Thesis Dissertation

LARGE-SCALE MULTI-MODAL ANALYSIS OF EMPLOYEE REVIEWS

Dimitrios Chatzigiannis

UNIVERSITY OF CYPRUS

COMPUTER SCIENCE DEPARTMENT

May 2026

UNIVERSITY OF CYPRUS
COMPUTER SCIENCE DEPARTMENT

Large-Scale Multi-Modal Analysis of Employee Reviews

Dimitrios Chatzigiannis

Supervisor
Dr. George Pallis

Thesis submitted in partial fulfilment of the requirements for the award of degree of Bachelor in Computer Science at University of Cyprus

I declare that this dissertation and any accompanying software are my own original work, conducted in full compliance with the University of Cyprus Code of Conduct for Research (https://www.ucy.ac.cy/legislation/kodikas-deontologias-kai-politikes/), and with full accountability on my part for the accuracy, integrity, and validity of all content.

May 2026

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Dr. George Pallis, for his continuous support, insightful feedback and encouragement throughout the development and writing of this thesis. His guidance greatly shaped the direction of this work.

I am also grateful to Dr. Demetris Paschalides for providing valuable resources and knowledge necessary to carry out this project.

Finally, I would like to thank my family and friends for their continued unwavering support and patience throughout my studies.

Abstract

In the modern era, many significant global disruptions have repeatedly changed work environments, employee experiences and priorities across almost every industry. From the COVID-19 pandemic, to the emergence of generative AI, work culture has shifted multiple times and nobody can be certain of what the future holds. This thesis presents a comprehensive, large-scale data-driven analysis that aims to find how workplace sentiment, discussion topics and employee reviews are affected, and how they have evolved across different temporal periods.

The analysis is conducted using a large corpus of 300,000 balanced employee reviews across multiple industries and time periods. It is conducted in two main phases with different goals and methods.

The first phase of the analysis consists of exploratory data analysis, carried out on the 300,000 reviews spanning 2008-2023, providing valuable insights on the dataset, and following that, a construction of predictive machine learning models for employee rating classification. Many classifiers are used and evaluated, including Random Forest, XGBoost, LightGBM, Logistic Regression and more methods, such as neural networks, ordinal regression and ensemble methods. With five-class predictions, the best model achieves roughly 59.5% accuracy, and after collapsing to three sentiment classes (Low, Medium, High) and adding multiple engineered features, the top models reach up to 76% accuracy and 0.75 F1 score.

In the second phase, topic modelling is performed using BERTopic, a transformer-based topic discovery framework, that is applied separately to three structured review fields: *pros, cons* and *summary*. BERTopic combines sentence transformer embeddings, UMAP dimensionality reduction, HDBSCAN clustering and class-based TF-IDF to produce coherent, semi-reproducible topics. Then, topic prevalence is tracked across three COVID periods to identify which workplace concerns and topics gained or lost prominence before, during and after the pandemic. Moreover, VADER-based sentiment analysis is used to both combined and separated review text (pros, cons, summary) with non-parametric statistical tests and estimates confirming period level differences in sentiment distributions.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background and Motivation

Throughout the last few decades, the rise of online platforms and social media has completely changed how people share their opinions and experiences. More specifically, employee review websites are now an essential data source for understanding workplace dynamics and experiences. The dataset used for this work consists of employee reviews with multiple metrics along with descriptions of positives and negatives. The combination of these structured ratings and free-form text makes this very suited to data analysis, allowing researchers to extract fine-grained insights on employee experiences.

At the same time, the COVID-19 pandemic, declared by the World Health Organization on 11 March 2020 [1], is considered by many the most impactful event in recent history, especially when concerning the workplace. Massive changes happened in the span of a few weeks, with a global remote working adoption, a great number of layoffs, changing health priorities and uncertainty about the future of work filling the minds of people. The pandemic exposed tensions around job security, management communication, work-life balance and the mental health of the employees, while also accelerating trends like flexible schedules and remote or hybrid work arrangements that were previously rare.

Understanding how employee opinions evolved across this period, and across time in general, represents a research question with practical significance for organisations, professionals and the employees themselves. Topic shifts, discourse and negatives along with positives changes that affect workplace satisfaction are very important and concern every individual who will eventually join the workforce. Moreover, with the newer global adoption of AI in the workplace (a possible future extension of this work), even more people including myself are worried and uncertain about what the future holds.

This thesis aims to address some of these concerns, pinpoint what factors influence

employee satisfaction and paint a comprehensive picture of workplace experience throughout these periods, using the balanced sample of employee reviews spanning 2008-2023, combining exploratory data analysis, machine learning techniques for classification, topic modelling with BERTopic [2] and sentiment analysis with VADER [3].

1.2 Problem Statement

Understanding the core factors that influence workplace satisfaction is a critical challenge for organisations that aim to retain talent and for employees that seek supportive work environments. While employee review platforms provide reviews with both numerical ratings and free-text descriptions, inferring significant data from them is still a difficult task. The core problem this thesis addresses is one methodology to find the factors that influence employee satisfaction, especially during periods of global disruption such as the COVID-19 pandemic.

Specifically, this work tackles these challenges in interpreting employee feedback:

1. **Disentangling mixed sentiments.** Employee reviews are not purely positive or negative. A single review usually contains different positive and negative statements (e.g. praising the work-life balance while criticising compensation). Treating a review as a single text might lead to struggles in capturing the distinct problems and benefits employees experience.

2. **Identifying specific drivers of satisfaction.** While star ratings show how satisfied employees are, they do not reveal why. This work attemps to find the actual topics that employees have strong opinions on (e.g. remote work policies, toxic management, layoff fears).

3. **Tracking shifts during global disruptions.** The priorities of the workforce are not always the same. Sudden changes can shift what employees value and dislike. This work aims to understand how one such major shift, the COVID-19 pandemic, changed the landscape of the workplace.

4. **Predicting overall perception.** With multiple aspects of a job (e.g., culture, leadership, compensation) continuously affecting an employee, it is often unclear which specific factors affect their overall rating of an employer the most. Determining the strongest predictors of satisfaction is vital for prioritizing employee well-being.

1.3 Research Questions

This thesis is guided by the following research questions:

RQ1: What data can be inferred from exploratory data analysis of employee reviews?

RQ2: Which features of an employee review are most predictive of the overall employee rating, and how well can machine learning models classify employee satisfaction from those features?

RQ3: What latent topics can be discovered in the *pros*, *cons*, and *summary* fields of employee reviews using topic modelling?

RQ4: How did the prevalence of those topics change across the Before, During, and Post COVID-19 periods?

RQ5: How did sentiment measured at both the combined-review and separate field-level shift across the three COVID-19 periods?

1.4 Contributions

The main contributions of this thesis are as follows:

1. **Comprehensive EDA of a large dataset of employee reviews.** We characterise the distribution of ratings employment types, geographic coverage, missing data patterns, and correlations between rating dimensions across a 300,000-review balanced sample.

2. **Multi-model supervised classification pipeline.** We train, evaluate, and compare multiple classifiers, some ordinal regression variants, a feed-forward neural network, and ensemble methods on the task of predicting overall employee satisfaction, demonstrating the better performance of gradient-boosted tree ensembles for this domain.

3. **Feature-level BERTopic models.** We train three separate BERTopic models, one per review field, producing interpretable topic representations for *pros*, *cons*, and *summary* text, and quantify topic quality through diagnostics such as diversity and outlier rate.

4. **Three-period longitudinal topic prevalence analysis.** We construct a three-way time partition (Before/During/Post COVID) and measure the change in topic prevalence across periods, identifying the strongest risers and fallers during and after the pandemic.

5. **Field-level VADER sentiment analysis with statistical testing.** We apply VADER both combined, and independently to the *pros*, *cons*, and *summary* fields and use

Kruskal-Wallis, Mann-Whitney U, and Cliff's delta to characterise period-level differences with appropriate statistical methods.

1.5 Thesis Structure

The remainder of this thesis is organised as follows.

Chapter 2 provides the technical background necessary to understand the methods employed. It covers the dataset, gradient boosting methods, neural networks for tabular data, ordinal regression, topic modelling including BERTopic and its components (sentence transformers, UMAP, HDBSCAN, and class-based TF-IDF), sentiment analysis and VADER, and non-parametric statistical testing.

Chapter 3 provides a detailed account of every implementation step, following the order in which the work was developed: EDA, predictive modelling, BERTopic training, topic prevalence analysis, combined-field sentiment analysis, and separate-field sentiment analysis.

Chapter 4 presents and interprets the results of all analyses, including model benchmarks, BERTopic diagnostics, topic evolution findings, and sentiment test outcomes.

Chapter 5 surveys related work on various topics such as remote work, the COVID-19 pandemic and employee review analysis in general.

Chapter 6 summarises the key findings, discusses limitations, and suggests directions for future research.

Chapter 2

Background

This chapter provides the technical foundations required to understand the methods that have been applied in this thesis.

2.1 The Dataset

The main data source is an already slightly preprocessed employee review dataset of about 7 million employee reviews. A subset of 300,000 reviews was drawn from the larger corpus, spanning the years 2008-2023, and was used as the final dataset. The dataset was balanced during preprocessing: 60,000 reviews were sampled at random from each of the five star-rating classes (1–5 stars). The original dataset is heavily skewed towards 4 and 5 star reviews, which means a classifier could achieve high accuracy simply by predicting the majority class, resulting in misleading metrics. The 60,000 figure was selected as a large number that satisfied the constraint that every class is balanced, while also improving efficiency significantly without having a sizeable impact on future results. This ensures that there is no obvious bias towards any of the classes. The total size of the final dataset in CSV form was roughly 167 MB.

A typical employee review consists of several structured components:

- **Overall rating** (1–5 stars): an integer rating reflecting the reviewer's assessment of the employer.
- **Sub-ratings**: numerical scores for work-life balance, culture and values, diversity and inclusion, senior leadership, career opportunities, and compensation and benefits.
- **CEO approval**: a categorical assessment (Approve, Disapprove, No Opinion).
- **Business outlook**: a categorical prediction (Positive, Neutral, Negative).

- **Pros field**: free text describing positive aspects of the employer and workplace.
- **Cons field**: free text describing negative aspects.
- **Summary field**: a brief free-text headline summarising the reviewer's overall impression.
- **Job title, location, and employment status**: demographic metadata attached to the review.

and a few more that will be referenced in the later chapters.

The fact that both structured ratings and unstructured text fields exist in these reviews makes the data particularly well-suited for analysis and research with multiple methods. The *pros* and *cons* fields contain contrasting sentiments for different aspects of the same review, while the *summary* field can often act as a general evaluation that might align with the underlying numerical rating.

A key concern with the dataset is selection bias: often times, employees who feel strong positive or negative emotions towards their workplace and/or employer and more likely to leave reviews than those with neutral experiences. The dataset used in this thesis originally consisted of approximately 7 million reviews, with a bias towards positive reviews (4–5 `rating_overall`). This also proved to affect the research negatively, since the significantly larger number of reviews reduced the effectiveness and speed of the analysis, due to lacking computational power. To combat these two issues, we decided to downscale the dataset to a sample of 300,000 reviews, stratified by the overall rating, with 60,000 reviews for each rating level from 1 to 5. This not only means that the dataset is considerably less imbalanced in regards to the overall ratings, but it also means analysis on these reviews can be carried out much more efficiently, with minimal changes to the results since the remaining number of reviews is still very large.

2.2 Gradient Boosting Methods

Gradient boosting is an ensemble learning technique that builds a predictive model by sequentially adding weak learners (typically shallow decision trees), each trained to correct the residual errors of the preceding ensemble [4]. It is one of the most successful supervised learning frameworks for tabular data.

2.2.1 Random Forest

Random Forest [5] constructs a "forest" of independent decision trees in parallel, each trained on different subsets of the data and random feature subspaces. It then averages the predictions of these trees and can reduce model variance and stabilize predictions, making it a reliable model that performs well with little hyperparameter tuning.

2.2.2 XGBoost

XGBoost (eXtreme Gradient Boosting) [6], is an efficient implementation of gradient boosting that constructs decision trees sequentially to minimize a loss function. It uses second-order gradients to better aproximate the loss function, which leads to more accurate tree construction at every iteration. It uses advanced regularization to penalize model complexity, preventing overfitting. It provides a robust framework with great speed and predictive power using techniques like tree pruning and handling of missing values.

2.2.3 LightGBM

LightGBM [7], is a gradient boosting framework engineered for extreme speed and memory efficiency, especially for huge datasets. It employs a "leaf-wise" growth strategy that splits the leaf node with the highest gain, resulting in deeper trees that significantly reduce loss. It uses a histogram-based approach which buckets continuous features into discrete bins instead of scanning every data point, reducing computational complexity. It also incorporates features such as Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) to handle large numbers of features without sacrificing training speed.

2.2.4 CatBoost

CatBoost [8] is a gradient boosting library that was optimized for datasets containing categorical features. It has the ability to handle categorical variables through "ordered target statistics", a technique that converts categories into numbers during training while preventing the "target leakage" associated with standard encoding methods. To improve generalization, CatBoost uses "ordered boosting", a permutation-based approach that ensures each model iteration is trained only on data available before the current observation, mostly eliminating prediction shift. It builds balanced, symmetric trees that are faster and more resistant to overfitting.

2.3 Ordinal Regression

When dealing with data like ratings (1 to 5), the order matters (5 is better than 4) but we don't necessarily know if the "distance" between a 1 and 2 means the same as the distance between a 4 and 5. Treating these as simple categories ignores the sequence, while treating them as standard numbers assumes those distances are uniform.

Ordinal regression models [9] solve this by assuming there's a latent continuous variable, and estimating a set of threshold parameters that determine class boundaries.

2.4 Neural Networks for Tabular Data

Deep learning has a lot of application on tabular data. Multi-Layer Perceptrons (MLPs) with batch normalisation [10] and dropout have been shown to perform competitively with gradient boosting methods when the dataset is large enough, and the features scaled suitably.

In this thesis, the architecture consisted of layers of sizes [input, 256, 128, 64, output], each followed by batch normalisation, ReLU activation and dropout ($p = 0.3$). The model was originally trained on a GPU, however due to technical issues and version mismatch it was then trained on a CPU for the final results.

2.4.1 Batch Normalisation

Batch normalisation [10] normalises each activation across a mini-batch to have zero mean and unit variance. It then applies a learned affine transformation. It accelerates training and reduces sensitivity to weight initialisation.

2.4.2 Dropout Regularisation

Dropout randomly sets a fraction of neuron activations to zero during training, preventing units from co-adapting and acting as an implicit ensemble of many sub-networks. A dropout rate of 30% was used in this thesis.

2.5 Ensemble Methods

Ensemble methods combine the outputs of multiple base models. Following are the methods that were used in this thesis specifically.

2.5.1 Hard Voting

In hard voting, each model in the ensemble casts a vote for one class label. The class receiving the most votes is the final prediction. This method reduces variance when base models make different errors.

2.5.2 Soft Voting

In soft voting, each model outputs class probabilities and the probabilities are averaged across models. The class with the highest mean probability is selected.

2.5.3 Stacking

Stacking (or "stacked generalisation") [11] trains a meta-learner on the out-of-fold predictions of base models. The meta-learner learns to combine the strenghts of each base model. In this thesis, a Logistic Regression was used as the meta-learner on top of the 3 best gradient boosting models.

2.6 Feature Engineering

Feature engineering is the process of transforming data into meaningful variables or "features", that machine learning models can use to learn patterns more effectively. In the context of employee rating prediction, the subjective nature of ratings can be hard for models to grasp, therefore we engineered the following features from the individual sub-ratings to hopefully increase performance:

- **Rating average**: mean of all numeric rating sub-scores.
- **Rating spread**: standard deviation of the numeric sub-scores, measuring consistency of sentiment across dimensions.
- **Rating range**: maximum minus minimum sub-score, capturing the total span of opinion.
- **Positive ratio**: proportion of sub-scores above 3.
- **Negative ratio**: proportion of sub-scores below 3.
- **Consistency score**: $1-$ (spread / 5), a normalised measure of rating coherence.

These features were then combined with the original features of the models and tested for performance.

2.7 Topic Modelling

Topic modelling is the process of using unsupervised machine learning methods to discover abstract "topics" that explain word co-occurence in a corpus.

2.7.1 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA), introduced by Blei et al. [12], is the most widely used topic model. It is a three-level hierarchical Bayesian model that organises a document collection into hidden thematic structures. It operates on the following assumptions:

- **Hierarchical Structure**: Each document is modeled as a combination of topics, where each topic is a probability distribution over a fixed vocabulary of words.
- **Generative Process**: The model assumes documents are generated through a two-stage process: first, the document selects a mixture of topics, and second, for each word in the document, it selects a topic from that mixture and then a word from that topic.
- **Latent Variables**: The topics themselves are hidden ("latent"), meaning the model infers them by analyzing the co-occurence patterns of words across the document collection.

Despite its popularity, LDA has well-known limitations:

- Topics are represented as bags of words with no semantic structure.
- The number of topics K must be specified in advance.
- The assumption of *exchangeability* (that the order of the words and documents does not matter) can be limiting as it ignores the sequence of the information.

2.7.2 Neural and Embedding-Based Topic Models

More recent approaches leverage pre-trained word or sentence embeddings to inject semantic knowledge into the topic discovery process. These models cluster documents in a dense embedding space where semantic similarity corresponds to geometric proximity, bypassing the sparse co-occurrence problem that affects LDA on short texts.

2.8 BERTopic

BERTopic is a neural topic modelling framework developed by Grootendorst in 2022 [2], which leverages pre-trained transformer-based language models and a class-based TF-IDF procedure to extract interpretable topics from large text corpora. It treats topic modelling as a clustering problem, grouping semantically similar documents together and then labeling each cluster with representative keywords. It consists of four sequential components: (i) document embedding, (ii) dimensionality reduction, (iii) density-based clustering, and (iv) topic representation via class-based TF-IDF.

2.8.1 Document Embedding with Sentence Transformers

The first step of BERTopic converts raw documents into dense numerical vector representations using a *sentence transformer* [13]. Each document is passed through the pre-trained transformer model, which produces a high-dimensional vector for each one.

In this thesis, the model `all-MiniLM-L6-v2` was used. This model is a distilled, six-layer version of a larger sentence transformer, retaining most of the semantic quality of the full model while being faster. It maps input text to a 384-dimensional embedding vector. The model was selected for its speed-accuracy trade-off, because of the significant amount of texts.

2.8.2 Dimensionality Reduction with UMAP

The 384-dimensional sentence embeddings are too high-dimensional for clustering algorithms to work efficiently, a problem known as the "curse of dimensionality". Uniform Manifold Approximation and Projection (UMAP) [14], introduced by McInnes et al., is commonly used with BERTopic to reduce the embeddings to a much lower dimensional space.

It preserves both local and global structure of the high-dimensional space in the low-dimensional representation, maintaining meaningful relationships between similar or dissimilar documents. The key hyperparameters used are somewhat standard:

- `n_components`: the output dimensionality (5 in this thesis).
- `n_neighbors`: the number of nearest neighbours used to build the high-dimensional graph (15 in this thesis). A higher value, such as 30 could also be used as 15 can sometimes be too local.
- `min_dist`: controls how tightly UMAP packs points in the low-dimensional space (0.0 allows clusters to be as dense as possible, producing well-seperated groupings)

- `metric`: the distance function in the high-dimensional space (`cosine` for normalised embedding vectors).

A fixed `random_state=42` was used throughout to ensure reproducibility.

2.8.3 Clustering with HDBSCAN

Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN), developed by McInnes et al. [15], is applied to the UMAP-reduced embeddings. Used after dimensionality reduction, documents are grouped into semantically coherent clusters that will form topics.

Key advantages of HDBSCAN for topic discovery are:

- It automatically discovers the optimal number of clusters based on the data.
- It handles outliers by assigning them to a noise cluster (topic –1), which prevents the pollution of topics.
- It handles clusters of varied shapes and densities, making it more powerful than centroid-based alternatives such as K-Means for real-world text data.

The parameters used in this thesis were once again standard:

- `min_cluster_size=150`: minimum number of documents required to form a topic cluster
- `min_samples=10`: minimum number of core pointss required for density estimation
- `metric="euclidean"`: how the algorithm calculates the distance between points in the reduced space created by UMAP.
- `cluster_selection_method="eom"`: short for "Excess Of Mass", standard for finding the most persisent clusters in the hierarchy.

2.8.4 Topic Representation with Class-Based TF-IDF (c-TF-IDF)

Once documents are assigned to clusters, BERTopic derives a representation for each topic using a modified TF-IDF scheme called *class-based TF-IDF*. The key change is that instead of computing TF-IDF at the document level, it instead treats each cluster as a single class document by concatenating all documents in that cluster. It then computes which words are most important within a class relative to all other classes, producing a set of top keywords per topic.

Thus, the output is a set of topics, each represented by its top-k most representative words. Also, BERTopic assigns three representative documents for each topic (real documents from the corpus that belong to that specific topic cluster).

2.8.5 Topic Diagnostics

BERTopic produces several diagnostics that were used to assess model quality in this thesis:

- **Number of topics**: the count of non-outlier topic clusters found.
- **Outlier rate**: the percentage of documents assigned to the noise cluster (topic –1). A high outlier rate may indicate that `min_cluster_size` is too restrictive.
- **Topic diversity**: the fraction of unique words across all topic representations (1.0 = every word appears in exactly one topic).
- **Top-10 coverage**: the share of non-outlier documents falling in the ten largest topics.

2.9 Sentiment Analysis

Sentiment analysis, also known as opinion mining, is the computational study of people's opinions, sentiments, appraisals, attitudes, and emotions toward entities and their attributes [16]. Its main goal is to determine the *polarity* of a piece of text: positive, negative, or neutral. Approaches to sentiment analysis can be broadly divided into lexicon-based and learning-based methods.

2.9.1 Lexicon-Based Approaches

Lexicon-based sentiment analysis is a technique that determines the semantic orientation of a text by aggregating the scores of individual "sentiment-bearing" words. These methods require no labelled training data, are computationally lightweight, and any score can be traced back to specific lexical items.

The general process typically involves **tokenization** (breaking the text into individual words or phrases), **lexicon lookup** (matching tokens against the lexicon to get their associated sentiment values) and **aggregation** (summing or averaging these scores to produce an final sentiment score for the text).

The biggest weakness of lexicon-based approaches is their inability to capture context: the word “great” is positive on its own, but negative in a phrase such as “great depression”. Negations, sarcasm and domain-specific language are also handled poorly.

2.9.2 Learning-Based Approaches

Supervised learning approaches are methods that train a model to recognize text patterns rather than relying on manually pre-determined dictionaries [16]. Deep learning models, and particularly fine-tuned transformer models such as BERT [17], now represent the state of the art on benchmark sentiment classification tasks, achieving near-human performance on many datasets. However, they require labelled training data and are more computationally expensive than lexicon-based methods.

For this thesis, the lexicon-based VADER tool was chosen, as described in the following section.

2.10 VADER: Valence Aware Dictionary and Sentiment Reasoner

VADER (Valence Aware Dictionary and sEntiment Reasoner) is a rule-based sentiment analyser [3] that was specifically designed for social media text. It combines a curated sentiment lexicon with a set of grammatical heuristics to handle phoenomena that simpler approaches might struggle with.

2.10.1 Sentiment Lexicon

VADER's lexicon assigns each word or emoji a *mean sentiment rating* on a scale from –4 (extremely negative) to +4 (extremely positive), letting the tool understand both the direction and the intensity of an opinion. It contains over 7,500 sentiment-bearing words and phrases.

2.10.2 Heuristic Rules

Beyond the base lexicon, VADER incorporates five classes of heuristics:

1. **Punctuation**: exclamation marks amplify sentiment intensity (*e.g.,* "Great!!!!" is more positive than "Great.").
2. **Capitalisation**: all-caps words express greater intensity (*e.g.,* "HAPPY" > "happy").
3. **Degree modifiers**: adverbs such as *very*, *slightly*, and *barely* amplify or attenuate adjacent words.
4. **Contrastive conjunctions**: the word *but* shifts sentiment weight towards the phrase following it.

5. **Negation**: common negation patterns shift or negate the polarity of following tokens (e.g. “not good” becomes negative) .

2.10.3 Compound Score

VADER produces four output scores for any input text: `pos`, `neu`, `neg` (the proportions of text in each class), and `compound`, a normalised, weighted composite score in the range $[-1, +1]$. The compound score is the primary metric used in this thesis. Following the convention established by Hutto and Gilbert [3], a compound score ≥ 0.05 is labelled *positive*, ≤ -0.05 is labelled *negative*, and the intermediate range is *neutral*.

2.10.4 Why VADER Was Chosen

VADER was selected for this thesis for several reasons:

- It requires no training data and can be applied out-of-the-box to a new domain.
- It is computationally efficient, enabling sentiment scoring of 300,000 reviews very quickly compared to other means.
- It is widely used as a baseline in computational social science research, making results comparable to prior work.
- The text fields are short and conversational, the same genre for which VADER was originally designed.

2.11 Non-Parametric Statistical Testing

When analyzing sentiment scores across a massive amount of reviews, standard statistical tests are often not suitable because they rely on assumptions (such as normal distribution, equal variance) that might not be true in such datasets. To address this, we use non-parametric tests, which do not make these strict assumptions.

2.11.1 Kruskal-Wallis Test

The Kruskal-Wallis test [18], instead of comparing the raw values of different groups, ranks all observations from lowest to highest and compares these mean ranks across groups. Then, it tests the null hypothesis that all groups come from the same distribution. A statistically significant result H indicates that at least one group tends to have higher or lower scores than the others.

2.11.2 Mann-Whitney U Test

The Mann-Whitny U test [19] is used for pairwise comparisons. Rather than checking for differences in means, it tests whether one of the two distributions is stochastically larger than the other (whether a randomly selected value from one group is more likely to be higher than one from the other group). It makes no assumptions and is not easily skewed by outliers.

2.11.3 Cliff's Delta Effect Size

Statistical significance tells us if a difference is unlikely to be due to chance, but with very large datasets, even tiny differences can produce significant p-values. To measure practical significance, we use Cliff's Delta [20]. It is a non-parametric effect size measure defined as:

$$\delta = \frac{|\{(i,j) : a_i > b_j\}| - |\{(i,j) : a_i < b_j\}|}{n_a \cdot n_b} \tag{2.1}$$

where a and b are two samples of sizes n_a and n_b. The value ranges from -1 to $+1$: $|\delta| < 0.147$ is considered negligible, 0.147–0.33 small, 0.33–0.474 medium, and ≥ 0.474 large. Because computing all pairwise comparisons is very computationally expensive in this case ($O(n^2)$), approximate Cliff's delta was computed using random subsamples of 20,000 observations per group.

2.12 COVID-19 and the Modern Workplace

The COVID-19 pandemic was declared a global pandemic by the World Health Organisation on 11 March 2020 [1] and triggered an enormous amount of sudden disruptions and changes in the workplace.

2.12.1 Immediate Impact (2020)

The first obvious effects of the pandemic included the mandatory closures of non-essential workplaces, the rapid adoption of remote work and a massive amount of layoffs [21]. Organisations struggled with the early challenges of remote collaboration and the maintenance of organisational culture in distributed teams.

2.12.2 During The Pandemic (2020–2021)

Throughout 2020 and 2021, the pandemic continued shaping workplace dynamics. Plans for return to offices were repeatedly postponed and cancelled, vaccine rollouts created

new debates around mandates and safety, and mental health, especially early in the pandemic, had deteriorated [22].

2.12.3 After The Pandemic(2022–)

From 2022 onwards, vaccination rates in developed countries reduced health concerns, however the workplace had been permanently changed. Remote and hybrid working arrangements became normalised in a lot of organisations, with rising expectations around work-life balance and flexibility, and inflation lead to concerns about compensation adequacy.

These phases provide the natural temporal structure for the COVID-period partitioning used in this thesis. The transition from "Before" to "During" is based on the WHO pandemic declaration (11 March 2020), and the transition from "During" to "Post" is set at 1 January 2022.

2.13 Key Python Libraries and Tools

The entire analysis was implemented mostly in Python 3.13, using a large amount of libraries, some important ones include:

- **pandas** [23]: data loading, cleaning, transformation, and aggregation.
- **NumPy** [24]: numerical arrays and vectorised computation.
- **scikit-learn** [25]: preprocessing, model training (Random Forest, Logistic Regression, stacking), evaluation metrics, and train/test splitting.
- **XGBoost** [6]: gradient boosted trees
- **LightGBM** [7]: gradient boosted trees with GPU support.
- **CatBoost** [8]: gradient boosted trees with native categorical feature handling.
- **PyTorch** [26]: neural network implementation and GPU training (however, after hardware conflicts in later stages of the work, GPU training was not used).
- **BERTopic** [2]: transformer-based topic modelling.
- **sentence-transformers** [13]: document embedding.
- **UMAP-learn** [14]: dimensionality reduction.
- **hdbscan** [15]: density-based clustering.

- **NLTK / VADER** [3]: lexicon-based sentiment analysis.
- **statsmodels** [27]: logistic regression with statistical summaries.
- **SciPy**: Kruskal-Wallis and Mann-Whitney U tests.
- **matplotlib** [28] and **seaborn**: data visualisation.
- **mord** [29]: ordinal regression models.

Chapter 3

Methodology

This chapter provides a detailed account of every implementation step. The presentation follows the chronological order in which the work was carried out. The results of each step along with intepretations are included in Chapter 4.

3.1 Exploratory Data Analysis

The first notebook loads the balanced dataset and performs a comprehensive exploratory analysis to characterise its structure and possibly infer easy-to-gather data before any modelling begins.

3.1.1 Data Loading and Configuration

The dataset was loaded using `pandas.read_csv`, and some global display options were set to help in the inspection of wide tables. Seaborn's dark theme was applied for visual clarity throughout the EDA.

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv('reviews_balanced_300k.csv')
sns.set_theme(style="dark")

pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)
pd.set_option('display.width', None)
pd.set_option('display.max_colwidth', None)
```

Listing 3.1: Loading the balanced dataset and configuring display options.

3.1.2 Dataset Overview and Variables

The unprocessed dataset contains 300,000 rows and 50 columns. Variable types include `int64` (numerical ratings, binary flags), `object` (categorical and free-text), `bool` (current employee flag), and `float64` (sparse ratings and funding figures). The `df.dtypes` and `df.describe()` calls confirmed that numerical ratings are all bounded within their expected ranges.

Table 3.1 summarises the most important columns present in the dataset. Some variable names differ after different preprocessing methods were used per notebook, however they are semantically the same. The most significant variables used in this thesis are described in detail below.

Table 3.1: Key variables in the balanced dataset.

Variable	**Type**	**Description**
`id`	int64	Unique review identifier
`review_date_time`	object	Timestamp of the review
`year`	int64	Year of review
`industryName`	object	Employer's industry category
`country_code`	object	ISO 3166-1 country code
`employee_count`	object	Company size band
`rating_overall`	int64	Overall star rating (1–5)
`rating_work_life_balance`	object	Work-life balance (0–5)
`rating_culture_and_values`	int64	Culture & values (0–5)
`rating_diversity_and_inclusion`	int64	D&I sub-rating (0–5)
`rating_senior_leadership`	int64	Senior leadership (0–5)
`rating_career_opportunities`	int64	Career opportunities (0–5)
`rating_compensation_and_benefits`	int64	Compensation & benefits (0–5)
`rating_ceo_imputed`	object	CEO approval (imputed)
`rating_business_outlook_imputed`	object	Business outlook (imputed)
`rating_recommend_to_friend_imputed`	object	Recommend to a friend (imputed)
`employment_status_imputed`	object	Employment type (imputed)
`is_current_job`	bool	Current employee flag
`length_of_employment`	int64	Years at company
`pros`	object	Free-text positive aspects
`cons`	object	Free-text negative aspects
`summary`	object	Free-text headline summary
`has_stress`	int64	Stress indicator from previous work

3.1.3 Missing Value Analysis

A missing-value analysis was performed using `df.isnull().sum()`. The findings, sorted by missing count, revealed:

- `processed_location`: 137,429 missing (45.8%) — excluded.
- `rating_ceo`: 103,885 missing (34.6%) — imputed version used.
- `rating_business_outlook`: 100,324 missing (33.4%) — imputed version used.
- `rating_recommend_to_friend`: 77,716 missing (25.9%) — imputed version used.
- `country_code`: 5,126 missing (1.7%).
- `industryName`: 1,797 missing (0.6%).
- Text fields (`pros`, `cons`, `summary`): negligible amount ($\leq$ 166 rows).

3.1.4 Dataset Cleanup

For this notebook, some of the columns were renamed for clarity. Seven individual employment status indicator columns for each type (`employment_status_FREELANCE`, `employment_status_PART_TIME`, etc) were dropped in favour of a single imputed categorical column. The binary columns could prove useful in other types of analysis (such as deep-learning) but for exploratory data analysis the single column is easier to work with.

3.1.5 Visual Exploration

Several visualisations were produced to understand the distribution of key variables:

Rating distribution. A count plot of `rating_overall` confirmed that the balanced sampling succeeded.

Temporal distribution. A histogram with KDE overlay of `review_date_time` was visualized to find the distribution of dates within the reviews, information which helped shape the direction of the later phases of this thesis.

Industry distribution. The top-10 industries present in the review samples, along with a count of every industry that appears, were visualised.

Employment type distribution. A count plot of `employment_status` was visualized to find the distribution of employment types in the employee reviews.

Country distribution. The top-10 countries by review count were visualised, in order to understand the main source of reviews geographically.

The plots and their analysis can be found in Section 4.1.

3.1.6 Correlation Analysis

A Pearson correlation heatmap of the six numerical sub-ratings (`work_life_balance`, `culture_and_values`, `diversity_and_inclusion`, `senior_leadership`, `career_opportunities`, `compensation_and_benefits`) was computed in order to estimate how each sub-rating is connected and what aspects of a workplace seem to be correlated.

3.1.7 Logistic Regression for Stress Prediction

A logistic regression model was fitted using `statsmodels.Logit` [27] to predict the binary variable `DV_has_stress` (whether a review contains stress-related language) from the six sub-ratings. The model was fitted on the full 300,000-row dataset with a constant term added via `sm.add_constant`.

The results were then visualised as predicted probability curves.

3.2 Predictive Modelling

The second notebook constructs and evaluates a comprehensive pipeline of supervised machine learning models for predicting the overall employee star rating.

3.2.1 Balanced Sample

The original dataset was highly imbalanced, with 4 and 5-star reviews being far more frequent than 1 and 2-star reviews. Moreover, the massive number of reviews proved to negatively impact the efficiency of this work after multiple attempts. Thus, before any modelling, the balanced sample of 300,000 total reviews was constructed. This sample was saved to `reviews_balanced_300k.csv` for use in all subsequent analyses and for a re-evaluation of the exploratory data analysis mentioned previously.

3.2.2 Feature Selection

Fifteen candidate features were identified based on domain knowledge and data availability:

- **Demographic**: `industryName`, `country_code`, `employment_status_imputed`
- **Imputed categorical ratings**: `rating_ceo`, `rating_business_outlook`, `rating_recommend_to_friend` (the imputed versions)

- **Numerical ratings**: `rating_work_life_balance`, `rating_culture_and_values`, `rating_diversity_and_inclusion`, `rating_senior_leadership`, `rating_career_opportunities`, `rating_compensation_and_benefits`
- **Job details**: `length_of_employment`, `is_current_job`, `employee_count`

COVID-period flags and funding columns were excluded: the former, if needed, could be derived from the review date, and the latter exhibit extreme skew and high missing rates.

A LightGBM model [7] with 100 estimators was trained on this feature set to rank features by importance. LightGBM was chosen for this screening step because its gradient-boosted trees handle mixed numeric/categorical features and compute split-gain importance scores in a single pass, making it computationally efficient.

3.2.3 Train/Test Split and Preprocessing

The 300,000-row dataset was split into 240,000 training and 60,000 test samples using a stratified 80/20 split with `random_state=42`. Each rating class contributes 48,000 training and 12,000 test samples.

Two preprocessing pipelines were created:

- **Label-encoded** version for tree-based models: categorical columns were label-encoded using `sklearn.preprocessing.LabelEncoder`. Tree based models don't need scaling because they split on feature thresholds, so the absolute magnitude of values is irrelevant. Therefore, label encoding (which just converts strings to integers) is all they need.
- **Standardised** version for linear models: all features were first label-encoded, then standardised to zero mean and unit variance using `sklearn.preprocessing.StandardScaler`. These models compute distances or dot products between feature vectors, so without standardisation, features with larger numeric ranges dominate and distort the geometry.

3.2.4 Baseline 5-Class Model Comparison

Six models were trained with default hyperparameters on the 5-class target:

1. **Random Forest** [5]: 100 trees.
2. **XGBoost** [6]: 100 estimators, multi-class log loss.
3. **LightGBM** [7]: 100 estimators.

4. **CatBoost** [8]: default parameters.

5. **Gradient Boosting**: scikit-learn's `GradientBoostingClassifier` with 100 estimators.

6. **Logistic Regression**: L2-regularised, max 1000 iterations.

A large comparison was conducted rather than committing to one model upfront, since the relative performance of each model cannot be determined without testing. Including both gradient boosting variants and a linear baseline allows the evaluation to distinguish whether performance gains come from a specific framewrk or from non-linearity in general.

GPU acceleration (CUDA) was originally implemented for XGBoost, LightGBM and CatBoost where supported, however technical difficulties meant that the CPU had to be used.

3.2.5 3-Class Target and Feature Engineering

To improve model performance and interpretability, the 5-class target was collapsed into three classes:

- **Low (0)**: ratings 1–2 (negative experiences)
- **Medium (1)**: rating 3 (neutral)
- **High (2)**: ratings 4–5 (positive experiences)

This gives a new label distribution of approximately 120,000 Low, 60,000 Medium, and 120,000 High reviews.

The 5-class rating target is ordinal but the class boundaries are subjective and noisy, thus collapsing to Low/Medium/High reduces this noise and produces a semantically meaningful target while keeping the key negative/neutral/positive distinction relevant to the research questions.

Six engineered features were added to the 11 strong features:

- **rating_average**: mean of all numeric sub-ratings.
- **rating_spread**: standard deviation of sub-ratings.
- **rating_range**: max minus min of sub-ratings.
- **positive_ratio**: proportion of sub-ratings exceeding 3.

- **negative_ratio**: proportion of sub-ratings below 3.
- **consistency_score**: $1 - \text{spread}/5$.

These aggregated features capture patterns in an employee's sub-rating profile that are difficult for individual columns to encode.

All six models were retrained on the 17-feature, 3-class dataset.

3.2.6 Ordinal Regression

Three ordinal regression models from the `mord` library [9, 29] were trained on the 3-class target using the standardised feature set:

- **OrdinalRidge**: proportional odds model with ridge (L_2) regularisation.
- **LogisticAT** (All-Threshold): a logistic regression model that simultaneously optimises all pairwise thresholds.
- **LogisticIT** (Immediate-Threshold): a variant that optimises only adjacent thresholds.

Standard multi-class classifiers treat the target classes as unordered. Since the rating target has a natural ordering (low < medium < high), ordinal regression explicitly encodes this ordering through threshold constraints, possibly reducing penalty for wrong classifications. By including ordinal models, we test whether using the ordinal structure yields gains over other types of classifiers.

3.2.7 Neural Network

The architecture consists of four linear layers with hidden dimensions [256, 128, 64], each followed by batch normalisation [10], ReLU activation, and 30% dropout. The output layer has three units with no activation (raw logits are passed to the cross-entropy loss).

```
class TabularNN(nn.Module):
    def __init__(self, input_size, hidden_sizes=[256, 128, 64], num_classes=3,
        dropout=0.3):
        super(TabularNN, self).__init__()

        layers = []
        prev_size = input_size

        for hidden_size in hidden_sizes:
            layers.append(nn.Linear(prev_size, hidden_size))
            layers.append(nn.BatchNorm1d(hidden_size))
            layers.append(nn.ReLU())
            layers.append(nn.Dropout(dropout))
```

```
            prev_size = hidden_size

        layers.append(nn.Linear(prev_size, num_classes))
        self.network = nn.Sequential(*layers)

    def forward(self, x):
        return self.network(x)
```

Listing 3.2: Neural network architecture definition in PyTorch.

The feed-forward network was included to test whether a non-linear architecture with learned feature interactions outperforms the tree-based models on this specific dataset. Batch normalisation and dropout were used to mitigate overfitting given the low feature count, and ReduceLROnPlateau was used to avoid committing to a fixed learning rate schedule.

The model was trained for 20 epochs, using the Adam optimiser (`lr=0.001`, `weight_decay=1e-5`) with a `ReduceLROnPlateau` scheduler (patience 3, factor 0.5) and a batch size of 256.

3.2.8 Ensemble Methods

Three ensemble approaches were applied to the top-3 performing models (XGBoost, LightGBM, and CatBoost):

1. **Hard voting**: majority vote across the three models' class predictions.
2. **Soft voting**: average of the three models' class probability outputs.
3. **Stacking** [11]: a Logistic Regression meta-learner trained on out-of-fold predictions from the three base models, using scikit-learn's `StackingClassifier`.

3.3 BERTopic Training

The third notebook trains three BERTopic topic models (one per review field) and appends topic assignments.

3.3.1 Framework Selection Reasoning

BERTopic was selected over normal LDA for several reasons: LDA treats document as a bag-of-words without taking into account positional structure as mentioned preivously. For short review text this is a big issue because the co-occurence counts on which LDA relies are sparse in short documents (such as these reviews) and inferring topics would

probably be difficult. Moreover, LDA also requires a fixed number of topics to be determined but in this case the structure of the corpus is already known and the goal is inferring all possible topics. Finally, LDA struggles with polysemy: *managemnet* can refer to supervisors in a review and to senior leadership in general in another, but LDA would assign it a single weight globally.

BERTopic addresses these limitations. By encoding each document with a sentence transformer, it captures semantic meaning and not just co-occurence. The `all-MiniLM-L6-v2` encoder was chosen specifically for balancing embedding quality and speed: at 384 dimensions, it is substantially faster than larger models while retaining great ability for clustering. UMap then reduces those embeddings to a lower dimensional space making the clustering more robust to noise. Finally, HDBScan was used because it does not require a pre-specified number of topics and handles noise document well via the outlier class (topic -1), which is appropriate for reviews when not every review is expected to fit a single, specific theme.

3.3.2 Data Preparation

The balanced dataset was loaded and the three text columns (`pros`, `cons`, `summary`) were cleaned by filling NaN values with empty strings and converting all values to string type. A composite `doc` field was also constructed ("PROS: ... CONS: ... SUMMARY: ...") for debugging purposes, though the three separate-field models are the focus.

Row filtering removed documents where the length of the concatenated text was below 20 characters, eliminating empty or near-empty reviews that would produce uninformative embeddings.

3.3.3 UMAP and HDBSCAN Configuration

Two factory functions were defined to create fresh instances of the dimensionality reducer and clusterer for each of the three models:

```
def make_umap():
    return UMAP(n_components=5, n_neighbors=15,
                min_dist=0.0, metric='cosine',
                random_state=42, low_memory=False)

def make_hdbscan():
    return HDBSCAN(min_cluster_size=150, min_samples=10,
                   metric='euclidean',
                   cluster_selection_method='eom',
                   prediction_data=True)
```

Listing 3.3: UMAP and HDBSCAN configuration for BERTopic.

The `prediction_data=True` flag is required for HDBSCAN to later support out-of-training cluster assignment, which BERTopic uses internally.

3.3.4 BERTopic Model Instantiation and Training

Three BERTopic models were created and trained independently:

```
topic_pros = BERTopic(language='english', verbose=True,
                      umap_model=make_umap(),
                      hdbscan_model=make_hdbscan())
topic_cons = BERTopic(language='english', verbose=True,
                      umap_model=make_umap(),
                      hdbscan_model=make_hdbscan())
topic_sum  = BERTopic(language='english', verbose=True,
                      umap_model=make_umap(),
                      hdbscan_model=make_hdbscan())

topics_pros, probs_pros = topic_pros.fit_transform(docs_pros)
topics_cons, probs_cons = topic_cons.fit_transform(docs_cons)
topics_sum,  probs_sum  = topic_sum.fit_transform(docs_summary)
```

Listing 3.4: BERTopic model instantiation and training.

Each call to `fit_transform` triggers the full BERTopic pipeline: (1) embedding all documents with `all-MiniLM-L6-v2`, (2) reducing dimensions with UMAP, (3) clustering with HDBSCAN, and (4) computing c-TF-IDF representations for each cluster.

3.3.5 Topic Diagnostics

A custom diagnostic function was implemented to summarise each model's output:

```
def topic_diagnostics(model, label):
    info = model.get_topic_info()
    non_outlier = info[info['Topic'] != -1]
    total_docs  = info['Count'].sum()
    outlier_pct = info.loc[info['Topic']==-1,'Count'].sum() \
                  / total_docs * 100
    n_topics    = len(non_outlier)
    all_words   = [w for tid in non_outlier['Topic']
                   for w, _ in model.get_topic(tid)]
    diversity   = len(set(all_words)) / len(all_words)
    top10_cov   = non_outlier.head(10)['Count'].sum() \
                  / total_docs * 100
    print(f'{label}: {n_topics} topics, '
          f'{outlier_pct:.1f}% outliers, '
          f'diversity={diversity:.3f}, '
          f'top10_cov={top10_cov:.1f}%')
```

Listing 3.5: Topic diagnostics function.

3.3.6 Model Persistence

The three trained models were saved using BERTopic's `safetensors` serialisation format, which stores both the cluster parameters and the underlying sentence transformer weights:

```
topic_pros.save('models/bertopic_pros',
                serialization='safetensors')
topic_cons.save('models/bertopic_cons',
                serialization='safetensors')
topic_sum.save('models/bertopic_summary',
               serialization='safetensors')
```

Listing 3.6: Saving BERTopic models.

Topic info tables were exported to CSV files in `outputs-bertopic/`, and the dataframe with added topic columns (`topic_pros`, `topic_cons`, `topic_summary`) was saved as a Parquet file at `outputs/reviews_with_topics.parquet`.

3.3.7 Three-Period Data Preparation and CADE Analysis

To supplement the macro-level topic modeling, an exploratory semantic shift analysis using Compass-aligned Distributional Embeddings (CADE) [30] was conducted to observe how specific workplace vocabulary changed meaning over time, by measuring word-level cosine similarity shifts between the Before, During, and Post-COVID periods (more on that in the next section). Ultimately, this analysis was supplementary, because of technical difficulties. Moreover, the only significant observations were terms like *remote* evolving from isolated pre-pandemic contexts to establish new COVID-era associations (e.g., *wfh*, *zoom*, *hybrid*), and a general increase in mentions of COVID, pandemics and similar words.

3.4 Topic Prevalence Analysis

This notebook quantifies how the share of each topic changed across three COVID periods.

3.4.1 COVID-19 Period Definitions

The temporal partitioning of the dataset into three periods is a central design decision for this thesis. The following boundaries were used, consistent with major public health and economic milestones:

BEFORE: Any review with `review_date_time` strictly before 11 March 2020. This date corresponds to the WHO's official declaration of COVID-19 as a global pandemic [1]. The BEFORE period spans 12 years (2008–March 2020) and contains **146,783 reviews**.

DURING: Reviews dated from 11 March 2020 (inclusive) to 31 December 2021 (inclusive). This captures the main pandemic phase, including national lockdowns, the first vaccine rollouts. The DURING period contains **76,673 reviews**.

POST: Reviews dated from 1 January 2022 onwards. This threshold captures the broad return to normality in most economies. The POST period contains **76,544 reviews**.

These boundaries were implemented using `pd.cut` with half-open intervals $[\text{start}, \text{end})$, and the resulting period assignment was stored as an ordered categorical variable with levels $\{\text{BEFORE} < \text{DURING} < \text{POST}\}$.

3.4.2 Data Loading

The topic-annotated Parquet file and the three topic-info CSV files were loaded.

3.4.3 Topic Prevalence Function

A general-purpose prevalence function was implemented:

```
def topic_prevalence(df, topic_col,
                     period_col='PERIOD',
                     topics=None, drop_outlier=True):
    d = df[df[period_col].notna()].copy()
    if drop_outlier:
        d = d[d[topic_col] != -1]
    if topics is not None:
        d = d[d[topic_col].isin(topics)]

    counts = (d.groupby([period_col, topic_col])
              .size().rename('count').reset_index())
    totals = (d.groupby(period_col)
              .size().rename('total').reset_index())
    merged = counts.merge(totals, on=period_col)
    merged['share'] = merged['count'] / merged['total']

    wide = merged.pivot(index=topic_col,
                        columns=period_col,
                        values='share').fillna(0.0)
    wide['delta_DURING_MINUS_BEFORE'] = \
        wide.get('DURING', 0) - wide.get('BEFORE', 0)
    wide['delta_POST_MINUS_DURING']   = \
        wide.get('POST', 0) - wide.get('DURING', 0)
    wide['delta_POST_MINUS_BEFORE']   = \
        wide.get('POST', 0) - wide.get('BEFORE', 0)
    return wide.sort_values('delta_POST_MINUS_BEFORE',
```

```
27                                    ascending=False)
```

Listing 3.7: Topic prevalence computation.

This function computes, for each topic and period, the share of non-outlier documents assigned to that topic, then expresses changes as additive deltas, in other words the percentage of share each topic gained or lost throughout the periods. Crucially, to calculate a topic's share, its document count is divided by the total number of *classified* documents (excluding outliers) for that period, rather than the absolute total number of reviews. BERTopic assigns noisy or unclassifiable text to an "outlier" group (Topic -1). If the volume of these outlier reviews fluctuates over time, dividing by the absolute total would artificially distort the percentages of valid topics. By normalizing against only the non-outlier documents, we ensure an accurate comparison of how topics grew or shrank relative to one another.

3.4.4 Topic Label Joining

A helper function appended human-readable topic names from the topic-info tables to the prevalence results:

```
1 def add_topic_labels(prev_df, info_df):
2     meta = info_df[['Topic','Name']].drop_duplicates()
3                                     .set_index('Topic')
4     return prev_df.join(meta, how='left')
```

Listing 3.8: Joining topic labels.

3.4.5 Targeted Topic Analysis

The first analysis tracked a manually curated set of topic IDs for each text field, those deemed semantically meaningful upon inspection of the topic labels. For each topic, its share of total documents was computed per period, yielding their prevalence across BEFORE, DURING, and POST. This provided a structured overview of how a broad range of specific themes shifted across the timeline.

3.4.6 Top-Volume Topics Baseline

The analysis was then narrowed to the ten highest-volume topics per field, filtered to a minimum of 800 associated documents. Restricting attention to these dominant topics offered a cleaner baseline: the themes that collectively account for the largest share of discourse, and how their relative prevalence evolved across the three periods.

3.4.7 Full-Corpus Riser and Faller Analysis

Finally, the prevalence function was applied to the full topic space (no topic filtering) for each field. Topics were ranked by the magnitude of their period-over-period share change, and the top 15 strongest risers and fallers were extracted, first for the POST vs. BEFORE transition, then for DURING vs. BEFORE. This helps us understand the trends and prevalence of topics, both with the years of difference between pre-pandemic and post-pandemic, and during the initial declaration.

3.5 Combined-Field Sentiment Analysis

This notebook concatenates the three text fields into a single combined text per review and applies VADER sentiment analysis.

3.5.1 VADER Selection Reasoning

VADER was chosen over fine-tuned transformer models and LLMs for several reasons: First, VADER was designed specifically for shortm informal social-media style text, which employee reviews very closely resemble: brief, opinionated and often punctuation-heavy. Second, the dataset includes 300,000 reviews, each with three text fields, meaning any approach will inevitably have to process about 900,000 text fields. A lexicon-based method can do this with negligible computational cost, whereas a transformer encoder would require significant GPU time and computational power. Third, no labelled sentiment data was available, and thus fine-tuning a transformer would require manual annotation or noisy heuristic labels, introducing more errors. Fourth, VADER's compound score is a continuous value on $[-1, 1]$, enabling the non-parametric effect-size estimation used in the comparative analysis across COVID periods. Finally, VADER is fully deterministic, meaning the same text will always give the same score, making it reproducible. The acknowledged trade-off is lower accuracy on highly domain-specific language compared to a fine-tuned model, but this is also mentiond as a limitation of the work.

3.5.2 Text Field Concatenation

A safe-text helper was applied to each field to replace NaN and malformed values with empty strings. The combined text was formed by concatenating the three fields with a period separator:

```
def _safe_text(x):
    if pd.isna(x): return ''
    x = str(x).strip()
    return '' if x.lower() == 'nan' else x
```

```
5
6  for c in ['pros','cons','summary']:
7      df[c] = df[c].map(_safe_text)
8
9  df['combined_text'] = (df['pros']     + '. '
10                       + df['cons']     + '. '
11                       + df['summary'])
```

Listing 3.9: Building the combined text field.

Afterwards, rows with total text length below 20 characters were removed, with 0 rows being affected, confirming that all the text fields are almost universally non-empty, and certainly not empty when combined.

3.5.3 VADER Sentiment Scoring

VADER was applied via NLTK's `SentimentIntensityAnalyzer` to the combined text field using `tqdm.pandas` progress tracking:

```
1  sia = SentimentIntensityAnalyzer()
2  tqdm.pandas(desc='Scoring sentiment')
3  df['sent_compound'] = df['combined_text'].progress_apply(
4      lambda t: sia.polarity_scores(t)['compound'])
```

Listing 3.10: VADER sentiment scoring with tqdm progress.

Scoring 300,000 reviews took approximately 79 seconds (3,793 reviews/second). Label assignment used the standard VADER thresholds: compound $\geq 0.05 \rightarrow$ positive, compound $\leq -0.05 \rightarrow$ negative, else neutral.

3.5.4 Sanity Check

A sanity check of sentiment labels against rating buckets (low 1–2, mid 3, high 4–5) was computed to validate the sentiment tool. The cross-tabulation reveals whether VADER correctly identifies low-star reviews as negative, mid-star as neutral, and high-star as positive. The Spearman rank correlation coefficient measures the relationship between the continuous compound sentiment score and the star rating. A correlation of 0.46 indicates acceptable validity, confirming that VADER sentiment scores and human-assigned star ratings are reasonably correlated.

3.5.5 Text Length

The period-level descriptive statistics table includes a `mean_text_len` column that reveals a substantial difference in review length across periods: BEFORE reviews average 373 characters, while DURING and POST reviews average approximately 222 characters.

Because VADER's compound score accumulates signal over the full length of the input, shorter texts have fewer opportunities to gain strongly positive or negative scores, compressing compound values towards zero. Consequently, part of the observed sentiment difference across COVID periods may reflect this change in writing length rather than a genuine shift in expressed sentiment. This is treated as a limitation of the combined-field approach.

3.5.6 Statistical Testing

A Kruskal-Wallis H test was applied across the three periods, followed by pairwise Mann-Whitney U tests for BEFORE vs. DURING, DURING vs. POST, and BEFORE vs. POST. Effect sizes were estimated using Cliff's delta, with a 20,000-observation random sample drawn separately for each pair using `numpy.random.default_rng(42)` to ensure reproducibility:

```
def cliffs_delta(a, b, max_n=20000, seed=42):
    a = np.asarray(a)
    b = np.asarray(b)
    rng = np.random.default_rng(seed)
    a_sample = rng.choice(a, size=min(len(a), max_n), replace=False)
    b_sample = rng.choice(b, size=min(len(b), max_n), replace=False)
    gt = np.sum(a_sample[:, None] > b_sample[None, :])
    lt = np.sum(a_sample[:, None] < b_sample[None, :])
    return (gt - lt) / (len(a_sample) * len(b_sample))
```

Listing 3.11: Randomised Cliff's delta estimation.

The three pairwise p-values were corrected for multiple comparisons using the Bonferroni method via `statsmodels.stats.multitest.multipletests`.

3.5.7 Industry-Level Shift Analysis

For each industry with at least 200 reviews in both the BEFORE and POST periods, the difference in mean compound sentiment (POST − BEFORE) was computed. Industries were ranked from most negative to most positive shift, identifying sectors whose employees' expressed sentiment deteriorated or improved most substantially after the pandemic.

3.6 Separate-Field Sentiment Analysis

This notebook scores VADER sentiment on each of the three text fields independently, adds a divergence feature, runs statistical tests, and performs a comprehensive subgroup shift analysis.

3.6.1 Motivation

Combining pros, cons, and summary into a single text inevitably introduces cancellation effects: a strongly positive *pros* field and a strongly negative *cons* field may produce a near-zero compound score when merged. By scoring each field separately, the analysis can detect patterns that cannot be found in a combined analysis.

3.6.2 Independent Field Scoring

Sentiment scoring was applied to each field independently. Rows where no individual field exceeds 10 characters are dropped. Moreover, any field with zero length is scored as NaN rather than being analyzed.

```
for field in text_fields:
    score_col = f"sent_{field}_compound"
    label_col = f"sent_{field}_label"
    len_col = f"{field}_len"

    df[score_col] = np.where(
        df[len_col] > 0,
        df[field].progress_apply(lambda t: sia.polarity_scores(t)["compound"]),
        np.nan
    )
    df[label_col] = df[score_col].map(lambda x: label_from_compound(x) if pd.notna
        (x) else np.nan)
```

Listing 3.12: Independent field scoring with NaN handling.

Scoring all three fields took approximately 80 seconds in total.

3.6.3 Pros-Minus-Cons Divergence Feature

A divergence feature was computed:

$$\text{pros\_minus\_cons}_i = \text{sent\_pros\_compound}_i - \text{sent\_cons\_compound}_i \tag{3.1}$$

This feature measures how much more positive the *pros* language is than the *cons* language for the same review. Values near 1 indicate a review where pros are strongly positive and cons are strongly negative (the typical pattern). Values near zero suggest both fields carry similar sentiment, and negative values show reviews where cons are more positive than pros.

3.6.4 Correlation with Star Ratings

To validate each feature as a signal of overall review sentiment, Spearman rank correlations were computed between each field's compound score and `rating_overall`, including the divergence feature:

```
for field in text_fields:
    rho = df[["rating_overall", f"sent_{field}_compound"]]
            .dropna().corr(method="spearman").iloc[0, 1]
    corr_table.append({"field": field, "spearman_rating_corr": rho})

rho_pmc = df[["rating_overall", "pros_minus_cons"]]
            .dropna().corr(method="spearman").iloc[0, 1]
corr_table.append({"field": "pros_minus_cons", "spearman_rating_corr": rho_pmc})
```

Listing 3.13: Spearman correlation per field including divergence.

3.6.5 Statistical Testing per Feature

For each of the four features (pros, cons, summary, pros_minus_cons), a Kruskal-Wallis H test was applied across all three periods, followed by pairwise Mann-Whitney U tests. Cliff's delta was estimated using a 20,000-observation random sample per pair (seeded at 42). The 12 pairwise p-values were corrected for multiple comparisons using the Bonferroni method, added as a `p_value_bonferroni` column alongside the uncorrected values.

3.6.6 Subgroup Shift Analysis

For each combination of feature and industry, a table was constructed showing mean sentiment in each period. The POST−BEFORE delta was computed for each subgroup entity, and the 10 most negative and 10 most positive shifts were extracted.

This analysis identifies, for example, which industries saw the largest *pros*-sentiment decline between BEFORE and POST COVID.

3.6.7 Output Export

All results were exported to the `outputs-sentiment-separate/` directory:

- `period_stats_fields.csv`: mean/median/std/count by period and field.
- `period_label_share_fields.csv`: sentiment label composition by period and field.
- `tests_fields.csv`: all statistical test results including Bonferroni-corrected p-values.
- `monthly_fields.csv`: monthly means and smoothed trends.
- `subgroup_shift_results.csv`: subgroup shift tables.
- `scored_separate_sample_5000.csv`: 5,000-row audit sample.

- `field_distributions_by_period.png`: distribution boxplots.
- `monthly_trends_by_feature.png`: time series plots.

Chapter 4

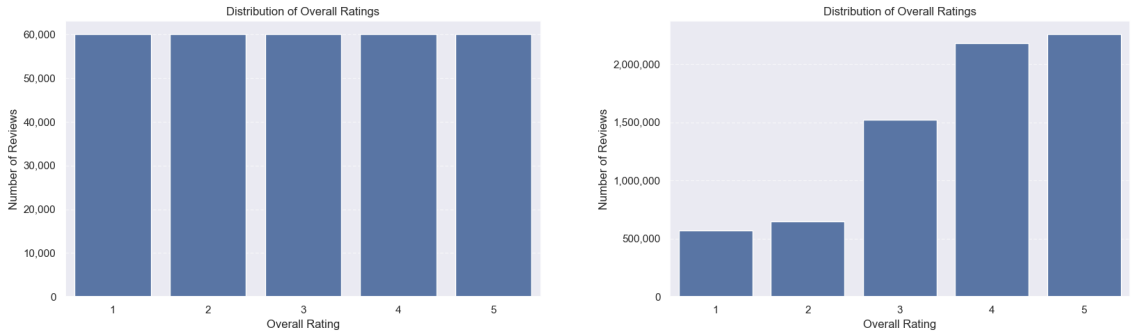
Evaluation

This chapter presents and interprets the results of all analyses. The presentation follows the same logical ordering as the implementation chapter: EDA findings, machine learning model benchmarks, BERTopic topic diagnostics, topic prevalence results, combined sentiment analysis, and separate-field sentiment analysis.

4.1 Exploratory Data Analysis Findings

4.1.1 Rating Distribution

The balanced sampling procedure produced an exactly uniform rating distribution: 60,000 reviews per class. The mean overall rating in the original dataset (before balancing) is substiantially higher and shows that users exhibit a positivity bias. By fixing 60,000 reviews per class, the analysis ensures that all rating levels receive equal representation.



(a) Distribution of ratings after balanced sampling.

(b) Distribution of ratings in original dataset.

Figure 4.1: Distribution of ratings in original and updated dataset.

4.1.2 Temporal Distribution

The temporal distribution shows that the corpus is dominated by reviews from 2016 onwards with a sharp peak around 2021, after the global pandemic declaration and worldwide workplace shifts. There are relatively few (but increasing) observations from 2008–2015. This temporal skew means that "BEFORE" statistics are substantially influenced by the 2016–2020 period, not the full 12-year window.

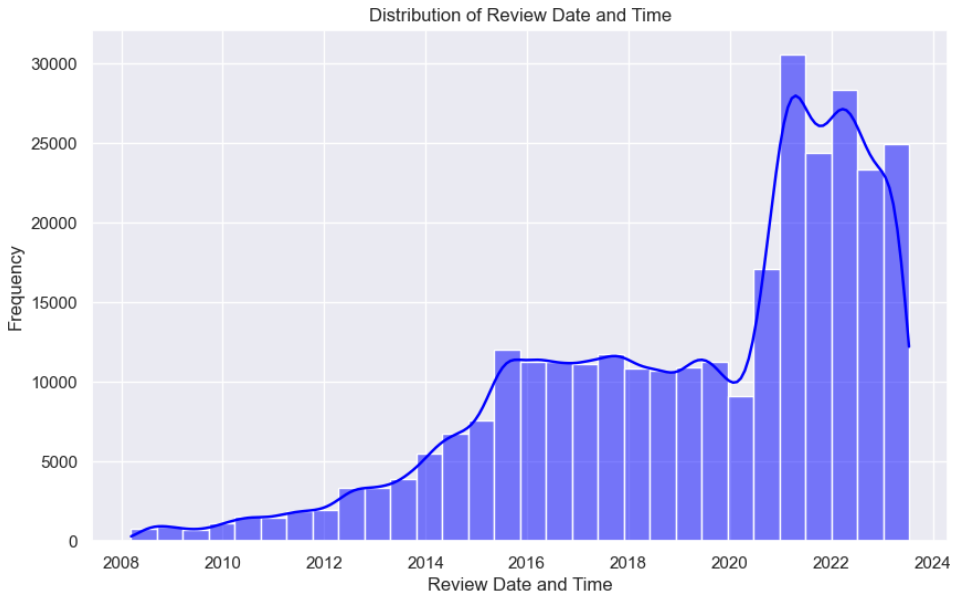


Figure 4.2: Temporal distribution of employee reviews.

4.1.3 Industry Distribution

The dominant industries are tech-related, with Information Technology Support Services, Enterprise Software & Network Solutions, Computer Hardware Development, and Telecommunications Services all featuring prominently.

Beyond tech, the corpus draws heavily from public-facing service sectors including Health Care Services & Hospitals, Colleges & Universities, and Restaurants & Cafes, as well as Banking & Lending and Business Consulting. The over-representation of technology and professional services shows that mostly tech-savvy, white-collar workers are leaving reviews. Workers in other industries such as agriculture are less likely to hold desk jobs, might not have such easy access to platforms for reviews and their employers might not even receive reviews at all. This structural skew means that findings about industry-level patterns might not be easily generalised to sectors outside the industries we mentioned.

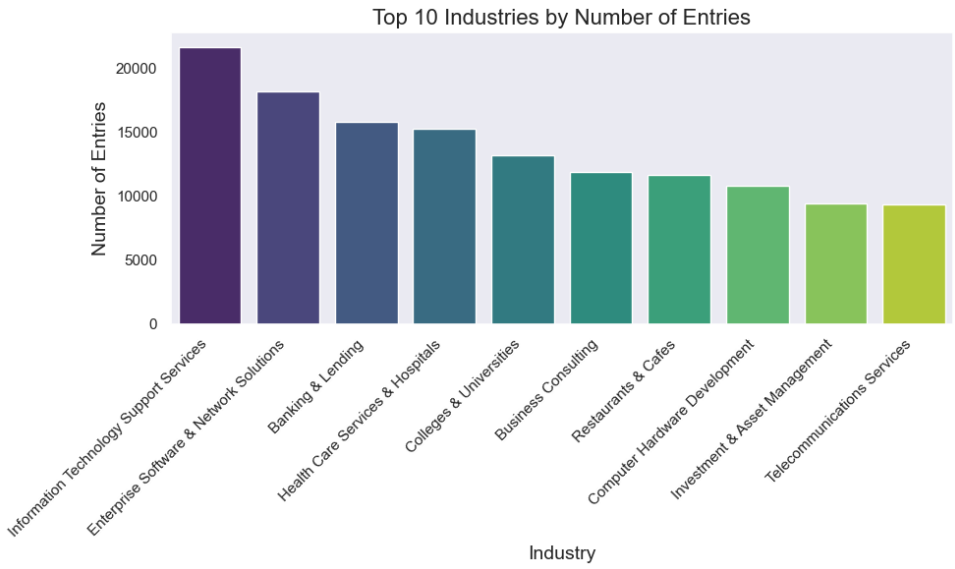


Figure 4.3: Distribution of industries.

4.1.4 Employment Type Distribution

It can be seen that "REGULAR" employment accounts for approximately 87.9% of all reviews, with PART_TIME at 12.0%, and all other statuses at or near 0%. This extreme concentration suggests employment type is a weak discriminator between review classes, which was taken into account in the predictive modelling section.

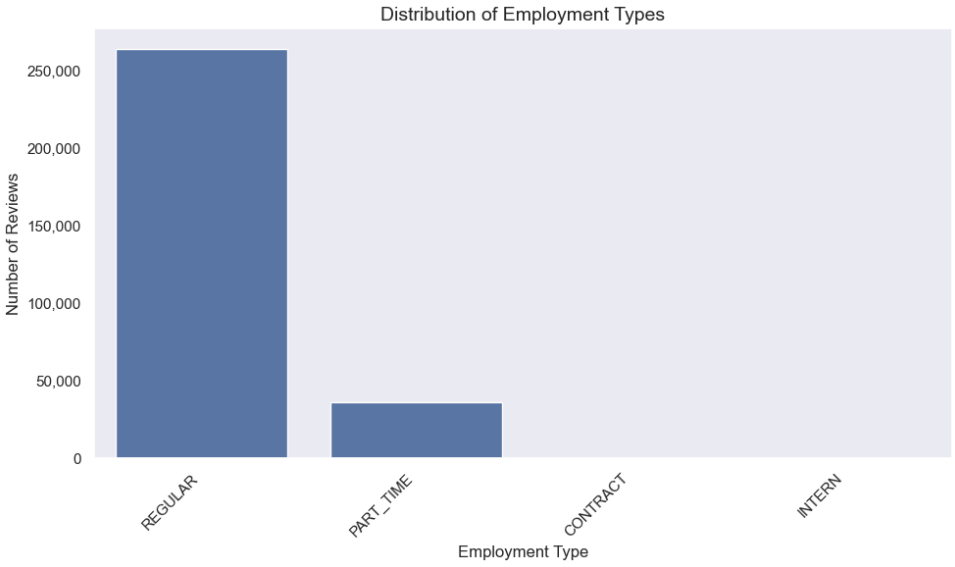


Figure 4.4: Distribution of employment types.

4.1.5 Country Distribution

The United States dominates the corpus by a large margin. Other than that, several other English-speaking and Western European countries appear in the top 10, along with India. The US dominance reflects a disproportionate representation of US technology companies in the dataset. India's presence could likely be a large volume of reviews from eployees at US multinationals operating Indian engineering and service centres, where English is the working language. The very few non-English speaking countries present underlines that this dataset does not accurately represent global workplace experience, but rather the English-speaking segment of it.

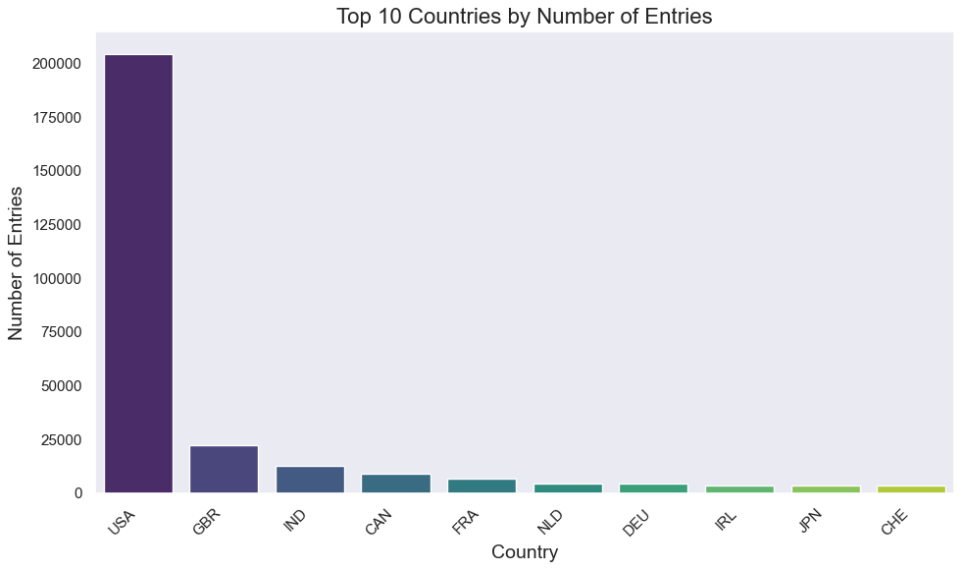


Figure 4.5: Distribution of countries.

4.1.6 Correlation Heatmap

The Pearson correlation matrix of the six sub-ratings showed uniformly positive correlations in the range $[0.36, 0.81]$.

The weakest correlations involved diversity and inclusion, which exhibits only moderate co-variation with the other sub-ratings, suggesting it captures a partially distinct dimension of workplace experience. The positive direction of all correlations reflects a "halo" effect: employees rating one dimension positively tend to view other dimensions positively as well. The relatively high ceiling ($\rho = 0.81$ for culture and leadership) is expected since both factors are influenced by the actions of management. The weaker D&I correlations suggest it captures a more structural and policy-driven dimension of the workplace that can be independent of overall satisfaction.

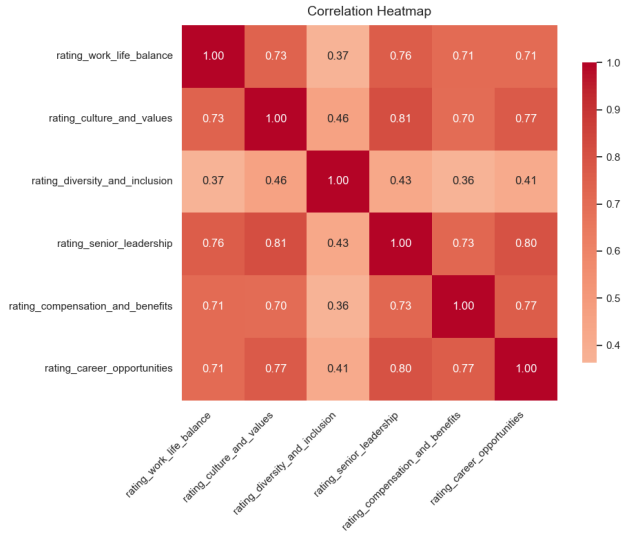


Figure 4.6: Sub-rating correlations heatmap.

4.1.7 Logistic Regression: Stress and Sub-Ratings

The logistic regression for stress prediction revealed two significant predictors:

- **Work-life balance** ($\beta < 0$, $p < 0.001$): increases in work-life balance score correspond to decreases in the log-odds of stress language.
- **Compensation and benefits** ($\beta > 0$, $p < 0.001$): higher compensation ratings are associated with *higher* odds of stress language. This finding is counter-intuitive but possible: high-compensation roles are typically high-responsibility positions in demanding industries, which may generate both financial satisfaction and elevated stress.

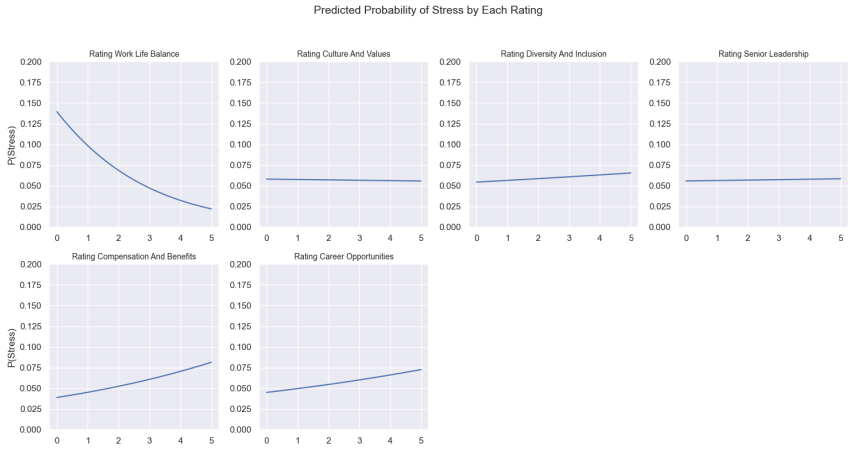


Figure 4.7: Stress prediction outcomes.

4.2 Machine Learning Model Results

4.2.1 Feature Importance Analysis

The LightGBM feature importance analysis revealed that `industryName` is the most important predictor of the overall rating, with an importance score of 1885, substantially higher than the next most important feature, `rating_career_opportunities` (1686). This finding highlights that industry-level differences in working conditions are a dominant driver of satisfaction variation.

Among the six individual sub-ratings, career opportunities, compensation, work-life balance, culture, senior leadership, and diversity rank in order of predictive importance. The imputed categorical ratings (CEO, business outlook, recommend to a friend) carry substantially less predictive power, likely because their coarse 3-level encoding loses much of the original signal.

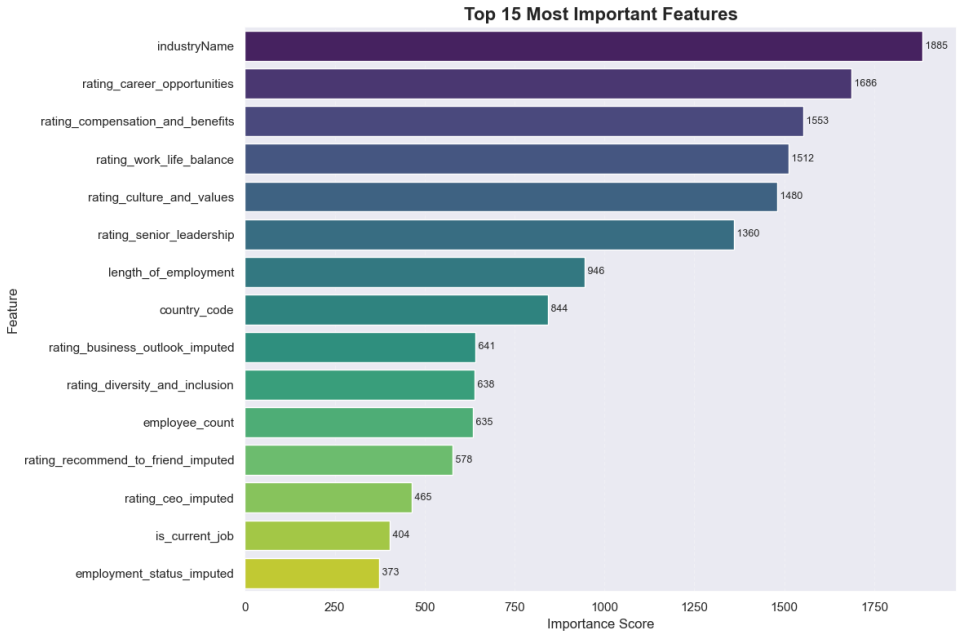


Figure 4.8: Top 15 features.

The dominance of `industryName` is consistent with the well-documented variation in working conditions, compenstions and cultures across industries. A tech company and a restaurant chain operate very differently: wages, work hours, career progression and management are all different by sector. This means that knowing the industry provides the model with knowledge about the likely rating range before any other features are considered.

The four removed features were `employment_status_imputed` (373), `rating_ceo_imputed` (465), `rating_recommend_to_friend_imputed` (578), and

`is_current_job` (404). An important fact we must mention is that the vast majority of reviews were written by full-time employees, while the few that remain were from part-time employees. Very few or no reviews were written by employees considered freelancers, interns, reserve, temporary and self-employed. Thus, even if employment status passed the preliminary screening, it would likely be removed because of the lack of variance.

4.2.2 Baseline 5-Class Results

Table 4.1 summarises the performance of all six baseline models on the 5-class rating prediction task using the original 11 strong features, after hyperparameter tuning.

Table 4.1: Baseline model performance on the 5-class rating prediction task.

Model	**Accuracy**	**F1 (w)**	**Time (s)**
LightGBM	0.5942	0.5959	1.70
CatBoost	0.5938	0.5952	25.50
XGBoost	0.5920	0.5938	1.92
Gradient Boosting	0.5919	0.5939	67.19
Random Forest	0.5567	0.5579	3.60
Logistic Regression	0.4028	0.3718	4.14

The four gradient boosting methods cluster tightly around 59–59.5% accuracy, substantially outperforming Random Forest (55.7%) and Logistic Regression (40.3%). The fact that a linear model achieves only 40% accuracy suggests that the decision boundary for this prediction task is non-linear, consistent with the complex interactions between sub-ratings.

LightGBM achieves the highest accuracy (59.42%) while being among the fastest models to train (1.70 seconds), making it the most efficient choice. CatBoost achieves comparable accuracy but requires about 15× more training time than LightGBM and XGBoost, while Gradient Boosting requires almost 40× more.

4.2.3 Enhanced 3-Class Results

After collapsing to three classes and adding six engineered features, all models showed substantial accuracy improvements (Table 4.2).

The absolute accuracy gain from 5-class to 3-class is approximately 16–17 percentage points across all tree-based models, reflecting the reduction in inter-class confusion

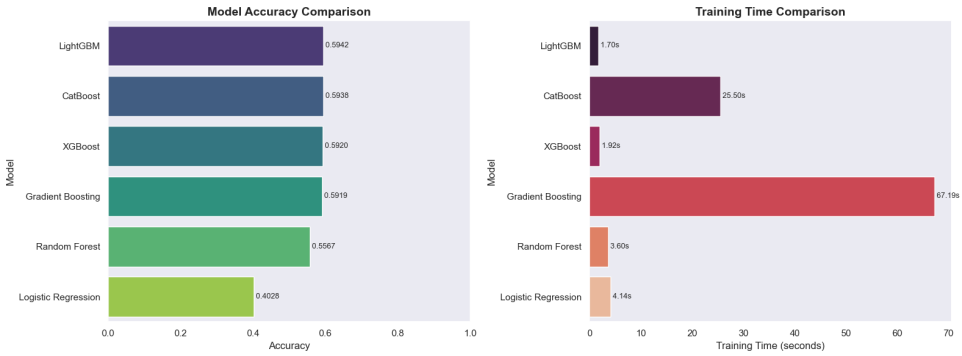


Figure 4.9: Comparison of model accuracy and training time on 5 classes.

Table 4.2: Model performance on the 3-class rating prediction task with engineered features.

Model	**Accuracy**	**F1 (w)**	**Time (s)**
LightGBM	0.7607	0.7509	1.19
XGBoost	0.7604	0.7510	1.21
CatBoost	0.7603	0.7502	18.82
Gradient Boosting	0.7591	0.7488	56.77
Neural Network	0.7592	0.7475	82.12
Logistic Regression	0.7439	0.7239	6.40
Random Forest	0.7330	0.7239	3.88

(particularly between adjacent classes such as 4 and 5, or 1 and 2 stars) when classes are merged.

Notably, the neural network (0.7592 accuracy) performs comparably to the gradient boosting trio, demonstrating that a well-regularised MLP can match specialised tree methods on this tabular task. However, it requires 70× more training time than LightGBM.

4.2.4 Ordinal Regression Results

The three ordinal regression models performed below the rest of the models:

Table 4.3: Ordinal regression model performance on the 3-class task.

Model	**Accuracy**	**F1 (w)**	**Time (s)**
LogisticIT	0.7037	0.6840	2.69
LogisticAT	0.6947	0.7087	3.38
OrdinalRidge	0.6472	0.6799	0.03

LogisticIT achieves 70.4% accuracy, which is 5.7 percentage points below LightGBM. While ordinal models are theoretically better suited to the ordered nature of the target variable, their linear decision boundaries limit performance on this dataset where the relationship between features and rating is non-linear.

4.2.5 Ensemble Results

The three ensemble methods (hard voting, soft voting, and stacking) all produced results nearly identical to the best individual models:

Table 4.4: Ensemble model performance.

Ensemble	Accuracy	F1 (w)
Stacking	0.7614	0.7508
Soft Voting	0.7608	0.7510
Hard Voting	0.7607	0.7509

Stacking achieves the highest accuracy (76.14%), but the improvement over standalone LightGBM (76.07%) is less than 0.1 percentage points, below the likely margin of sampling variability. The three gradient boosting models' high pairwise correlation of predictions limits the diversity gain that ensembles typically provide.

4.3 BERTopic Results

4.3.1 Topic Counts and Diagnostics

Table 4.5 shows the diagnostic metrics for the three BERTopic models.

Table 4.5: BERTopic diagnostic metrics for the three review fields.

Field	Topics	Outlier %	Diversity	Top-10 Cov.
Pros	283	43.2%	0.427	11.7%
Cons	277	40.8%	0.483	13.7%
Summary	521	32.8%	0.565	10.8%

The outlier rates of approximately 40–45% for pros and cons reflect the short nature of these fields, where many individual reviews cannot be confidently assigned to any coherent cluster. The summary field has a lower outlier rate but substantially more topics,

consistent with the fact that it includes more diverse content which usually combines pros and cons.

BERTopic's `reduce_outliers` function was tested. The function assigns all outlier documents to their closest topic cluster, completely removing outliers, however we deemed that this could potentially alter the results BERTopic originally produced and could significantly reduce granularity in the topic assignments, therefore we chose to use the original, non-reduced topic assignments for analysis.

The high outlier rate is also informative on its own. Pros and cons fields are constrained to short, list-style text. Such text produces embeddings that sit in sparse regions of the embedding space, which HDBSCAN correctly identifies as not belonging to any cluster. This is a reflection of the fact that a large share of employee reviews might not focus on a specific thematic at all. Forcing these documents into topics could introduce noise and inflate current clusters without adding any real thematic meaning.

4.3.2 Top Pros Topics

Table 4.6 shows the 10 largest topics discovered in the *pros* field.

Table 4.6: Top 10 BERTopic topics in the *pros* field.

Topic	Name	Count
0	team_teams_members_teamwork	10,166
1	pros_pro_no_only	5,048
2	home_from_work_option	3,743
3	vacation_off_holidays_days	2,793
4	pto_unlimited_generous_holidays	2,537
5	office_office_nice_location	2,422
6	management_managers_manager_middle	2,301
7	training_trainers_program_trainer	2,135
8	campus_university_beautiful_students	1,995
9	brand_name_brands_recognition	1,885

Already, a lot of significant data can be inferred from these results: The most common workplace benefits mentioned positively by employees are teamwork, hybrid or remote work options, vacation, holidays and PTO (paid time-off), good office locations, topics regarding managers, good training/onboarding, and brand name recognition. The second topic (pros_pro_no_only) after manual review seems to include mostly employee reviews that explicitly state "no pros", and the 9th topic is mostly associated with work in public education sectors.

Many more topics with a significant number of assignments were inferred, some of the significant ones including flexibility_flexible_arrangements_environment, insurance_health_medical_coverage, discount_employee_discounts_50, pay_benefits_decent_salary and culture_company_corporate_employee.

4.3.3 Top Cons Topics

Table 4.7 shows the 10 largest topics in the *cons* field.

Table 4.7: Top 10 BERTopic topics in the *cons* field.

Topic	Name	Count
0	cons_con_any_no	9,630
1	training_trainers_was_trainer	4,437
2	promotion_promotions_promoted_based	4,329
3	culture_bad_work_corporate	3,959
4	hr_department_they_employees	3,832
5	balance_life_worklife_work	3,691
6	management_managers_care_manager	3,245
7	students_university_student_faculty	2,927
8	micro_micromanagement_micromanaging_micromanaged	2,627
9	career_advancement_growth_progression	2,438

The cons topics seem almost mirrors of the pros topics: Poor (or lack of) training, slow and difficult promotions, bad company culture, issues with HR (Human Resources), bad work-life balance, discontent with management, micromanagement and lack of career growth and advancements. Other than that, the first topic as before includes reviews that explicity mention “no cons” in the text field, while the 8th topic seems to revolve specifically around teaching positions, with representative topics mentioning non-competitive pay, high teacher turnover and general issues with curriculum inconsistency and lack of expression.

Other topics include call_calls_center_phone which revolves around call center employees, politics_political_office_internal, layoffs_layoff_lay_laid and importantly, india_indian_bangalore_indians which includes discontent with the work culture in India generally, including management, pay and opportunities.

4.3.4 Top Summary Topics

Unlike the previous two sections, topics in the summary field mostly include adjectives and affirmations of the previous fields, along with general thoughts regarding the specific

workplace, such as good_yeah_sometimes_yes and place_work_great_to. Therefore, we decided not to include a full analysis of these topics.

4.4 Topic Prevalence Over Time

4.4.1 Pros Topics: Changes Across Periods

Table 4.8 shows the prevalence of some top pros topics across the three COVID periods, sorted by the POST–BEFORE delta.

Table 4.8: Pros topic prevalence (% of non-outlier docs) across COVID periods.

Topic Name	**BEFORE**	**DURING**	**POST**	**Δ P−B**
team_teams_members_teamwork	4.80	6.75	7.15	+2.35
remote_remotely_fully_work	0.49	0.91	2.03	+1.55
pto_unlimited_generous_holidays	1.37	1.35	1.83	+0.46
home_from_work_option	2.12	2.00	2.51	+0.39
balance_life_worklife_work	0.56	0.82	0.89	+0.33
pros_pro_no_only	3.07	2.76	2.97	−0.10
training_trainers_program_trainer	1.36	1.15	1.17	−0.19
vacation_off_holidays_days	1.80	1.42	1.58	−0.21
office_offices_nice_location	1.65	1.22	1.23	−0.43
401k_match_matching_401	1.42	0.87	0.68	−0.73

The most striking finding in the pros field is the **remote work topic** (remote_remotely _fully_work), which grew from 0.49% of pros mentions BEFORE the pandemic to 0.91% DURING and 2.03% POST, with a total increase of 1.55 percentage points. This reflects the fundamental shift in employee expectations around flexible location policies, with remote work becoming an increasingly prominent positive aspect of employment in the post-pandemic period.

Teamwork and collaboration (team_teams_members_teamwork) was the single largest riser, growing from 4.80% to 7.15% POST (+2.35 pp). This may reflect employees placing greater value on teamwork and collaboration after the disruptions of pandemic-era remote work, and a massively positive reception to good teamwork and collaboration in the workplace.

Conversely, the 401k_match_matching_401 topic fell sharply (from 1.42% to 0.87%, −0.73 pp), as did office_offices_nice_location (−0.43 pp). These declines likely reflect a shift in what employees considered positive: tangible benefits became less important relative to flexibility and team dynamics.

4.4.2 Cons Topics: COVID-Specific Rise and Fall

The single most dramatic topic-level change in the entire analysis is observed in the *cons* field. Topic covid_pandemic_covid19_during appeared in only 0.04% of BEFORE cons mentions, surged to 1.55% DURING the pandemic, and then declined to 0.73% POST, still substantially above pre-pandemic levels.

This topic captured a cluster of reviews explicitly mentioning COVID, the pandemic, and related workplace changes as negative aspects of their employer. The magnitude of this surge (from 0.04% to 1.55%, a 1.51 pp rise) makes it the largest DURING-period single-topic change observed in the cons field.

Other significant DURING-period risers in cons included:

- toxic_environment_culture_work: from 0.50% to 1.35% DURING, then rising further to 1.79% POST. This sustained increase suggests that pandemic conditions accelerated a longer-term deterioration in perceived workplace culture.
- cons_con_any_no: from 4.01% to 6.86% DURING. This captures a possible significant increase in positive sentiment during the pandemic, with a larger share of employee reviews mentioning no negatives at all in regards to their workplace.
- hours_long_sometimes_working: from 0.69% to 1.16% DURING, consistent with increased workload demands during the crisis, and from 1.16% to 0.99% POST, a net increase revealing that employees seem to be facing longer work hours more commonly in general after COVID.

Topics that *declined* during the pandemic include:

- layoffs_layoff_lay_laid: from 1.41% BEFORE to 0.64% DURING. This counterintuitive finding may reflect a "survivor bias" effect: employees who remained at companies through the pandemic's mass-layoff period may have been less likely to cite layoffs as a con, either because they had accepted this as a broader societal phenomenon or because review volume among recently laid-off workers declined.
- senior_seniors_management_seniority: from 1.46% to 0.73% DURING. This may reflect an increase in senior management communication and visibility during the crisis, or a shift in employee priorities toward more immediate concerns.
- hr_department_they_employees: from 2.72% to 1.67% DURING, the largest absolute faller. HR-related complaints may have been less prioritized relative to more urgent pandemic-era concerns.

However, these are all merely speculation and could instead reflect a general change in workplace priorities, however topics such as layoffs_layoff_lay_laid probably fit the earlier theory since there have been significant global layoffs consistently the last decade, especially with the rise of Generative AI (Artificial Intelligence).

Table 4.9 summarises the strongest rises and falls in cons topics during the pandemic.

Table 4.9: Top cons topic changes: DURING minus BEFORE delta (%).

Topic	**BEFORE**	**DURING**	**Δ D−B**
Strongest risers			
cons_con_any_no	4.01	6.86	+2.85
covid_pandemic_covid19_during	0.04	1.55	+1.51
toxic_environment_culture_work	0.50	1.35	+0.85
nothing_everything_bad_good	0.44	1.14	+0.70
hours_long_sometimes_working	0.69	1.16	+0.47
Strongest fallers			
hr_department_they_employees	2.72	1.67	−1.05
layoffs_layoff_lay_laid	1.41	0.64	−0.77
senior_seniors_management_seniority	1.46	0.73	−0.73
training_trainers_was_trainer	2.78	2.06	−0.73
students_university_student_faculty	1.99	1.32	−0.67

4.4.3 Full-Corpus Topic Risers

Further analysis was conducted on the entire corpus rather than the most important and prevalent topics in all three fields, however the results are largely the same. A few worth noting are:

- diversity_diverse_inclusive_inclusion in *pros* field: An increase from 0.42% to 0.76% DURING-POST, revealing an increase in diversity and inclusion generally within the workplace.
- brand_name_brands_recognition in *pros* field: A decrease from 1.38% to 0.76% DURING-POST, revealing that brand name recognition seems to have taken less priority for employees.
- remote_remotely_hybrid_office in *cons* field: An increase from 0.20% to 0.31% DURING, and 0.72% POST, showing that employees value remote work options significantly more POST-COVID and negatively respond to the lack thereof.

- senior_seniors_junior_leadership in *cons* field: A decrease from 1.46% to 0.68% DURING-POST, underlining the fact that issues with senior management seem to be of lower priority in more recent times.

4.5 Combined Sentiment Analysis Results

4.5.1 Methodological Limitations of the Combined-Text Approach

Two structural limitations affect the interpretation of all results in this section and should be kept in mind throughout.

Text length confound. As mentioned previously, BEFORE-period reviews average approximately 373 characters, compared to roughly 222 characters for DURING and POST reviews. Because VADER accumulates positive and negative signal across the entire input string, longer texts tend to produce more extreme compound scores. Any observed difference in mean compound sentiment between BEFORE and the later periods may therefore partly reflect this length difference rather than a true shift in employee sentiment.

Positive-sentiment floor from combining fields. The combined text concatenates the *pros*, *cons*, and *summary* fields into a single string. Because the pros field is structurally positive, the combined text produces an overall positive compound score even for low-rated reviews: reviews with 1–2 star ratings are still labelled *positive* by VADER in 55.1% of cases (Table 4.10). This will be mentioned further later in this work.

Table 4.10: VADER sentiment label by star-rating bucket (row-normalised).

Rating	Positive	Neutral	Negative
1–2 stars	55.1%	3.6%	41.2%
3 stars	80.0%	3.8%	16.2%
4–5 stars	93.1%	2.0%	4.8%

4.5.2 Period-Level Descriptive Statistics

Table 4.11 shows the period-level sentiment statistics for the combined review text.

All three periods exhibit positive mean compound sentiment (all means > 0), consistent with the overall positivity bias of the reviews. The DURING period shows a small upward bump in the mean compound score ($+0.0077$ vs BEFORE), while POST dips slightly below BEFORE. However, the standard deviations are large relative to the mean differences, suggesting substantial overlap between distributions.

Table 4.11: Combined-text VADER compound sentiment by COVID period.

Period	N	Mean	Median	Std	Mean Rating
BEFORE	146,783	0.4095	0.6597	0.5863	2.806
DURING	76,673	0.4172	0.6369	0.5470	3.188
POST	76,544	0.4013	0.6249	0.5562	3.184

Interestingly, mean ratings are substantially higher during and after the pandemic (3.19 vs 2.81) despite the compound sentiment scores showing only minor differences. This could be possibly driven by selection effects (employees who remained employed and were not laid off may have been more satisfied on average) or by genuine appreciation for employer responses to the crisis.

4.5.3 Sentiment Label Composition

Table 4.12 shows the composition of sentiment labels across periods.

Table 4.12: Sentiment label composition (%) by COVID period for combined text.

Period	Positive	Neutral	Negative
BEFORE	74.98	2.82	22.21
DURING	76.23	3.19	20.58
POST	75.01	3.30	21.69

The DURING period shows the highest positive-review share (76.23%) and the lowest negative share (20.58%), while BEFORE and POST are nearly indistinguishable in their label composition. The neutral share increases slightly across periods, possibly reflecting the growing prevalence of short, formulaic reviews. These results are of coursed skewed by the fact that many negative reviews are classified as positive due to the nature of the data.

4.5.4 Statistical Tests

- **Kruskal-Wallis**: $H = 334.49$, $p = 2.33 \times 10^{-73}$. The three distributions are statistically distinguishable.
- **BEFORE vs DURING**: $U = 5.80 \times 10^{9}$, $p = 3.71 \times 10^{-33}$, Cliff's $\delta = 0.0307$.
- **DURING vs POST**: $U = 2.98 \times 10^{9}$, $p = 1.33 \times 10^{-6}$, Cliff's $\delta = 0.0206$.

- **BEFORE vs POST**: $U = 5.86 \times 10^9$, $p = 6.88 \times 10^{-65}$, Cliff's $\delta = 0.0497$.

While all comparisons are highly statistically significant (reflecting the large sample sizes), the Cliff's delta effect sizes are all in the "negligible" range ($|\delta| < 0.147$). This means that although the period distributions differ systematically, the practical magnitude of the sentiment shift is small and the combined review text did not undergo a massive transformation across the COVID periods.

4.5.5 Industry-Level Shifts

Industries with the largest negative POST–BEFORE sentiment shifts include: National Agencies ($\Delta = -0.118$), Sports & Recreation ($\Delta = -0.085$), Hotels & Resorts ($\Delta = -0.077$), Car & Truck Rental ($\Delta = -0.067$), and Accounting & Tax ($\Delta = -0.066$).

Industries with the largest positive POST–BEFORE shifts include: Research & Development ($\Delta = +0.093$), Information Technology Support Services ($\Delta = +0.079$), Education & Training Services ($\Delta = +0.070$), and HR Consulting ($\Delta = +0.040$).

Service-sector industries (hospitality, recreation, government) saw the largest negative sentiment shift, consistent with the known pandemic-related hardships in those sectors. Knowledge and economy industries (R&D, IT, education) improved, possibly reflecting the successful transition to remote and hybrid work arrangements.

4.6 Separate-Field Sentiment Results

4.6.1 Field-Level Descriptive Statistics

Table 4.13 shows the period-level statistics for each text field and the divergence feature.

The field-level analysis reveals different patterns across the three fields:

- **Pros**: compound sentiment declined from BEFORE (0.558) through DURING (0.498) to POST (0.493). Employees expressed slightly less positive sentiment in their pros field during and after the pandemic, even as overall star ratings rose.
- **Cons**: sentiment in the cons field was consistently negative across all periods (mean ≈ -0.09), with very small declines over time. The changes are small.
- **Summary**: compound sentiment *rose* sharply during the pandemic (from 0.110 to 0.184) before declining again POST (0.168). The DURING summary sentiment is approximately 67% higher than BEFORE, suggesting that employees who wrote reviews during the pandemic tended to use more positive or hedged language in their overall assessment summary.

Table 4.13: VADER compound sentiment by COVID period and text field. Mean (Median) values.

Feature	Period	N	Mean	Median
Pros	BEFORE	146,783	0.558	0.659
	DURING	76,673	0.498	0.586
	POST	76,544	0.493	0.586
Cons	BEFORE	146,782	−0.088	−0.015
	DURING	76,673	−0.095	0.000
	POST	76,544	−0.098	0.000
Summary	BEFORE	146,761	0.110	0.000
	DURING	76,601	0.184	0.052
	POST	76,472	0.168	0.000
Pros−Cons	BEFORE	146,782	0.645	0.671
	DURING	76,673	0.593	0.625
	POST	76,544	0.591	0.625

- **Pros-minus-Cons**: the divergence declined from 0.645 BEFORE to 0.591 POST. However, this feature has a Spearman correlation of only 0.007 with the overall star rating, meaning it is non-discriminative. The period-level change is statistically significant for BEFORE vs DURING and BEFORE vs POST only due to the large sample size, not because the feature carries practical meaning. Mean `pros_minus_cons` by rating bucket (low(1–2): 0.606, mid(3): 0.636, high(4–5): 0.621) is not monotonically increasing, further confirming that the feature does not provide meaningful results.

4.6.2 Sentiment Label Composition by Field

Table 4.14 shows the label composition for each field.

The summary field undergoes the most dramatic shift: positive labels rise from 36.85% BEFORE to 50.13% DURING (an increase of 13.28 percentage points), while neutral labels fall from 45.50% to 33.45%. This suggests that during the pandemic, reviewers were less likely to write a short, neutral-tone summary and more likely to express clear positive or negative opinions.

In the pros field, the positive share declines from 84.55% to 81.33%, and the neutral share increases from 10.53% to 13.26%. This pattern suggests that pros language became vaguer or less positive overall.

The cons field shows a slight shift towards neutrality during the pandemic (neutral

Table 4.14: Sentiment label composition (%) by field and COVID period.

Field	Period	Positive	Neutral	Negative
Pros	BEFORE	84.55	10.53	4.92
	DURING	81.84	13.14	5.02
	POST	81.33	13.26	5.41
Cons	BEFORE	32.58	18.23	49.19
	DURING	26.80	26.10	47.10
	POST	26.79	25.62	47.59
Summary	BEFORE	36.85	45.50	17.65
	DURING	50.13	33.45	16.41
	POST	47.66	35.39	16.95

share rises from 18.23% to 26.10%), a theory we have for that is many pandemic- related cons were factual descriptions ("mandatory remote work," "no in-person events") rather than affectively charged complaints, and VADER could not capture the actual sentiment of these phrases.

4.6.3 Correlation with Star Ratings

Spearman rank correlations between each field's compound score and `rating_overall` confirm that the summary field is the strongest signal ($\rho = 0.527$), followed by pros ($\rho = 0.274$) and cons ($\rho = 0.218$). The divergence feature provides a negligible signal ($\rho = 0.007$). The weak correlations for pros and cons are expected: pros language is mostly positive regardless of star rating, and cons language is mostly negative.

4.6.4 Statistical Tests per Field

Table 4.15 summarises the statistical test results.

Note: A positive Cliff's δ for pros indicates BEFORE was higher than DURING/POST (pros declined); a negative δ for summary indicates DURING was higher than BEFORE (summary rose). Signs follow the convention $\delta = P(a > b) - P(b > a)$ where a is the first named period.

Key observations from Table 4.15:

1. **Pros field**: the BEFORE vs POST comparison yields the largest Cliff's δ (0.134, approaching the "small" threshold of 0.147), confirming a substantive decline in pros sentiment over the full pandemic period. Notably, the DURING vs POST

Table 4.15: Statistical test results for VADER compound sentiment by field.

Feature	Pair	p-value	Cliff's δ
Pros	BEF vs DUR	< 0.001	$+0.121$
	DUR vs POST	0.132	$+0.010$
	BEF vs POST	< 0.001	$+0.134$
Cons	BEF vs DUR	0.500	$+0.006$
	DUR vs POST	0.087	$+0.004$
	BEF vs POST	0.281	$+0.007$
Summary	BEF vs DUR	< 0.001	-0.116
	DUR vs POST	< 0.001	$+0.022$
	BEF vs POST	< 0.001	-0.100
Pros-Cons	BEF vs DUR	< 0.001	$+0.047$
	DUR vs POST	0.936	$+0.004$
	BEF vs POST	< 0.001	$+0.054$

comparison is not significant ($p = 0.132$), meaning the decline in pros sentiment had already largely occurred by the DURING period and did not deepen further.

2. **Cons field**: no comparison is statistically significant, and all effect sizes are negligible ($|\delta| < 0.008$, Kruskal-Wallis $p = 0.261$). This is a key finding: despite the topic analysis showing a major surge in COVID-related cons topics, the sentiment of cons text was unchanged.

3. **Summary field**: the largest effects in the analysis, with the DURING period showing substantially elevated positivity (Cliff's $\delta = -0.116$ for BEFORE vs DURING, meaning DURING was more positive). The POST period decreases slightly but remains above BEFORE levels ($\delta = -0.100$ for BEFORE vs POST).

4. **Pros-minus-Cons**: statistically significant for BEFORE vs DURING and BEFORE vs POST ($\delta \approx 0.05$), but the DURING vs POST comparison is not significant ($p = 0.936$). Given the near-zero correlation of this feature with star ratings (Spearman $\rho = 0.007$), the practical meaning of these shifts is limited.

4.6.5 Industry-Level Subgroup Shifts

Industry-level subgroup analysis was conducted for each feature to identify which sectors showed the largest POST–BEFORE sentiment shift.

For the **pros** field, sentiment declined across all industries. The largest drops were in Internet & Web Services ($\Delta = -0.108$), Beauty & Personal Accessories Stores ($\Delta = -0.104$), and Wholesale ($\Delta = -0.103$), possibly because service and industries were influenced more severely by the pandemic. The smallest declines occurred in Research & Development ($\Delta = -0.012$) and Information Technology Support Services ($\Delta = -0.013$), suggesting knowledge-economy sectors were influenceed less.

As for the **cons** field, National Agencies ($\Delta = -0.058$), Aerospace & Defense ($\Delta = -0.058$), and Cable, Internet & Telephone Providers ($\Delta = -0.061$) saw the largest increases in cons negativity POST. Education & Training Services ($\Delta = +0.068$) and Publishing ($\Delta = +0.046$) showed the largest improvements. These results should be interpreted cautiously given the overall null result for cons at the period level.

4.6.6 Pros-Cons Label Agreement

The cross-tabulation of pros and cons labels revealed the following:

- 39.15% of reviews reflect the expected pattern: *positive pros / negative cons*.
- 25.34% show *positive pros / positive cons* (employees found something good to say in both fields).
- 18.55% show *positive pros / neutral cons*.
- Only 1.31% show *negative pros / positive cons* (an unusual inversion likely indicating a confused or ironic review).

4.6.7 Discussion of Field-Level vs Combined Analysis

The comparison between combined and field-level results demonstrates the value of preserving the structural separation of review fields:

1. The combined analysis showed a slight upward trend in mean compound sentiment DURING the pandemic (+0.0077). This positive bump is driven by the *summary* field, which rose sharply in positivity. The reason *pros* declined but *summary* rose could mean that listing specific positive attributes could have been harder to name during this period of disruption, whereas the *summary* field usually involves a final verdict. Employees who retained their jobs through the pandemic may have framed their summary in terms of gratitude for keeping their jobs through these times, instead of workplace quality.

2. The separate-field analysis revealed that this aggregate positive bump co-existed with a decline in pros sentiment. Employees were writing less positive content in the pros field even as their summary tone became more positive.

3. The cons field showed no significant sentiment change, meaning the increase in COVID-related cons topics was driven by a shift in topic (what employees complained about) rather than a change in emotional intensity.

4.6.8 Sentiment Overtime

Finally, Figure 4.10 shows the overall sentiment levels for each of the fields mentioned. The main section we should focus on is the *summary* field which seems to capture a more complete image of the evolution of sentiment since it usually includes an overall opinion of the reviewer regarding the workplace, without being forced to follow "guidelines" such as including positives or negatives, like the *pros* and *cons* fields.

We can see that sentiment was on a steady decline up until COVID, where it experienced a sudden spike, and has been on decline again since then. Moreover, we can see a slight increase during the early 2010s, which could theoretically reflect the recovery from the Great Recession and the stabilization of the US economy.

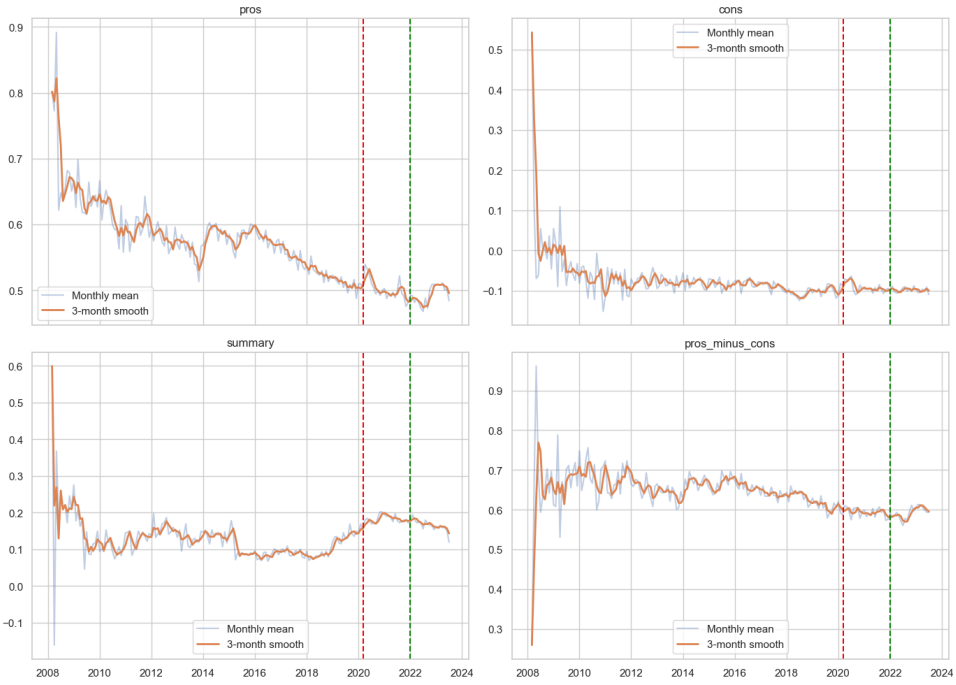


Figure 4.10: Sentiment overtime for all fields.

4.7 Summary of Key Findings

The following are some of the most important findings across all analyses:

1. **Gradient boosting dominates rating prediction.** LightGBM, XGBoost, and CatBoost achieve near-identical performance on the 3-class task (≈76% accuracy), performing better than Random Forest, ordinal regression, and Logistic Regression. The neural network is competitive but not superior. Ensemble methods provide no meaningful gain over standalone gradient boosting in this specific use case.

2. **Industry is the most important predictor.** `industryName` outranks all numerical sub-ratings in feature importance, possibly confirming that industry-level differences in working conditions are a dominant driver of overall satisfaction variation.

3. **Remote work is the defining post-pandemic pros shift.** The remote work pros topic (remote_remotely_fully_work) grew from 0.47% to 2.02% of pros mentions POST (a more than fourfold increase) confirming the normalisation of remote and hybrid work as a valued employment benefit.

4. **COVID-19 became the dominant DURING-period cons topic.** Unsurprisingly, the covid/pandemic topic surged from near-zero (0.04%) to 1.55% of cons mentions DURING, being the largest single DURING-period topic change in the cons field before declining to 0.73

5. **Pros sentiment declined, summary sentiment rose.** Despite small aggregate effects in the combined-text analysis, separate-field sentiment analysis reveals a divergence: pros language became less positive over the pandemic, while summary language became more positive. Cons sentiment remained largely unchanged.

6. **Effects are statistically significant but practically modest.** All period-level differences in sentiment are confirmed by non-parametric tests at very high significance levels ($p < 0.001$), but Cliff's delta effect sizes are small (≤ 0.14), indicating gradual rather than dramatic shifts.

Chapter 5

Related Work

This chapter surveys prior research related to the methods and questions addressed.

5.1 Employee Review Analysis

5.1.1 Remote Work

A comprehensive analysis of remote work and how it fares across different organisations was conducted by Chandra et al. [31], motivated by the alterations caused by the COVID-19 pandemic. In this work, the authors compiled multiple companies from various credible sources such as the Fortune 500 list for the year 2021 and constructed two lists, Desirable and Less-Desirable Organizations, based on whether they could find these organizations recognized as good companies in regards to remote work during the pandemic or not. They then gathered over 140,000 reviews consisting of extensive company-specific data, spanning early 2019 to early 2021, capturing both the pre-COVID and peri-COVID era. They also added additional information, mainly sentiment score on the *pros* and *cons* section of each review, using a fine-tuned XLM-RoBERTa transformer model. Using this dataset, they built an algorithmic prediction task to identify the pre-pandemic factors that predict a good remote work environment, achieving an accuracy of 76% and F-1 score of 73%. They adopted a decomposition of work culture into 41 dimensions across different groups, such as interests, work values and more. The task revealed that factors such as good work-life balance, and work culture pre-pandemic could be correlated with successful remote work adoption during the pandemic, while less-desirable organisations seemed to have signs of bad management and general dissatisfaction pre-pandemic.

5.1.2 Former and Current Employee Sentiment

An investigation on how former and current employees differ in emotional attitudes towards four companies (Amazon, Apple, Microsoft, Google) was conducted by Pan et al. [32]. Using a dataset of 41,000 reviews, the authors split the employees into former and current groups. Then, they analyzed ratings on multiple dimensions to compare scores per company. Additionally, review text (*pros*, *cons*, *summary*) is also analyzed to find which factors each group focuses on. They find that current employees give higher scores on average than former employees, and they usually prioritize work-life balance and senior management, and former employees prioritize senior management and culture & values. Both groups show discontent in regards to the senior management.

5.1.3 Sentiment Classification

Gaye et al. [33] proposed a framework for sentiment-analysis, for classifying reviews by employees from the "world's big six' companies into positive or negative classes. The authors built a large dataset by combining four fields per review: *pros*, *cons*, *summary*, and *advice to management*. After prepocessing and assigning a polarity score to every review using the TextBlob lexicon, they used the results as ground truth for a supervised learning stage. Three methods were used to represent the text: TF-IDF, bag-of-words and GloVe word embeddings. Seven standard classifiers were then trained. The paper introduces a hybrid/voting ensemble called Regression Vector-Stochastic Gradient Descent Classifier (RV-SGDC) which bomines logistic regression, SVC and SGDC in hard voting. The framework was also tested on the Twitter US Airline Sentiment dataset, achieving 0.97 accuracy.

5.1.4 Sentiment during COVID-19

Pant et al. [34], after conducting literature review on sentiments, tools and employee wellbeing and building a conceptual framework, applied a sentiment-analysis approach to survey data collected from employees located in India and abroad, who worked during the COVID-19 pandemic lockdowns. The analysis distinguishes between sentiment and specific emotions, and quantifies how often participants reported different emotions about their work. The authors found that many participants felt excited, seeing the pandemic as an opportunity and a challenge, but many others felt negative emotions linked to loneliness and uncertainty.

Chapter 6

Conclusion

6.1 Summary of Contributions

This thesis presented a comprehensive study of exploratory data analysis, model comparison, workplace sentiment and topic evolution in employee reviews across three COVID-19 periods (Before, During and Post), using a balanced dataset of 300,000 employee reviews. The work addressed five main research questions.

Addressing RQ1 (Exploratory Data Analysis): A comprehensive analysis of the employee reviews revealed that employee reviews are biased towards positive ratings. Moreover, the vast majority of reviews were written during and after COVID, while the most dominant industries seem to be tech related. Most reviewers are regular, full-time employees and the vast majority of them are located in the USA. Additionally, sub-ratings within reviews seem to be mostly positively correlated, with diversity and inclusion being the only seemingly isolated factor in the reviews. Finally, stress levels are predicted to increase with higher compensation and benefits, reflecting more significant responsibilities, and better work-life balance directly decreases stress levels.

Addressing RQ2 (Rating Prediction): A model comparison showed that gradient boosting methods (LightGBM, XGBoost, CatBoost) outperform both linear and simpler non-linear models on this specific rating prediction task. The best accuracy on the 5-class task is approximately 59.4%, increasing to 76.1% after collapsing to 3 classes and adding six engineered features. The most important predictor was `industryName`, theoretically confirming that industry-level variation heavily affects satisfaction levels. For this specific dataset, ensemble methods provide virtually no gains over the best individual model, while ordinal regression models perform well but worse than gradient boosting models.

Addressing RQ3 (Topic Discovery): Three BERTopic models were successfully trained on the *pros*, *cons*, and *summary* fields, finding hundreds of topics per field. The pros model revealed multiple themes around compensation and benefits, teamwork and

collaboration, remote work, flexible scheduling, and travel opportunities. The cons model highlighted frustrations with promotion opportunities, HR policies, micromanagement, training quality, poor career growth opportunities and low pay. The summary model, trained on shorter and more varied text, produced a larger number of topics with less thematic coherence.

Addressing RQ4 (Topic Prevalence Over Time): The topic prevalence analysis resulted in several important findings. The remote work pros topic rose from significantly across the study period (the largest single-topic shift observed). The COVID-specific cons topic surged DURING before partially retreating POST, showing a very clear pandemic-specific focus. Brand naame recognition declined sharply as a mentioned pro. Layoffs-related cons decreased during the pandemic, while toxic environments and long working hours as cons topics rose.

Addressing RQ5 (Sentiment Changes): The combined-text VADER analysis showed statistically significant but practically small period-level differences. The separate-field analysis revealed a more nuanced and divergent picture: *pros* sentiment declined, *summary* sentiment rose sharply during the pandemic, and *cons* sentiment showed no significant change. These field-level patterns would have been hidden in a combined-text analysis, demonstrating the value of the separate-field approach.

6.2 Limitations

Several limitations of this work should be acknowledged.

Selection bias. The dataset includes several layers of selection bias. Employees who feel strongly (positively or negatively) are more motivated to write a review than employees with neutral experiences, which likely inflates the representation of extreme sentiment. Moreover, the majority of employees seem to be white-collar, English-speaking workers in technology, finance and services. Sectors like manufacturing, agriculture and more, are under-represented. These factors mean that findings should be interpreted as reflecting the population of employees that chose to write a review and not the global workforce. Since the dataset is dominated by US-based workers, findings may not generalize to other countries and cultures where workplace norms are different.

Tenure bias. Reviews written by former employees may reflect exit experiences more than typical day-to-day conditions. Conversely, current employees may not be feel completely free to expresse their thoughts out of concern for professional reputations. The dataset contains a mix of both groups, and their motivations for writing differ systematically.

VADER's limitations. VADER is a lexicon-based tool optimised for social media text. It does not handle domain-specific language and sarcasm well. The Spearman corre-

lation between compound scores and star ratings indicates acceptable validity but a lot of noise. A fine-tuned transformer-based sentiment model would likely produce more accurate field-level scores, at the cost of computational resources, however it was not feasible within the time frame of this work.

Unequal period sizes. The BEFORE period spans 12 years and contains almost twice as many reviews as DURING or POST. This imbalance means that BEFORE statistics are averages across a long window, while DURING and POST are tighter, within a shorter window. Research focused on specific sub-periods (e.g., the first three months of the pandemic vs. 2021) should probably use narrower windows.

Causal inference. All period-level comparisons are observational. Without a control group unaffected by COVID-19 (which is impossible in this case), results are speculative. Observed changes may reflect a combination of COVID effects, time trends, and survivorship bias (workers who retained employment through the pandemic may differ from those who left).

BERTopic outlier rate. Almost half of the *pros* and *cons* documents were assigned to the outlier cluster (topic –1). The topic prevalence analysis was computed only on non-outlier documents, meaning the findings reflect a subset of the dataset: those reviews with coherent enough text to be assigned to a specific cluster.

Topic model stability. BERTopic results can vary across runs due to the components of UMAP and HDBSCAN, even with a fixed random seed. While `random_state=42` was set throughout, exact reproducibility was not possible and small variations within the results existed each time.

6.3 Future Work

Several directions could extend and strengthen the findings of this thesis.

Transformer-based sentiment analysis. Replacing VADER with a fine-tuned model (e.g., a fine-tuned RoBERTa model) could produce more semantically correct sentiment scores, capturing sarcasm and domain-specific language more accurately.

Dynamic topic models. Replacing the static BERTopic + temporal slicing approach with a dynamic topic model would allow topics themselves to evolve over time, potentially revealing shifts that are invisible to the fixed-topic prevalence approach.

Multilingual extension. Extending the analysis to non-English employee reviews could test whether the observed COVID-period patterns are universal or specific to English-speaking workplaces.

Company-level analysis. Aggregating results for each company could allow an analysis of how individual companies' review profiles evolved, potentially identifying which companies' employees' sentiment evolved most positively or negatively to the pandemic

and overtime.

Integration with macroeconomic indicators. Correlating monthly topic prevalence and sentiment trends with external macroeconomic time series (unemployment rates, remote work adoption statistics, infection rates) could find the association between labour market conditions and employee review content more accurately.

6.4 Closing Remarks

Work is a significant part of everyone's lives. By researching and studying the factors that affect employee satisfaction within the workplace, and analyzing sentiment throughout different periods, it's possible to accurately determine what good work culture looks like and how it positively affects the lives of people. With the COVID-19 pandemic, large transformations occurred in the employment landscape, and multiple patterns, such as the rise of remote work, have been uncovered. Painting a full picture of how employees' experiences have evolved through it could prove an invaluable source of information for the future of employment in general.

This work demonstrates that employee reviews, when analyzed with the appropriate methods, represent a very valuable source of information. As analysis methods continue to advance, the potential for real-time understanding of conditions within the workplace from employee reviews will increase.

Bibliography

[1] World Health Organization. WHO Director-General's Opening Remarks at the Media Briefing on COVID-19 – 11 March 2020. *WHO Press*, 2020. `https://www.who.int`.

[2] Maarten Grootendorst. Bertopic: Neural topic modeling with a class-based tf-idf procedure. *arXiv preprint arXiv:2203.05794*, 2022.

[3] Clayton Hutto and Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the international AAAI conference on web and social media*, volume 8, pages 216–225, 2014.

[4] Jerome H Friedman. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232, 2001.

[5] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[6] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[7] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

[8] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31, 2018.

[9] Peter McCullagh. Regression models for ordinal data. *Journal of the Royal Statistical Society: Series B (Methodological)*, 42(2):109–127, 1980.

[10] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, pages 448–456. pmlr, 2015.

[11] David H Wolpert. Stacked generalization. *Neural networks*, 5(2):241–259, 1992.

[12] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

[13] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 3982–3992, 2019.

[14] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[15] Leland McInnes, John Healy, Steve Astels, et al. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, 2(11):205, 2017.

[16] Bo Pang and Lillian Lee. *Opinion mining and sentiment analysis*. Now Publishers Inc, 2008.

[17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pages 4171–4186, 2019.

[18] William H Kruskal and W Allen Wallis. Use of ranks in one-criterion variance analysis. *Journal of the American statistical Association*, 47(260):583–621, 1952.

[19] Henry B Mann and Donald R Whitney. On a test of whether one of two random variables is stochastically larger than the other. *The annals of mathematical statistics*, pages 50–60, 1947.

[20] Norman Cliff. Dominance statistics: Ordinal analyses to answer ordinal questions. *Psychological bulletin*, 114(3):494, 1993.

[21] Tomaz Cajner, Leland D Crane, Ryan A Decker, John Grigsby, Adrian Hamins-Puertolas, Erik Hurst, Christopher Kurz, and Ahu Yildirmaz. The us labor market during the beginning of the pandemic recession. Technical report, National Bureau of Economic Research, 2020.

[22] Matthias Pierce, Holly Hope, Tamsin Ford, Stephani Hatch, Matthew Hotopf, Ann John, Evangelos Kontopantelis, Roger Webb, Simon Wessely, Sally McManus, et al.

Mental health before and during the covid-19 pandemic: a longitudinal probability sample survey of the uk population. *The Lancet Psychiatry*, 7(10):883–892, 2020.

[23] Wes McKinney et al. Data structures for statistical computing in python. *scipy*, 445(1):51–56, 2010.

[24] Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. Array programming with numpy. *nature*, 585(7825):357–362, 2020.

[25] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32, 2019.

[27] Skipper Seabold, Josef Perktold, et al. Statsmodels: econometric and statistical modeling with python. *scipy*, 7(1):92–96, 2010.

[28] John D Hunter. Matplotlib: A 2d graphics environment. *Computing in science & engineering*, 9(3):90–95, 2007.

[29] Fabian Pedregosa. mord: Ordinal regression in python. `https://pythonhosted.org/mord/`, 2014. Version 0.6.

[30] Federico Bianchi, Valerio Di Carlo, Paolo Nicoli, and Matteo Palmonari. Compass-aligned distributional embeddings for studying semantic differences across corpora. *arXiv preprint arXiv:2004.06519*, 2020.

[31] Mohit Chandra and Munmun De Choudhury. What makes some workplaces more favorable to remote work? unpacking employee experiences during covid-19 via glassdoor. In *Proceedings of the 15th ACM Web Science Conference 2023*, pages 312–323, 2023.

[32] Xiao Pan, Wenli Du, and Zhuoming Ren. Capturing the difference between former and current employees' emotional attitudes based on the online. In *Proceedings of the 2nd international conference on internet, education and information technology (IEIT 2022)*, 2023.

[33] Babacar Gaye, Dezheng Zhang, and Aziguli Wulamu. Sentiment classification for employees reviews using regression vector-stochastic gradient descent classifier (rv-sgdc). *PeerJ Computer Science*, 7:e712, 2021.

[34] Sudhir Kumar Pant and Manjari Agarwal. A study of sentiments of employees during covid-19. *Telecom Business Review*, 2021.