
4	General experience	0.83	0.58	0.68	165
5	Information provision	0.85	0.69	0.76	313
6	Employee attitude & behavior	0.91	0.81	0.85	617
7	Handling	0.79	0.70	0.74	477
8	No topic found	0.78	0.29	0.43	173
9	Knowledge & skills of employee	0.73	0.54	0.62	284
10	Price & quality	0.71	0.33	0.45	15
11	Physical service provision	0.67	0.53	0.59	142
12	Waiting time	0.80	0.65	0.72	99
13	Speaking to the right person	0.62	0.36	0.46	22
14	Correctness of handling	0.71	0.23	0.34	22
15	Functionalities web & app	1.00	0.41	0.58	17
16	Ease of process	0.73	0.56	0.63	106
17	Reception & Registration	0.79	0.60	0.68	97
18	Friendliness	0.94	0.88	0.91	454
19	Quality of information	0.91	0.46	0.61	67
20	Information provision web & app	0.67	0.33	0.44	12
21	Clarity of information	0.93	0.77	0.84	197
22	Solution oriented	0.67	0.44	0.53	181
23	Availability of employee	0.93	0.45	0.61	31
24	Price & costs	0.71	0.36	0.48	14
25	Speed of processing	0.81	0.72	0.76	467
26	Professionalism	0.73	0.61	0.67	36
27	Opening hours & accessibility	0.00	0.00	0.00	1
28	Ease of use web & app	0.80	0.22	0.35	18
29	Keeping up to date	0.50	0.07	0.12	15
30	Integrity & fulfilling responsibilities	0.80	0.30	0.43	54
31	Payout & return	0.00	0.00	0.00	1
32	No subtopic found	0.78	0.29	0.43	173
33	Quality of customer service	0.71	0.47	0.56	47
34	Facilities	0.52	0.31	0.38	49
35	Objection & evidence	0.00	0.00	0.00	4
36	General experience subtopic	0.83	0.58	0.68	165
37	Efficiency of process	0.95	0.62	0.75	66
38	Genuine interest	0.89	0.67	0.76	81
39	Expertise	0.83	0.51	0.63	75
40	Helpfulness	0.84	0.55	0.67	172
41	Personal approach	0.79	0.48	0.59	23
42	Communication	0.70	0.43	0.53	53
	macro avg	**0.73**	**0.46**	**0.55**	5349
	weighted avg	0.82	0.62	0.70	5349
	hamming loss			0.033	5349

Table A.1: Results overview on the original dataset using one-step classification with SVMs after hyper-parameter optimization.

A.1.2 Two-step Approach

	Topic	Precision	Recall	F1-score	Support
1	Making contact with employee	0.78	0.60	0.68	144
2	Processes	0.77	0.62	0.69	165
3	Digital possibilities	0.78	0.40	0.53	35
4	General experience	0.83	0.58	0.68	165
5	Information provision	0.85	0.69	0.76	313
6	Employee attitude & behavior	0.91	0.81	0.85	617
7	Handling	0.79	0.70	0.74	477
8	No topic found	0.78	0.29	0.43	173
9	Knowledge & skills of employee	0.72	0.54	0.62	284
10	Price & quality	0.71	0.33	0.45	15
11	Physical service provision	0.67	0.53	0.59	142
12	Waiting time	0.76	0.68	0.72	99
13	Speaking to the right person	0.69	0.41	0.51	22
14	Correctness of handling	0.86	0.27	0.41	22
15	Functionalities web & app	0.86	0.35	0.50	17
16	Ease of process	0.69	0.58	0.63	106
17	Reception & Registration	0.73	0.59	0.65	97
18	Friendliness	0.91	0.90	0.90	454
19	Quality of information	0.85	0.61	0.71	67
20	Information provision web & app	0.62	0.42	0.50	12
21	Clarity of information	0.89	0.78	0.83	197
22	Solution oriented	0.67	0.47	0.55	181
23	Availability of employee	0.80	0.39	0.52	31
24	Price & costs	0.71	0.36	0.48	14
25	Speed of processing	0.79	0.72	0.75	467
26	Professionalism	0.70	0.72	0.71	36
27	Opening hours & accessibility	0.00	0.00	0.00	1
28	Ease of use web & app	0.75	0.17	0.27	18
29	Keeping up to date	0.60	0.20	0.30	15
30	Integrity & fulfilling responsibilities	0.75	0.39	0.51	54
31	Payout & return	0.00	0.00	0.00	1
32	No subtopic found	0.78	0.29	0.43	173
33	Quality of customer service	0.78	0.53	0.63	47
34	Facilities	0.53	0.35	0.42	49
35	Objection & evidence	0.00	0.00	0.00	4
36	General experience subtopic	0.83	0.58	0.68	165
37	Efficiency of process	0.83	0.61	0.70	66
38	Genuine interest	0.81	0.69	0.75	81
39	Expertise	0.75	0.57	0.65	75
40	Helpfulness	0.75	0.58	0.66	172
41	Personal approach	0.81	0.57	0.67	23
42	Communication	0.69	0.55	0.61	53
	macro avg	**0.71**	**0.49**	**0.56**	5349
	weighted avg	0.80	0.64	0.70	5349
	hamming loss			0.034	5349

Table A.2: Results overview on the original dataset using two-step classification with SVMs after hyper-parameter optimization.

A.2 Undersampled Training Set

A.2.1 Two-step Approach

	Topic	Precision	Recall	F1-score	Support
1	Making contact with employee	0.69	0.69	0.69	144
2	Processes	0.74	0.69	0.71	165
3	Digital possibilities	0.75	0.51	0.61	35
4	General experience	0.79	0.55	0.65	165
5	Information provision	0.80	0.72	0.76	313
6	Employee attitude & behavior	0.90	0.80	0.85	617
7	Handling	0.84	0.43	0.57	477
8	No topic found	0.55	0.27	0.36	173
9	Knowledge & skills of employee	0.67	0.54	0.60	284
10	Price & quality	0.62	0.33	0.43	15
11	Physical service provision	0.59	0.66	0.62	142
12	Waiting time	0.72	0.77	0.74	99
13	Speaking to the right person	0.53	0.45	0.49	22
14	Correctness of handling	0.73	0.50	0.59	22
15	Functionalities web & app	0.70	0.41	0.52	17
16	Ease of process	0.66	0.65	0.65	106
17	Reception & Registration	0.61	0.71	0.66	97
18	Friendliness	0.94	0.85	0.89	454
19	Quality of information	0.82	0.67	0.74	67
20	Information provision web & app	0.67	0.50	0.57	12
21	Clarity of information	0.85	0.79	0.82	197
22	Solution oriented	0.67	0.44	0.53	181
23	Availability of employee	0.72	0.42	0.53	31
24	Price & costs	0.62	0.36	0.45	14
25	Speed of processing	0.84	0.43	0.57	467
26	Professionalism	0.69	0.75	0.72	36
27	Opening hours & accessibility	0.00	0.00	0.00	1
28	Ease of use web & app	0.83	0.28	0.42	18
29	Keeping up to date	0.50	0.13	0.21	15
30	Integrity & fulfilling responsibilities	0.69	0.44	0.54	54
31	Payout & return	0.00	0.00	0.00	1
32	No subtopic found	0.55	0.27	0.36	173
33	Quality of customer service	0.52	0.66	0.58	47
34	Facilities	0.51	0.45	0.48	49
35	Objection & evidence	0.00	0.00	0.00	4
36	General experience subtopic	0.79	0.55	0.65	165
37	Efficiency of process	0.82	0.64	0.72	66
38	Genuine interest	0.81	0.73	0.77	81
39	Expertise	0.66	0.61	0.63	75
40	Helpfulness	0.73	0.66	0.69	172
41	Personal approach	0.81	0.57	0.67	23
42	Communication	0.56	0.57	0.56	53
	macro avg	**0.65**	**0.51**	**0.56**	5349
	weighted avg	0.77	0.60	0.67	5349
	hamming loss			0.038	5349

Table A.3: Results overview on the undersampled dataset using two-step classification with SVMs after hyper-parameter optimization.

A.3 Oversampled Training Set

A.3.1 Two-step Approach

	Topic	Precision	Recall	F1-score	Support
1	Making contact with employee	0.77	0.60	0.67	144
2	Processes	0.80	0.61	0.69	165
3	Digital possibilities	0.79	0.43	0.56	35
4	General experience	0.81	0.57	0.67	165
5	Information provision	0.87	0.68	0.76	313
6	Employee attitude & behavior	0.90	0.81	0.85	617
7	Handling	0.80	0.69	0.74	477
8	No topic found	0.82	0.31	0.45	173
9	Knowledge & skills of employee	0.72	0.54	0.62	284
10	Price & quality	0.71	0.33	0.45	15
11	Physical service provision	0.68	0.54	0.60	142
12	Waiting time	0.76	0.66	0.70	99
13	Speaking to the right person	0.71	0.45	0.56	22
14	Correctness of handling	0.78	0.32	0.45	22
15	Functionalities web & app	0.88	0.41	0.56	17
16	Ease of process	0.71	0.57	0.63	106
17	Reception & Registration	0.73	0.61	0.66	97
18	Friendliness	0.90	0.90	0.90	454
19	Quality of information	0.89	0.61	0.73	67
20	Information provision web & app	0.67	0.50	0.57	12
21	Clarity of information	0.90	0.78	0.83	197
22	Solution oriented	0.68	0.47	0.56	181
23	Availability of employee	0.80	0.39	0.52	31
24	Price & costs	0.71	0.36	0.48	14
25	Speed of processing	0.80	0.70	0.75	467
26	Professionalism	0.70	0.72	0.71	36
27	Opening hours & accessibility	0.00	0.00	0.00	1
28	Ease of use web & app	0.60	0.17	0.26	18
29	Keeping up to date	0.60	0.20	0.30	15
30	Integrity & fulfilling responsibilities	0.79	0.35	0.49	54
31	Payout & return	0.00	0.00	0.00	1
32	No subtopic found	0.82	0.31	0.45	173
33	Quality of customer service	0.76	0.53	0.62	47
34	Facilities	0.56	0.37	0.44	49
35	Objection & evidence	0.00	0.00	0.00	4
36	General experience subtopic	0.81	0.57	0.67	165
37	Efficiency of process	0.87	0.59	0.70	66
38	Genuine interest	0.82	0.69	0.75	81
39	Expertise	0.74	0.57	0.65	75
40	Helpfulness	0.75	0.59	0.66	172
41	Personal approach	0.81	0.57	0.67	23
42	Communication	0.69	0.51	0.59	53
	macro avg	**0.71**	**0.49**	**0.57**	5349
	weighted avg	0.81	0.63	0.70	5349
	hamming loss			0.033	5349

Table A.4: Results overview on the oversampled dataset using two-step classification with SVMs after hyper-parameter optimization.

Appendix B

Results with Transfer Learning

B.1 Original Training Set

B.1.1 One-step Approach

	Topic	Precision	Recall	F1-score	Support
1	Making contact with employee	0.81	0.76	0.78	144
2	Processes	0.76	0.86	0.81	165
3	Digital possibilities	0.69	0.71	0.70	35
4	General experience	0.84	0.62	0.72	165
5	Information provision	0.80	0.83	0.82	313
6	Employee attitude & behavior	0.92	0.91	0.92	617
7	Handling	0.82	0.85	0.84	477
8	No topic found	0.69	0.56	0.62	173
9	Knowledge & skills of employee	0.74	0.77	0.76	284
10	Price & quality	1.00	0.20	0.33	15
11	Physical service provision	0.77	0.63	0.69	142
12	Waiting time	0.84	0.76	0.80	99
13	Speaking to the right person	0.59	0.45	0.51	22
14	Correctness of handling	0.75	0.14	0.23	22
15	Functionalities web & app	0.91	0.59	0.71	17
16	Ease of process	0.71	0.82	0.76	106
17	Reception & Registration	0.75	0.61	0.67	97
18	Friendliness	0.92	0.95	0.94	454
19	Quality of information	0.83	0.64	0.72	67
20	Information provision web & app	0.75	0.50	0.60	12
21	Clarity of information	0.89	0.87	0.88	197
22	Solution oriented	0.73	0.73	0.73	181
23	Availability of employee	0.75	0.48	0.59	31
24	Price & costs	1.00	0.21	0.35	14
25	Speed of processing	0.83	0.85	0.84	467
26	Professionalism	0.97	0.83	0.90	36
27	Opening hours & accessibility	0.00	0.00	0.00	1
28	Ease of use web & app	0.50	0.22	0.31	18
29	Keeping up to date	0.00	0.00	0.00	15
30	Integrity & fulfilling responsibilities	0.79	0.41	0.54	54
31	Payout & return	0.00	0.00	0.00	1
32	No subtopic found	0.69	0.56	0.62	173
33	Quality of customer service	0.80	0.68	0.74	47
34	Facilities	0.76	0.45	0.56	49
35	Objection & evidence	0.00	0.00	0.00	4
36	General experience subtopic	0.84	0.62	0.72	165
37	Efficiency of process	0.88	0.76	0.81	66
38	Genuine interest	0.86	0.78	0.82	81
39	Expertise	0.70	0.60	0.65	75
40	Helpfulness	0.86	0.77	0.82	172
41	Personal approach	0.00	0.00	0.00	23
42	Communication	0.71	0.68	0.69	53
	macro avg	**0.70**	**0.56**	**0.61**	5349
	weighted avg	0.81	0.77	0.78	5349
	hamming loss			0.026	5349

Table B.1: Results overview on the original dataset using one-step classification with BERT after hyper-parameter optimization.

B.1.2 Two-step Approach

	Topic	Precision	Recall	F1-score	Support
1	Making contact with employee	0.73	0.82	0.77	144
2	Processes	0.80	0.80	0.80	165
3	Digital possibilities	0.73	0.77	0.75	35
4	General experience	0.82	0.67	0.74	165
5	Information provision	0.78	0.81	0.79	313
6	Employee attitude & behavior	0.92	0.91	0.91	617
7	Handling	0.84	0.77	0.81	477
8	No topic found	0.67	0.53	0.59	173
9	Knowledge & skills of employee	0.74	0.77	0.76	284
10	Price & quality	0.50	0.20	0.29	15
11	Physical service provision	0.80	0.72	0.76	142
12	Waiting time	0.78	0.90	0.84	99
13	Speaking to the right person	0.58	0.64	0.61	22
14	Correctness of handling	0.56	0.45	0.50	22
15	Functionalities web & app	0.59	0.76	0.67	17
16	Ease of process	0.73	0.76	0.75	106
17	Reception & Registration	0.80	0.76	0.78	97
18	Friendliness	0.94	0.96	0.95	454
19	Quality of information	0.79	0.78	0.78	67
20	Information provision web & app	0.89	0.67	0.76	12
21	Clarity of information	0.81	0.89	0.85	197
22	Solution oriented	0.73	0.80	0.76	181
23	Availability of employee	0.63	0.61	0.62	31
24	Price & costs	0.50	0.21	0.30	14
25	Speed of processing	0.84	0.78	0.81	467
26	Professionalism	0.75	0.83	0.79	36
27	Opening hours & accessibility	0.00	0.00	0.00	1
28	Ease of use web & app	0.50	0.44	0.47	18
29	Keeping up to date	0.44	0.27	0.33	15
30	Integrity & fulfilling responsibilities	0.63	0.59	0.61	54
31	Payout & return	0.00	0.00	0.00	1
32	No subtopic found	0.67	0.53	0.59	173
33	Quality of customer service	0.66	0.66	0.66	47
34	Facilities	0.70	0.65	0.67	49
35	Objection & evidence	0.00	0.00	0.00	4
36	General experience subtopic	0.82	0.67	0.74	165
37	Efficiency of process	0.79	0.76	0.78	66
38	Genuine interest	0.82	0.84	0.83	81
39	Expertise	0.69	0.56	0.62	75
40	Helpfulness	0.85	0.80	0.82	172
41	Personal approach	0.79	0.48	0.59	23
42	Communication	0.64	0.70	0.67	53
	macro avg	**0.67**	**0.63**	**0.65**	5349
	weighted avg	0.80	0.78	0.79	5349
	hamming loss			0.027	5349

Table B.2: Results overview on the original dataset using two-step classification with BERT after hyper-parameter optimization.

B.2 Undersampled Training Set

B.2.1 Two-step Approach

	Topic	Precision	Recall	F1-score	Support
1	Making contact with employee	0.74	0.78	0.76	144
2	Processes	0.69	0.84	0.75	165
3	Digital possibilities	0.60	0.80	0.68	35
4	General experience	0.67	0.72	0.69	165
5	Information provision	0.74	0.82	0.78	313
6	Employee attitude & behavior	0.93	0.85	0.89	617
7	Handling	0.87	0.54	0.67	477
8	No topic found	0.49	0.57	0.53	173
9	Knowledge & skills of employee	0.56	0.77	0.65	284
10	Price & quality	0.38	0.33	0.36	15
11	Physical service provision	0.66	0.67	0.66	142
12	Waiting time	0.76	0.83	0.79	99
13	Speaking to the right person	0.64	0.73	0.68	22
14	Correctness of handling	0.52	0.55	0.53	22
15	Functionalities web & app	0.52	0.76	0.62	17
16	Ease of process	0.62	0.84	0.71	106
17	Reception & Registration	0.64	0.67	0.65	97
18	Friendliness	0.96	0.87	0.91	454
19	Quality of information	0.56	0.85	0.67	67
20	Information provision web & app	0.71	0.83	0.77	12
21	Clarity of information	0.82	0.84	0.83	197
22	Solution oriented	0.56	0.76	0.65	181
23	Availability of employee	0.68	0.55	0.61	31
24	Price & costs	0.38	0.36	0.37	14
25	Speed of processing	0.88	0.54	0.67	467
26	Professionalism	0.77	0.83	0.80	36
27	Opening hours & accessibility	0.00	0.00	0.00	1
28	Ease of use web & app	0.35	0.44	0.39	18
29	Keeping up to date	0.40	0.53	0.46	15
30	Integrity & fulfilling responsibilities	0.63	0.61	0.62	54
31	Payout & return	0.00	0.00	0.00	1
32	No subtopic found	0.49	0.57	0.53	173
33	Quality of customer service	0.49	0.77	0.60	47
34	Facilities	0.67	0.63	0.65	49
35	Objection & evidence	0.00	0.00	0.00	4
36	General experience subtopic	0.67	0.72	0.69	165
37	Efficiency of process	0.74	0.76	0.75	66
38	Genuine interest	0.70	0.90	0.79	81
39	Expertise	0.35	0.75	0.48	75
40	Helpfulness	0.80	0.81	0.80	172
41	Personal approach	0.70	0.30	0.42	23
42	Communication	0.58	0.72	0.64	53
	macro avg	**0.59**	**0.64**	**0.61**	5349
	weighted avg	0.74	0.73	0.72	5349
	hamming loss			0.037	5349

Table B.3: Results overview on the undersampled dataset using two-step classification with BERT after hyper-parameter optimization.

B.3 Oversampled Training Set

B.3.1 Two-step Approach

	Topic	Precision	Recall	F1-score	Support
1	Making contact with employee	0.73	0.88	0.80	144
2	Processes	0.83	0.78	0.81	165
3	Digital possibilities	0.82	0.51	0.63	35
4	General experience	0.77	0.71	0.74	165
5	Information provision	0.79	0.83	0.81	313
6	Employee attitude & behavior	0.93	0.90	0.91	617
7	Handling	0.82	0.81	0.81	477
8	No topic found	0.70	0.49	0.57	173
9	Knowledge & skills of employee	0.78	0.75	0.77	284
10	Price & quality	0.38	0.33	0.36	15
11	Physical service provision	0.74	0.79	0.76	142
12	Waiting time	0.81	0.88	0.84	99
13	Speaking to the right person	0.68	0.95	0.79	22
14	Correctness of handling	0.61	0.50	0.55	22
15	Functionalities web & app	0.64	0.53	0.58	17
16	Ease of process	0.75	0.73	0.74	106
17	Reception & Registration	0.74	0.78	0.76	97
18	Friendliness	0.95	0.95	0.95	454
19	Quality of information	0.74	0.72	0.73	67
20	Information provision web & app	0.83	0.42	0.56	12
21	Clarity of information	0.82	0.88	0.85	197
22	Solution oriented	0.75	0.76	0.75	181
23	Availability of employee	0.51	0.74	0.61	31
24	Price & costs	0.38	0.36	0.37	14
25	Speed of processing	0.81	0.81	0.81	467
26	Professionalism	0.88	0.78	0.82	36
27	Opening hours & accessibility	0.14	1.00	0.25	1
28	Ease of use web & app	0.71	0.28	0.40	18
29	Keeping up to date	0.60	0.40	0.48	15
30	Integrity & fulfilling responsibilities	0.70	0.59	0.64	54
31	Payout & return	0.00	0.00	0.00	1
32	No subtopic found	0.70	0.49	0.57	173
33	Quality of customer service	0.82	0.68	0.74	47
34	Facilities	0.73	0.67	0.70	49
35	Objection & evidence	0.33	0.25	0.29	4
36	General experience subtopic	0.77	0.71	0.74	165
37	Efficiency of process	0.85	0.77	0.81	66
38	Genuine interest	0.83	0.83	0.83	81
39	Expertise	0.67	0.63	0.65	75
40	Helpfulness	0.83	0.73	0.78	172
41	Personal approach	0.73	0.35	0.47	23
42	Communication	0.56	0.79	0.66	53
	macro avg	**0.69**	**0.66**	**0.66**	5349
	weighted avg	0.80	0.78	0.79	5349
	hamming loss			0.027	5349

Table B.4: Results overview on the oversampled dataset using two-step classification with BERT after hyper-parameter optimization.

Bibliography

J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, et al. GPT-4 Technical Report. *arXiv preprint arXiv:2303.08774*, 2023.

A. Adhikari, A. Ram, R. Tang, and J. Lin. DocBERT: BERT for Document Classification. *arXiv preprint arXiv:1904.08398*, 2019.

C. C. Aggarwal and C. Zhai. A survey of text classification algorithms. *Mining text data*, pages 163–222, 2012.

N. A.-R. Al-Serw. Undersampling and oversampling: An old and a new approach. *Analytics Vidhya*, 2021.

J. Alammar. The illustrated transformer. `https://jalammar.github.io/illustrated-transformer/`, 2018a.

J. Alammar. The illustrated BERT, ELMo, and co. (How NLP Cracked Transfer Learning). `https://jalammar.github.io/illustrated-bert/`, 2018b.

I. Ameer, N. Bölücü, M. H. F. Siddiqui, B. Can, G. Sidorov, and A. Gelbukh. Multi-label emotion classification in texts using transfer learning. *Expert Systems with Applications*, 213:118534, 2023.

A. Anaby-Tavor, B. Carmeli, E. Goldbraich, A. Kantor, G. Kour, S. Shlomov, N. Tepper, and N. Zwerdling. Not Enough Data? Deep Learning to the Rescue! *arXiv preprint arXiv:1911.03118*, 2019.

E. M. Bender and B. Friedman. Data Statements for Natural Language Processing: Toward Mitigating System Bias and Enabling Better Science. *Transactions of the Association for Computational Linguistics*, 6:587–604, 2018.

J. Bogatinovski, L. Todorovski, S. Džeroski, and D. Kocev. Comprehensive comparative study of multi-label classification methods. *Expert Systems with Applications*, 203: 117215, 2022.

M. R. Boutell, J. Luo, X. Shen, and C. M. Brown. Learning multi-label scene classification. *Pattern recognition*, 37(9):1757–1771, 2004.

J. Cervantes, F. Garcia-Lamont, L. Rodríguez-Mazahua, and A. Lopez. A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing*, 408:189–215, 2020.

W.-C. Chang, H.-F. Yu, K. Zhong, Y. Yang, and I. S. Dhillon. Taming pretrained transformers for extreme multi-label text classification. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 3163–3171, 2020.

F. Charte, A. J. Rivera, M. J. del Jesus, and F. Herrera. Addressing imbalance in multilabel classification: Measures and random resampling algorithms. *Neurocomputing*, 163:3–16, 2015.

Z. Chase, N. Genain, and O. Karniol-Tambour. Learning Multi-Label Topic Classification of News Articles. 2014.

A. Chaudhary. A Visual Survey of Data Augmentation in NLP, 2020. `https://amitness.com/2020/05/data-augmentation-for-nlp`.

V. K. Chauhan, K. Dahiya, and A. Sharma. Problem formulations and solvers in linear SVM: a review. *Artificial Intelligence Review*, 52(2):803–855, 2019.

N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, 16: 321–357, 2002.

G. Chen, D. Ye, Z. Xing, J. Chen, and E. Cambria. Ensemble application of convolutional and recurrent neural networks for multi-label text categorization. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2377–2383. IEEE, 2017.

K. Chen, B.-L. Lu, and J. T. Kwok. Efficient classification of multi-label and imbalanced data using min-max modular classifiers. In *The 2006 IEEE International Joint Conference on Neural Network Proceedings*, pages 1770–1775. IEEE, 2006.

W.-J. Chen, Y.-H. Shao, C.-N. Li, and N.-Y. Deng. MLTSVM: A novel twin support vector machine to multi-label learning. *Pattern Recognition*, 52:61–74, 2016.

K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv preprint arXiv:1406.1078*, 2014.

K. Chowdhary. Natural Language Pocessing. *Fundamentals of Artificial Intelligence*, pages 603–649, 2020.

C. Cortes and V. Vapnik. Support-vector networks. *Machine learning*, 20:273–297, 1995.

S. M. H. Dadgar, M. S. Araghi, and M. M. Farahani. A novel text mining approach based on TF-IDF and Support Vector Machine for news classification. In *2016 IEEE International Conference on Engineering and Technology (ICETECH)*, pages 112–116. IEEE, 2016.

A. C. de Carvalho and A. A. Freitas. A Tutorial on Multi-label Classification Techniques. *Foundations of Computational Intelligence*, 5:177–195, 2009.

O. De Clercq, L. De Bruyne, and V. Hoste. News topic classification as a first step towards diverse news recommendation. *Computational Linguistics in the Netherlands Journal*, 10:37–55, 2020.

W. De Vries, A. van Cranenburgh, A. Bisazza, T. Caselli, G. van Noord, and M. Nissim. BERTje: A Dutch BERT Model. *arXiv preprint arXiv:1912.09582*, 2019.

J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv preprint arXiv:1810.04805*, 2018.

K. Doshi. Transformers Explained Visually (Part 3): Multi-head attention, Deep Dive. *Towards Data Science*, 2021.

S. Dumais and H. Chen. Hierarchical Classification of Web Content. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 256–263, 2000.

K. Ethayarajh. How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. *arXiv preprint arXiv:1909.00512*, 2019.

F. Figueiredo, L. Rocha, T. Couto, T. Salles, M. A. Gonçalves, and W. Meira Jr. Word co-occurrence features for text classification. *Information Systems*, 36(5):843–858, 2011.

R. Gandhi. Support Vector Machine — Introduction to Machine Learning Algorithms. *Towards Data Science*, 2018.

K. Garcia and L. Berton. Topic detection and sentiment analysis in Twitter content related to COVID-19 from Brazil and the USA. *Applied soft computing*, 101:107057, 2021.

S. Godbole and S. Sarawagi. Discriminative methods for multi-labeled classification. In *Pacific-Asia conference on knowledge discovery and data mining*, pages 22–30. Springer, 2004.

T. Gonçalves and P. Quaresma. A preliminary approach to the multilabel classification problem of portuguese juridical documents. In *Portuguese Conference on Artificial Intelligence*, pages 435–444. Springer, 2003.

S. R. Gunn et al. Support Vector Machines for Classification and Regression. *ISIS technical report*, 14(1):5–16, 1998.

M. A. Hadi and F. H. Fard. Evaluating pre-trained models for user feedback analysis in software engineering: A study on classification of app-reviews. *Empirical Software Engineering*, 28(4):88, 2023.

E.-H. Han, G. Karypis, and V. Kumar. Text Categorization Using Weight Adjusted k-Nearest Neighbor Classification. In *Advances in Knowledge Discovery and Data Mining*, pages 53–65. Springer, 2001.

Z. S. Harris. Distributional structure. *Word*, 10(2-3):146–162, 1954.

F. Herrera, F. Charte, A. J. Rivera, and M. J. del Jesus. *Multilabel classification*. Springer, 2016.

J. Howard and S. Ruder. Universal Language Model Fine-tuning for Text Classification. *arXiv preprint arXiv:1801.06146*, 2018.

M. Jaderberg, K. Simonyan, A. Vedaldi, and A. Zisserman. Synthetic Data and Artificial Neural Networks for Natural Scene Text Recognition. *arXiv preprint arXiv:1406.2227*, 2014.

H. Jelodar, Y. Wang, R. Orji, and S. Huang. Deep sentiment classification and topic discovery on novel coronavirus or COVID-19 online discussions: NLP using LSTM recurrent neural network approach. *IEEE Journal of Biomedical and Health Informatics*, 24(10):2733–2742, 2020.

T. Joachims. Text categorization with Support Vector Machines: Learning with many relevant features. In *European conference on machine learning*, pages 137–142. Springer, 1998.

D. Jurafsky and J. H. Martin. *Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition*. Pearson Prentice Hall, 2nd edition, 2009.

K. S. Kalyan, A. Rajasekharan, and S. Sangeetha. AMMUS: A Survey of Transformer-based Pretrained Models in Natural Language Processing. *arXiv preprint arXiv:2108.05542*, 2021.

S. Khan. BERT, RoBERTa, DistilBERT, XLNet — which one to use. *Towards Data Science*, 2021.

D. P. Kingma and J. Ba. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 2014.

K. Kowsari, K. Jafari Meimandi, M. Heidarysafa, S. Mendu, L. Barnes, and D. Brown. Text Classification Algorithms: A Survey. *Information*, 10(4):150, 2019.

M. Kubát and S. Matwin. Addressing the Curse of Imbalanced Training sets: One-Sided Selection. In *International Conference on Machine Learning*, pages 179–186, 1997.

V. Kumar, A. Choudhary, and E. Cho. Data Augmentation using Pre-trained Transformer Models. *arXiv preprint arXiv:2003.02245*, 2020.

K. Lee, D. Palsetia, R. Narayanan, M. M. A. Patwary, A. Agrawal, and A. Choudhary. Twitter Trending Topic Classification. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 251–258, 2011.

K. N. Lemon and P. C. Verhoef. Understanding Customer Experience Throughout the Customer Journey. *Journal of Marketing*, 80(6):69–96, 2016.

L. Lenc and P. Král. Word Embeddings for Multi-label Document Classification. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 431–437, 2017.

S. Li and J. Ou. Multi-label Classification of Research Papers Using Multi-label K-Nearest Neighbour Algorithm. In *Journal of Physics: Conference Series*, volume 1994. IOP Publishing, 2021.

X. Li, M. Cui, J. Li, R. Bai, Z. Lu, and U. Aickelin. A hybrid medical text classification framework: Integrating attentive rule construction and neural network. *Neurocomputing*, 443:345–355, 2021.

Z. Li, H. Zhu, Z. Lu, and M. Yin. Synthetic Data Generation with Large Language Models for Text Classification: Potential and Limitations. *arXiv preprint arXiv:2310.07849*, 2023.

X. Liu and C. Wang. An Empirical Study on Hyperparameter Optimization for Fine-tuning Pre-trained Language Models. *arXiv preprint arXiv:2106.09204*, 2021.

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. *arXiv preprint arXiv:1907.11692*, 2019.

K. Maharana, S. Mondal, and B. Nemade. A review: Data pre-processing and data augmentation techniques. *Global Transitions Proceedings*, 3(1):91–99, 2022.

A. Malte and P. Ratadiya. Evolution of transfer learning in natural language processing. *arXiv preprint arXiv:1910.07370*, 2019.

S. Mehra, R. Louka, and Y. Zhang. ESGBERT: Language Model to Help with Classification Tasks Related to Companies' Environmental, Social, and Governance Practices. *arXiv preprint arXiv:2203.16788*, 2022.

T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*, 2013.

A. Mountassir, H. Benbrahim, and I. Berrada. Addressing the Problem of Unbalanced Data Sets in Sentiment Analysis. In *International Conference on Knowledge Discovery and Information Retrieval*, volume 2, pages 306–311. SCITEPRESS, 2012.

B. Noori. Classification of Customer Reviews Using Machine Learning Algorithms. *Applied Artificial Intelligence*, 35(8):567–588, 2021.

T. Nugent, N. Stelea, and J. L. Leidner. Detecting Environmental, Social and Governance (ESG) Topics Using Domain-Specific Language Models and Data Augmentation. In *Flexible Query Answering Systems*, pages 157–169. Springer, 2021.

S. J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

J. Pennington, R. Socher, and C. D. Manning. GlovVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, 2014.

T. Pranckevičius and V. Marcinkevičius. Comparison of Naïve Bayes, Random Forest, Decision Tree, Support Vector Machines, and Logistic Regression Classifiers for Text Reviews Classification. *Baltic Journal of Modern Computing*, 5(2):221, 2017.

A. Rane and A. Kumar. Sentiment Classification System of Twitter Data for US Airline Service Analysis. In *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, volume 1, pages 769–773. IEEE, 2018.

S. Ruder, M. E. Peters, S. Swayamdipta, and T. Wolf. Transfer Learning in Natural Language Processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pages 15–18, 2019.

V. Sanh, L. Debut, J. Chaumond, and T. Wolf. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.

K. Sechidis, G. Tsoumakas, and I. Vlahavas. On the Stratification of Multi-label Data. In *Machine Learning and Knowledge Discovery in Databases*, pages 145–158. Springer, 2011.

P. C. Sen, M. Hajra, and M. Ghosh. Supervised Classification Algorithms in Machine Learning: A Survey and Review. In *Emerging Technology in Modelling and Graphics: Proceedings of IEM Graph 2018*, pages 99–111. Springer, 2020.

K. Shah, H. Patel, D. Sanghvi, and M. Shah. A Comparative Analysis of Logistic Regression, Random Forest and KNN Models for the Text Classification. *Augmented Human Research*, 5:1–16, 2020.

Z. Shaheen, G. Wohlgenannt, and E. Filtz. Large Scale Legal Text Classification Using Transformer Models. *arXiv preprint arXiv:2010.12871*, 2020.

M. S. Sorower. A Literature Survey on Algorithms for Multi-label Learning. *Oregon State University, Corvallis*, 18(1):1–25, 2010.

K. Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

I. Spasic, P. Burnap, M. Greenwood, and M. Arribas-Ayllon. A Naïve Bayes Approach to Classifying Topics in Suicide Notes. *Biomedical Informatics Insights*, 5(Suppl. 1): 87–97, 2012.

M. A. Tahir, J. Kittler, and F. Yan. Inverse random under sampling for class imbalance problem and its application to multi-label classification. *Pattern Recognition*, 45(10): 3738–3750, 2012.

J. Tanha, Y. Abdi, N. Samadi, N. Razzaghi, and M. Asadpour. Boosting methods for multi-class imbalanced data classification: an experimental review. *Journal of Big Data*, 7:1–47, 2020.

A. N. Tarekegn, M. Giacobini, and K. Michalak. A review of methods for imbalanced multi-label classification. *Pattern Recognition*, 118:107965, 2021.

B. Trstenjak, S. Mikac, and D. Donko. KNN with TF-IDF based Framework for Text Categorization. *Procedia Engineering*, 69:1356–1364, 2014.

G. Tsoumakas and I. Katakis. Multi-label classification: An overview. *International Journal of Data Warehousing and Mining (IJDWM)*, 3(3):1–13, 2007.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention Is All You Need. *Advances in Neural Information Processing Systems*, 30, 2017.

J. Villena Román, S. Collada Pérez, S. Lana Serrano, and J. C. González Cristóbal. Hybrid approach combining machine learning and a rule-based expert system for text categorization. AAAI, 2011.

L. Voita. Sequence to Sequence (seq2seq) and Attention. `https://lena-voita.github.io/nlp_course/seq2seq_and_attention.html`, 2022.

S. I. Wang and C. D. Manning. Baselines and Bigrams: Simple, Good Sentiment and Topic Classification. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 90–94, 2012.

X. Wang, F. Wei, X. Liu, M. Zhou, and M. Zhang. Topic Sentiment Analysis in Twitter: A Graph-based Hashtag Sentiment Classification Approach. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*, pages 1031–1040, 2011.

K. Weiss, T. M. Khoshgoftaar, and D. Wang. A survey of transfer learning. *Journal of Big data*, 3:1–40, 2016.

Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le. XLNet: Generalized Autoregressive Pretraining for Language Understanding. *Advances in Neural Information Processing Systems*, 32, 2019.

J. Ye, J. Gao, Q. Li, H. Xu, J. Feng, Z. Wu, T. Yu, and L. Kong. ZeroGen: Efficient Zero-shot Learning via Dataset Generation. *arXiv preprint arXiv:2202.07922*, 2022.

G. Yenduri, G. Srivastava, P. K. R. Maddikunta, R. H. Jhaveri, W. Wang, A. V. Vasilakos, T. R. Gadekallu, et al. Generative Pre-trained Transformer: A Comprehensive Review on Enabling Technologies, Potential Applications, Emerging Challenges, and Future Directions. *arXiv preprint arXiv:2305.10435*, 2023.

V. Yogarajan, J. Montiel, T. Smith, and B. Pfahringer. Transformers for Multi-label Classification of Medical Text: An Empirical Comparison. In *International Conference on Artificial Intelligence in Medicine*, pages 114–123. Springer, 2021.

K. M. Yoo, D. Park, J. Kang, S.-W. Lee, and W. Park. GPT3Mix: Leveraging Large-scale Language Models for Text Augmentation. *arXiv preprint arXiv:2104.08826*, 2021.

M.-L. Zhang and Z.-H. Zhou. ML-KNN: A lazy learning approach to multi-label learning. *Pattern recognition*, 40(7):2038–2048, 2007.