Athens University of Economics & Business

Machine Learning & Content Analytics

Main Assignment

ΠΑΠΑΔΑΤΟΥ-ΓΙΓΑΝΤΕ ΙΩΑΝΝΑ
28-09-2025

Contents

Introduction

In recent years, online reviews have become an important source of information for both customers and companies. Platforms such as Glassdoor allow employees to share their experiences about their workplace, covering aspects such as culture, career opportunities, management, benefits, and work–life balance. For companies, these reviews are a valuable source of feedback, but the large volume of unstructured text makes it difficult to extract useful insights manually.

Natural Language Processing (NLP) and Machine Learning (ML) methods provide an effective way to analyze such data. One popular approach is aspect-based sentiment analysis, where the goal is not only to identify whether a review is positive or negative, but also to detect which aspect of the company the sentiment refers to. This helps businesses understand employee opinions more precisely and supports evidence-based decision-making in areas like human resources, management practices, and company culture.

Our Project

This project focuses on applying content analysis and machine learning methods to a large collection of employee reviews taken from the Glassdoor platform. The reviews provide a rich source of information about employees' experiences, covering different aspects such as salary and benefits, career development opportunities, company culture, management practices, and work–life balance. However, because reviews are written in free text, they are difficult to analyze systematically without computational tools.

The approach was to design and implement a full analysis pipeline that could transform this unstructured text into structured insights. The process began with cleaning and normalizing the raw text, followed by the identification of sentences that refer to specific aspects of the employee experience. For each identified aspect, a sentiment classification step was applied, using a transformer-based model, in order to determine whether the expressed opinion was positive, neutral, or negative. Finally, we aggregated the results to generate a higher-level view of employee perceptions.

Our Goals

The goals of this project are both technical and business-oriented. On the technical side, the focus is placed on demonstrating how modern Natural Language Processing (NLP) methods, and in particular transformer-based models such as DistilBERT, can be applied to aspect-based sentiment analysis in a real-world dataset. The intention is to construct a complete pipeline capable of performing text cleaning, identifying relevant aspects, classifying sentiment with reliable accuracy, and evaluating performance through widely accepted metrics including precision, recall, F1-score, and accuracy.

On the business side, the project aims to generate insights that can support a better understanding of employees' perspectives. By identifying which aspects of the workplace are most frequently discussed and by quantifying the polarity of the sentiment attached to them, organizations can obtain a clearer view of their strengths and weaknesses. For instance, consistently negative sentiment regarding management alongside positive sentiment about benefits highlights both areas requiring attention and areas where the company is already performing well. In this way, the analysis contributes to evidence-based decision making for improving employee satisfaction and organizational performance.

Methodology

This section describes the methodological framework followed in the project. The process began with the selection of an appropriate dataset and continued with several stages of data processing and normalization. Aspect extraction and sentiment annotation were then applied to prepare the data for supervised learning. Finally, a transformer-based architecture was fine-tuned to classify sentiment at the aspect level. The following subsections present the main steps in detail.

Data Collection

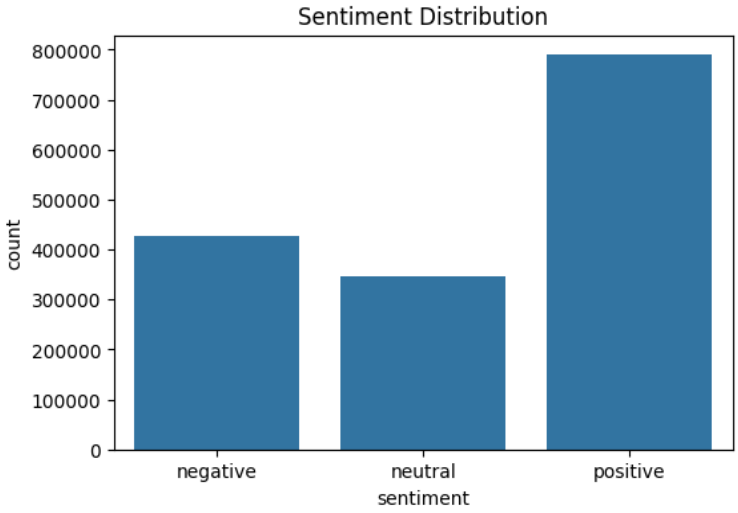
The dataset was obtained from the Kaggle platform and contains over 830.000 employee reviews collected from Glassdoor. Each review includes both structured and unstructured information, such as company name, job title, location, review headline, pros and cons, and date of submission, along with numerical ratings for overall satisfaction and aspect- specific ratings for categories including work–life balance, culture and values, career opportunities, compensation and benefits, and senior management. The availability of both free-text reviews and numerical ratings makes the dataset particularly suitable for supervised machine learning tasks in sentiment analysis.

Dataset Overview

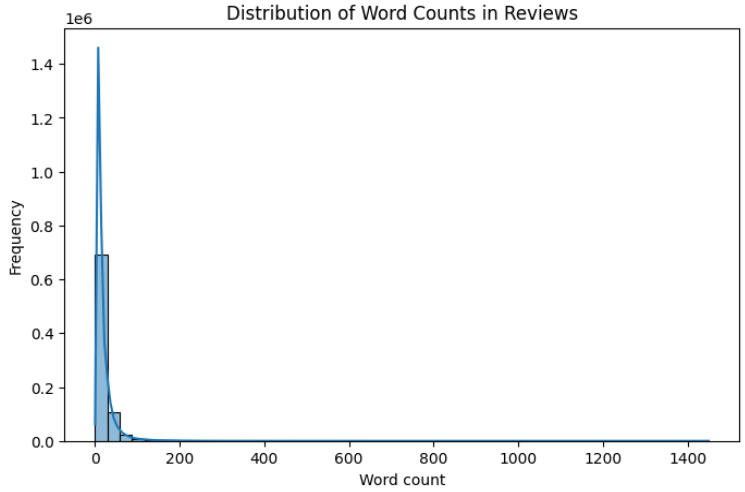
The raw dataset contained 838.566 overall ratings, with slightly fewer available entries for each aspect, 688.672 for work–life balance, 647.193 for culture and values, and 136.066 for diversity and inclusion. Descriptive statistics show that the average overall rating is approximately 3,65 out of 5, while aspect ratings follow a similar distribution: work–life balance has a mean of 3,38, culture and values 3,59, career opportunities 3,46, compensation and benefits 3,40, and senior management 3,18. The relatively lower score for senior management compared to other aspects highlights a consistent area of employee dissatisfaction.

Statistic	overall_rating	work_life_balance	culture_values	diversity_inclusion	career_opp	comp_benefits	senior_mgmt
count	838,566	688,672	647,193	136,066	691,065	688,484	682,69
mean	3.66	3.38	3.59	3.97	3.46	3.40	3.18
std	1.17	1.31	1.32	1.19	1.27	1.22	1.33
min	1.00	1.00	1.00	1.00	1.00	1.00	1.00
25%	3.00	2.00	3.00	3.00	3.00	3.00	2.00
50%	4.00	4.00	4.00	4.00	4.00	4.00	3.00
75%	5.00	4.00	5.00	5.00	5.00	5.00	4.00
max	5.00	5.00	5.00	5.00	5.00	5.00	5.00

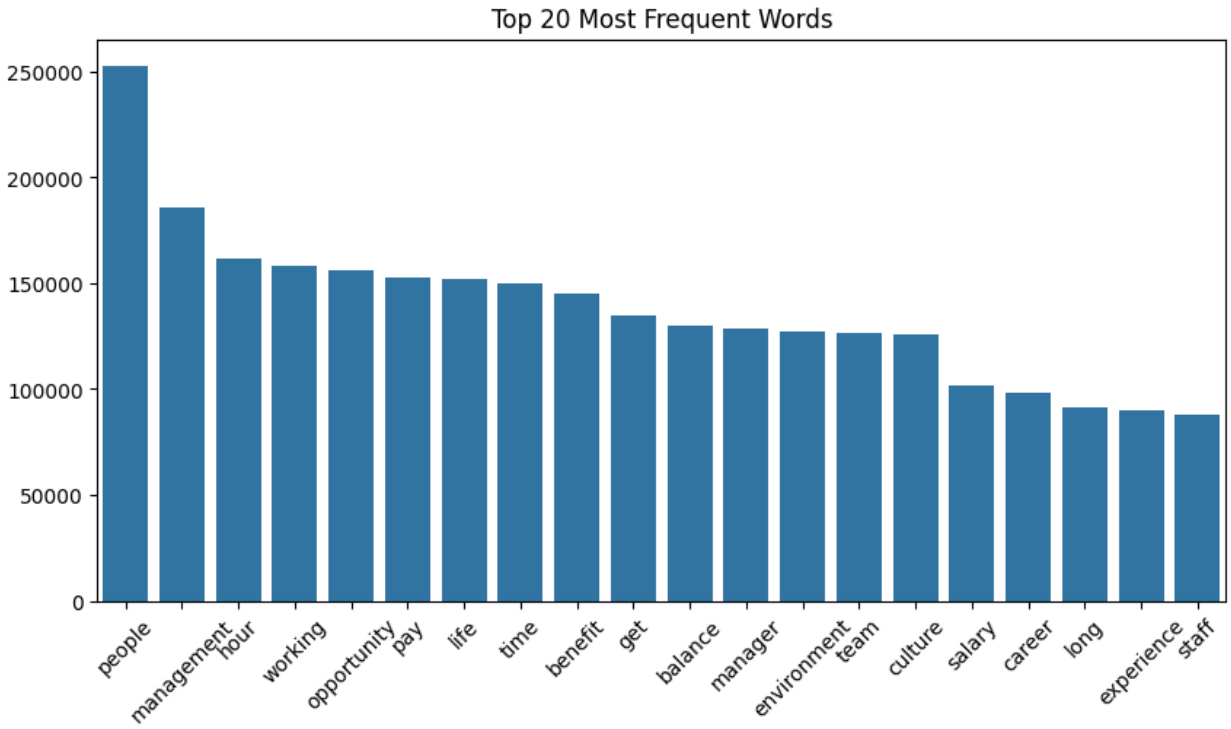
Median values for most aspect ratings are 4, suggesting that reviews are generally skewed towards positive experiences, although substantial numbers of negative ratings remain present.
Exploratory text analysis confirmed this distribution. The processed dataset contained 1.561.883 aspect-level sentences, with the following sentiment breakdown: 789.351 positive, 472.091 negative and 345.441 neutral.



Most reviews are relatively short, as shown in the word-count distribution, though a small number extends to several hundred words.



Frequent terms in the corpus include “people,” “management,” “opportunity,” “pay,” and “culture,” reflecting the central themes in employee experiences.



Data Processing, Annotation, and Normalization

The raw dataset had several text fields such as review headlines, pros, and cons, along with numerical aspect ratings. These fields contained missing values, inconsistent formats, or irrelevant tokens. To make the data more consistent, first, the three text columns ("headline," "pros," and "cons") were combined into a single text field. Missing values were replaced with empty strings instead of being removed, so that as many reviews as possible could be kept. The review dates were converted into a standardized datetime format with the pandas.to_datetime() function, and invalid or unclear dates were automatically set to "NaT" to avoid errors in later steps. This approach allowed the dataset to keep partial records that still had useful information, while avoiding the risk of bias that could come from deleting too many entries.

The text was then normalized by converting all characters to lowercase so that words like "Management" and "management" would be treated the same. User mentions and hashtags were removed with regular expressions, and all non-alphabetic characters, numbers, and punctuation were stripped out, as they usually do not add meaning for sentiment analysis. The cleaned text was then split into tokens using the NLTK word tokenizer. Also, common stopwords were removed with the NLTK English stopword list, and lemmatization was applied with the WordNet lemmatizer. Lemmatization was chosen instead of stemming because it reduces words to their proper base form while keeping their meaning intact.

After this cleaning step, each review was broken down into sentences using the NLTK sentence tokenizer. Aspect extraction was done by checking for keywords that matched six predefined categories: work–life balance, culture and values, diversity and inclusion, career opportunities, compensation and benefits, and senior management. These categories were selected because they match the numerical ratings in the dataset and represent key themes of employee experience. A sentence was linked to an aspect if it contained at least one of the keywords for that category. The choice of a keyword-based approach made the method

transparent and easy to interpret. Although more advanced approaches such as topic modeling could have been used, keyword matching ensured a direct and reliable connection between the text and the dataset's rating categories.

Sentiment labels at the aspect level were then derived from the corresponding numerical ratings. Reviews with ratings of 4 or 5 were considered positive, those with ratings of 1 or 2 were considered negative, and those with a rating of 3 were treated as neutral. This method made use of the dataset's built-in ratings as a form of supervision, removing the need for manual labeling. Even though this assumes that ratings reflect the sentiment expressed in the text, later consistency checks showed that the assumption was generally valid.

At the end of this process of preprocessing, cleaning, and annotation, the dataset contained more than 1.56 million aspect-level sentences. Each entry included a cleaned sentence, its assigned aspect category, and a sentiment label (positive, neutral, or negative). The dataset was then split into training, validation, and test sets using stratified sampling, so that the distribution of sentiment labels was preserved across splits.

Algorithms and NLP Architectures

For the supervised learning task, a transformer-based neural network was selected as the core model. Specifically, the architecture used in this project was DistilBERT, a lighter version of BERT that retains most of its performance while being more computationally efficient. The choice of DistilBERT was motivated by the very large size of the dataset. Based on its design, DistilBERT offers a good balance between speed, memory usage, and accuracy, making it appropriate for large-scale sentiment classification. The architecture uses a self-attention mechanism to capture contextual relationships between words, allowing the meaning of each word to be understood in relation to the entire sentence. This is particularly important in sentiment analysis, where the polarity of a statement often depends on context.

Before training, the text data had to be transformed into a numerical representation suitable for the model. This was done through the Hugging Face tokenizer, which applies WordPiece tokenization. Each sentence was split into subword tokens, mapped to unique integer IDs, and standardized through padding and truncation to a maximum sequence length of 80 tokens. This ensured that all sentences, regardless of length, were represented in a consistent input format. Attention masks were also created to distinguish meaningful tokens from padded positions, preventing the model from misinterpreting padding as real content.

As mentioned above, the dataset was divided into training, validation, and test sets to allow proper evaluation of model performance. A PyTorch DataLoader was used to handle mini-batch training, with batches of 32 sentences in the training phase and larger batches of 64 sentences during validation and testing to improve efficiency. The DataLoader also ensured that shuffling was applied to the training set at each epoch, which reduces the risk of the model overfitting to the order of the data.

The model was fine-tuned using AdamW, a variant of the Adam optimizer with weight decay regularization. A small learning rate of 2e-5 was chosen to allow gradual updates to the pretrained model weights without destroying the linguistic knowledge learned during pretraining. The cross-entropy loss function was used to optimize classification across the three sentiment categories. To further stabilize training, gradient clipping was applied to avoid exploding gradients, a common issue in transformer models. The training process was run for up to five epochs, although an early stopping mechanism was introduced. This mechanism monitored the F1-score on the validation set and stopped training if no improvement was observed for two consecutive epochs, preventing overfitting and saving computational resources.

Initially, an attempt was made to train the model using the high-level Hugging Face TrainingArguments and Trainer API, which provides built-in functionality for optimization and evaluation. However, due to execution constraints, this approach could not be successfully applied in the available environment. For this reason, a custom training loop was implemented with PyTorch, giving more direct control over optimization and validation, and ensuring that the training process could be monitored step by step.

Another practical limitation concerned the scale of the dataset and the available computational resources. Training the full dataset without a GPU would have required several days, which was not feasible on a local CPU-based setup. Even when using Google Colab, where a GPU was available, the execution time for the complete dataset exceeded the session time limits and resulted in training interruptions. Consequently, the decision was made to train the model on a sample of the dataset. This approach, while not ideal since it reduces representativeness and may affect generalization, allowed the full training and evaluation pipeline to be completed within the available time and resource constraints.

To measure the performance of the model, standard classification metrics were applied. These included precision, recall, F1-score, and accuracy, calculated in a weighted manner to account for class imbalance between positive, negative, and neutral samples. The F1-score was treated as the main evaluation metric, since it provides a balanced view of precision and recall. During training, these metrics were computed for both the training and validation sets after each epoch, enabling continuous monitoring of learning progress.

Once trained, the best-performing model checkpoint was saved together with the tokenizer, allowing the system to be reused for inference without retraining. A custom inference pipeline was developed to analyze new reviews. First, the review text is cleaned and split into sentences. Each sentence is then linked to aspects through keyword matching and finally passed through the fine-tuned transformer to predict sentiment labels. The output consists of predicted sentiment classes (positive, neutral, negative) with associated probabilities, making the results interpretable and actionable.

Finally, the trained model was used for business-oriented analysis by aggregating predictions at the aspect level and comparing them with the numerical ratings. This allowed the system not only to achieve high classification accuracy but also to provide insights into organizational strengths and weaknesses, such as whether negative feedback is concentrated on senior management or compensation.

In summary, the chosen transformer-based architecture combined with supervised fine-tuning allowed for accurate and scalable sentiment classification at the aspect level. The design choices—such as using DistilBERT, applying careful preprocessing, employing AdamW optimization with gradient clipping, monitoring performance with early stopping, and implementing a custom inference pipeline—ensured that the system was both efficient and robust, making it suitable for large-scale real-world applications.

Experiments-Setup, Configuration

The experimental setup was initially attempted on a personal laptop environment. However, the hardware proved unable to handle the computational demands of large-scale transformer training. The repeated attempts to train the model locally not only resulted in unacceptably long execution times but also caused the laptop to fail due to excessive load. This experience underlined the resource-intensive nature of modern NLP architectures and highlighted the need for more controlled and robust execution environments.

Following this, all experiments were carried out using Jupyter Notebook (version 1.1.1), launched through the PyCharm integrated console, which allowed a modular and interactive workflow. The implementation was based on Python 3.11, and all necessary libraries (PyTorch, Hugging Face Transformers, NLTK, Scikit-learn, Pandas, NumPy, Matplotlib, and Seaborn) were installed in their latest stable versions.

The available hardware was a CPU-only setup (Intel 12-core processor with 16 GB RAM), which imposed strict limitations on what could be achieved in practice. Running the model on the complete dataset required approximately 50 hours per epoch, a timescale that rendered iterative experimentation infeasible. Even with access to Google Colab GPUs, training times were still around 5 hours per epoch, often exceeding Colab's runtime restrictions and leading to frequent interruptions.

For these reasons, it was decided to conduct experiments on a sample of 20.000 reviews. With this reduction in dataset size, training time per epoch was reduced to about 2 hours on CPU, which made it possible to complete training and evaluation cycles within reasonable time limits. Although sampling reduces the representativeness of the dataset and introduces potential generalization constraints, this approach enabled a complete end-to-end experimentation pipeline to be executed. The decision thus reflects a trade-off between computational feasibility and methodological rigor.

The dataset was divided into training, validation, and test sets in an 80–10–10 split, using stratified sampling to preserve the relative distribution of positive, negative, and neutral sentiment labels across subsets. To ensure reproducibility and consistency across runs, a fixed random seed was applied, and model checkpoints were saved together with the tokenizer for reliable reuse.

The experiments also emphasized monitoring and evaluation. Performance metrics, including accuracy, precision, recall, and F1-score, were computed in a weighted manner to account for class imbalance. The F1-score was prioritized as the main evaluation metric, given its balance between precision and recall, and was also the metric used to trigger early stopping during training.

Results & Quantitative Analysis

During training, the model showed an improvement on the training set. More specifically, accuracy increased from about 59% in the first epoch to over 80% by the fifth epoch, while the F1-score grew from 0.51 to 0.80 in the same period. However, the validation set did not follow the same trend. Validation accuracy started at 59% in the first epoch and peaked at 58.5% in the second epoch, before gradually declining to around 55% in the final epoch. Similarly, the validation F1-score reached its highest value of 0.566 during the third epoch and then decreased slightly in later epochs. This pattern indicates that the model was learning well on the training sample but began to overfit, losing its ability to generalize to unseen data after the third epoch.

Epoch	Train Accuracy	Train F1	Val Accuracy	Val F1
1	0.588	0.510	0.590	0.502
2	0.626	0.553	0.585	0.517
3	**0.670**	**0.634**	**0.560**	**0.566**
4	0.731	0.719	0.558	0.551
5	0.807	0.804	0.554	0.559

A closer look at the results shows that the model handled positive reviews better than neutral or negative ones. In the early stages of training, there were even warnings that some labels had no predicted samples at all, meaning the model completely ignored certain sentiment classes. As training progressed, these issues were reduced, but neutral reviews continued to be the most difficult for the model to classify. They were often predicted as positive, suggesting that the model learned to recognize strong positive and negative

expressions but struggled with more subtle or balanced opinions. Negative reviews were sometimes missed as well, showing lower recall, although they were not as problematic as the neutral ones.

Although the training results were strong, with the model fitting the data very closely, the validation performance reached a plateau around the third epoch and did not improve afterwards. The highest F1-score shows that the model was able to capture useful patterns in the text but could not achieve high accuracy under the given constraints. One of the main reasons for this limitation was the use of a relatively small sample of the dataset. The decision to train on 20.000 reviews was necessary because training on the full dataset would have taken several days on a CPU, as mentioned above. While the sample made it possible to complete the full training, validation, and testing process, it also reduced the representativeness of the data and limited the generalization power of the model.

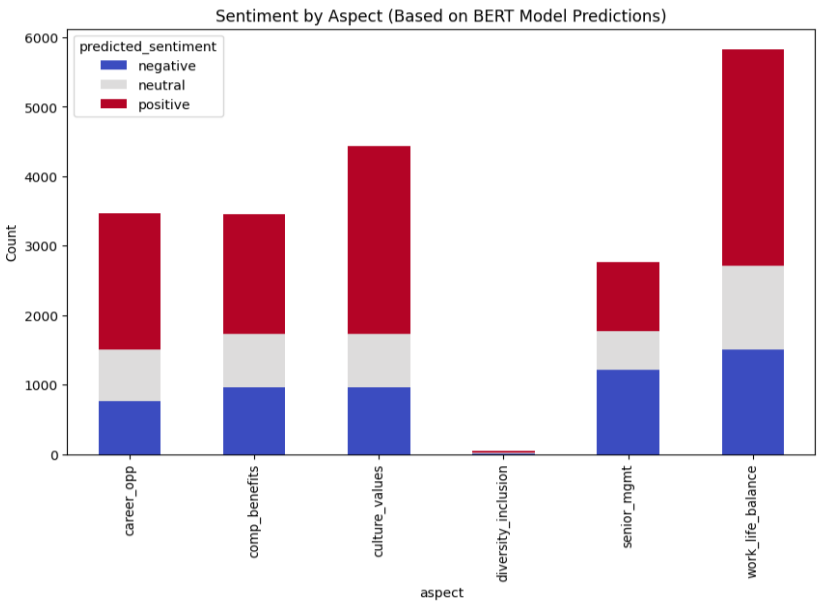
The final step of the experimental pipeline involved evaluating the trained model on both the validation and test datasets. For this purpose, a dedicated evaluation function was implemented. The function evaluate_model takes the model, a data loader, and the device as inputs. It sets the model to evaluation mode and ensures that gradient computation is disabled, which reduces memory usage and speeds up inference. For each batch in the data loader, the function retrieves the inputs and labels, moves them to the appropriate device, and then obtains the model predictions. These predictions are collected along with the true labels, and at the end they are passed to the compute_metrics function in order to calculate performance scores.

Using this function, the model was first evaluated on the validation set and then on the test set. The validation results were as follows: precision = 0.565, recall = 0.554, F1-score = 0.559, and accuracy = 0.554. The test results were very similar, with precision = 0.556, recall = 0.546, F1-score = 0.550, and accuracy = 0.546. The closeness of validation and test scores shows that the model did not overfit to the validation data and that its generalization ability is consistent across unseen examples.

To evaluate the trained model on unseen data, a function called predict_sentiments was created. This function takes each sentence in the dataset, tokenizes it, and then passes it through the fine-tuned BERT model. The model outputs a set of probabilities for each sentiment class, and the class with the highest probability is selected as the prediction. Each prediction is then mapped back to its label (positive, negative, or neutral) using the label2id dictionary. The predicted sentiments are added as a new column in the original DataFrame, so that for every aspect we now have both the aspect name and its predicted sentiment.

This step allowed to apply the trained model on the entire dataset, moving beyond the validation and test sets. In this way, we could generate large-scale predictions and later use them to analyze the distribution of sentiments across different aspects.

Forward to the analysis, the predicted sentiments from the fine-tuned BERT model were grouped by aspect. The results were then visualized in a stacked bar chart, where each bar shows the distribution of positive, negative, and neutral feedback for a given aspect.



To make the results more actionable, the proportions of each sentiment were calculated for every aspect. Based on these percentages, a simple set of business recommendations was generated. If an aspect had more than 40% negative feedback, it was flagged as requiring urgent attention. If an aspect had more than 60% positive feedback, it was highlighted as a strong point worth continuing to invest in. All other cases were considered mixed feedback and recommended for monitoring.

From the results, career opportunities and compensation & benefits both show mixed feedback, with a balance of positive and negative reviews. This suggests that while some employees value these areas, others are dissatisfied, so improvements may be needed. Culture and values and diversity and inclusion appear as strong points, with over 60% of the feedback being positive, indicating that employees generally perceive these aspects favorably. On the other hand, senior management stands out with a high percentage of negative feedback, which signals an area where immediate action is necessary. Finally, work-life balance also shows mixed feedback, again suggesting that this aspect should be closely monitored to prevent it from becoming a source of dissatisfaction.

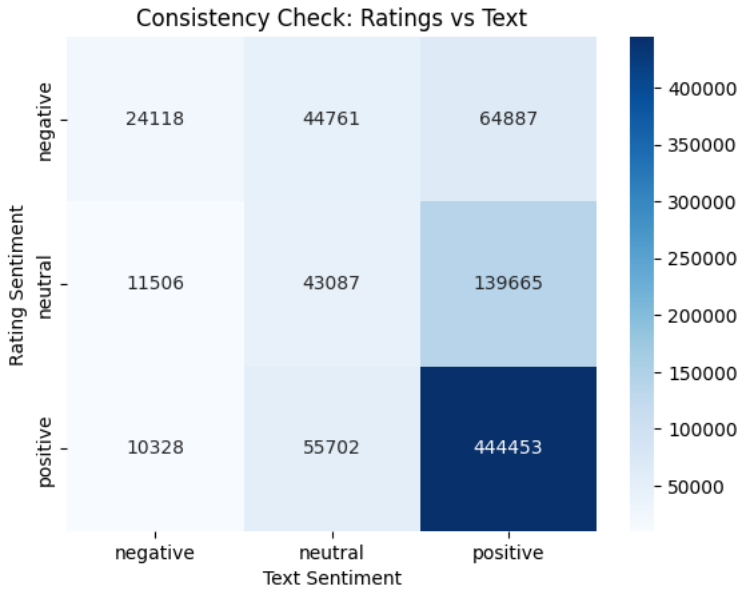
Aspect	**Positive (%)**	**Negative (%)**	**Neutral (%)**	**Recommendation**
Career Opportunities	56.5	21.9	21.6	Mixed feedback → Monitor
Compensation & Benefits	49.8	27.9	22.3	Mixed feedback → Monitor
Culture & Values	60.8	20.1	19.1	Strong point → Continue investing
Diversity & Inclusion	61.7	19.2	19.1	Strong point → Continue investing
Senior Management	28.0	44.0	28.0	High negatives → Immediate action needed
Work-Life Balance	53.5	25.8	20.7	Mixed feedback → Monitor

To evaluate more the reliability of the dataset, a consistency check was performed between the numerical ratings given by users and the sentiment expressed in the review text. For this purpose, the TextBlob library was used to assign a sentiment label to every review text, based on its polarity score. At the same time, the numerical ratings were converted into sentiment labels: ratings of 4 and 5 were considered positive, ratings of 1 and 2 negative, and a rating of 3 neutral.

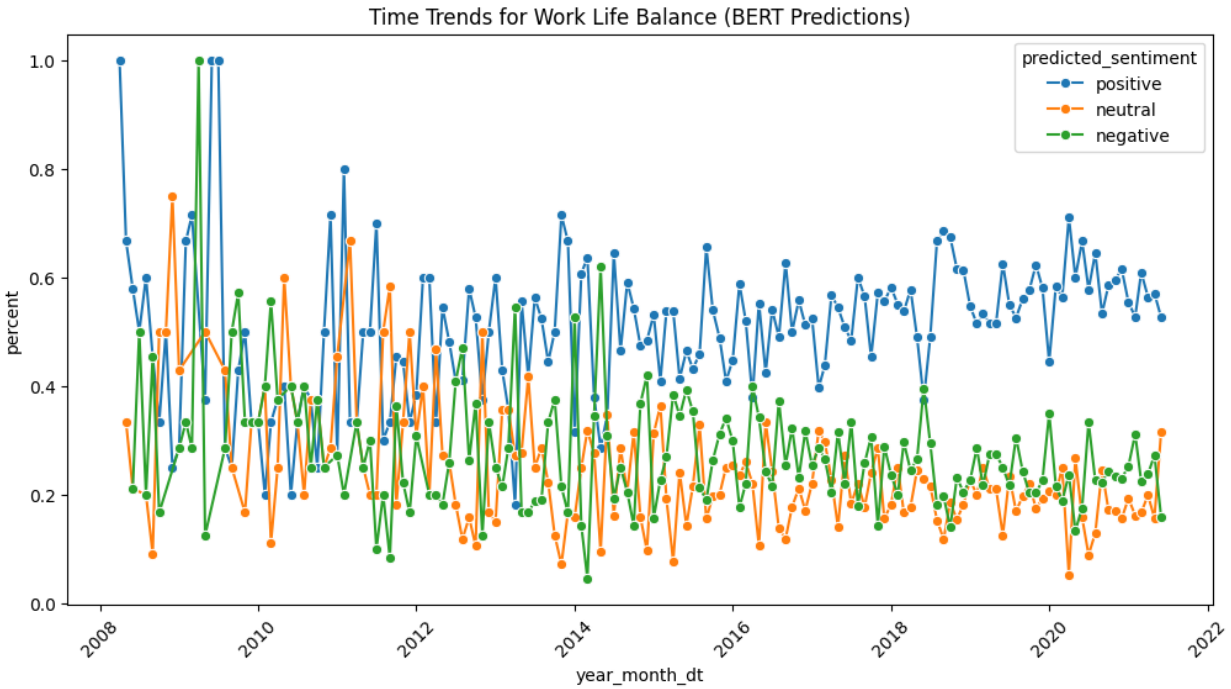
After this transformation, the two sets of labels were compared and the percentage of cases where the text sentiment agreed with the rating sentiment was calculated. Although the consistency was reasonably high, several mismatches were identified. These mismatches show situations where the written text does not align with the numerical score.

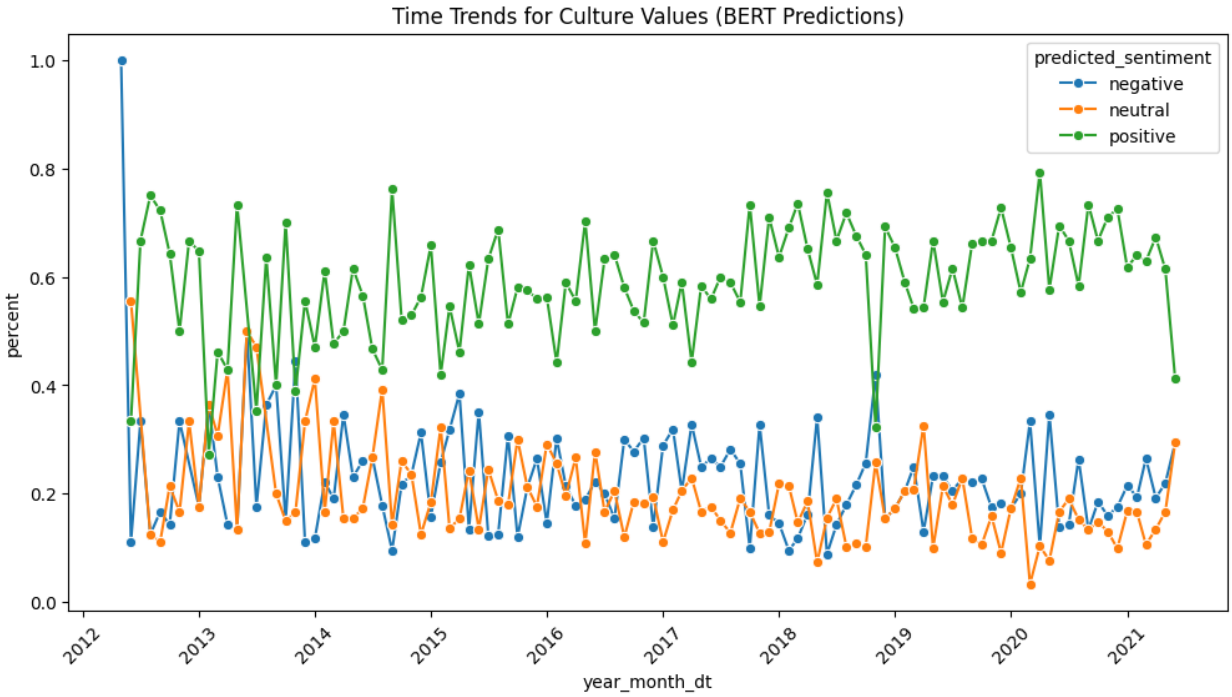
For example, some reviews contained negative comments about salary or management but were still given a relatively high overall rating. On the other hand, other reviews included positive statements about colleagues or workplace culture, but the user assigned a low numerical rating. These inconsistencies suggest that user ratings may sometimes reflect a general impression or external factors not directly captured in the written text.

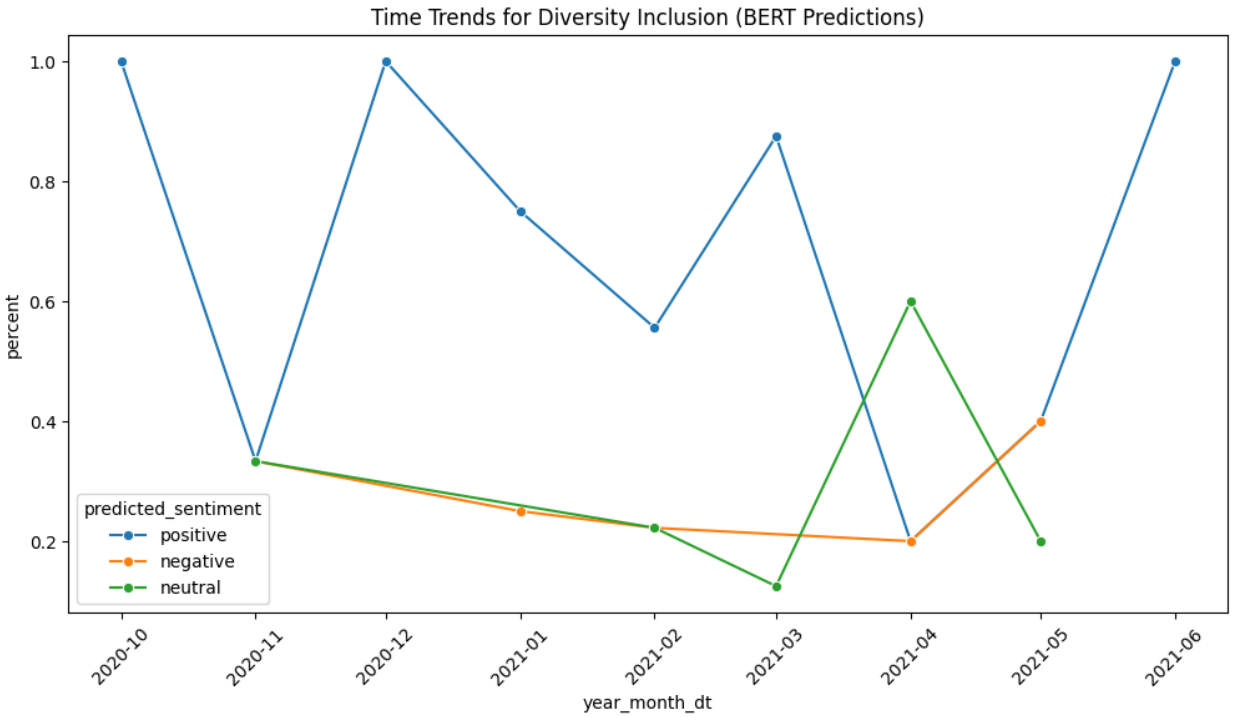
A confusion matrix was also created to visualize the agreement between the two labeling approaches. This heatmap shows how often text-based sentiment matches or diverges from rating-based sentiment across positive, negative, and neutral categories.

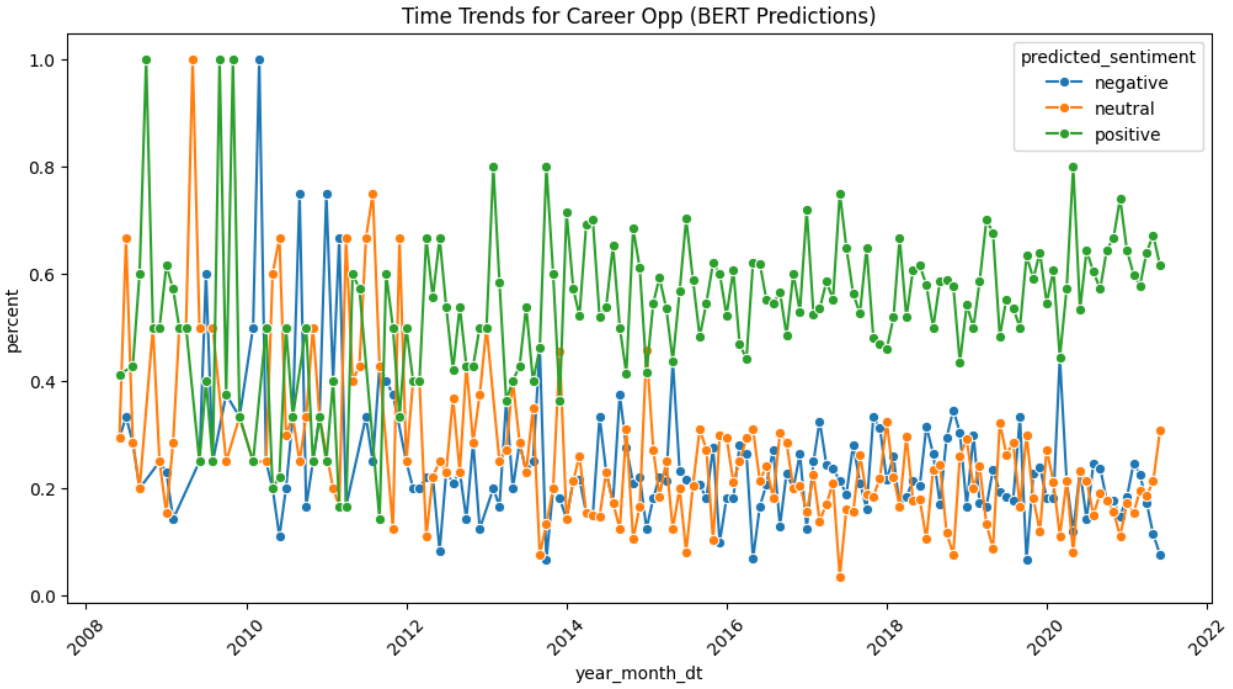


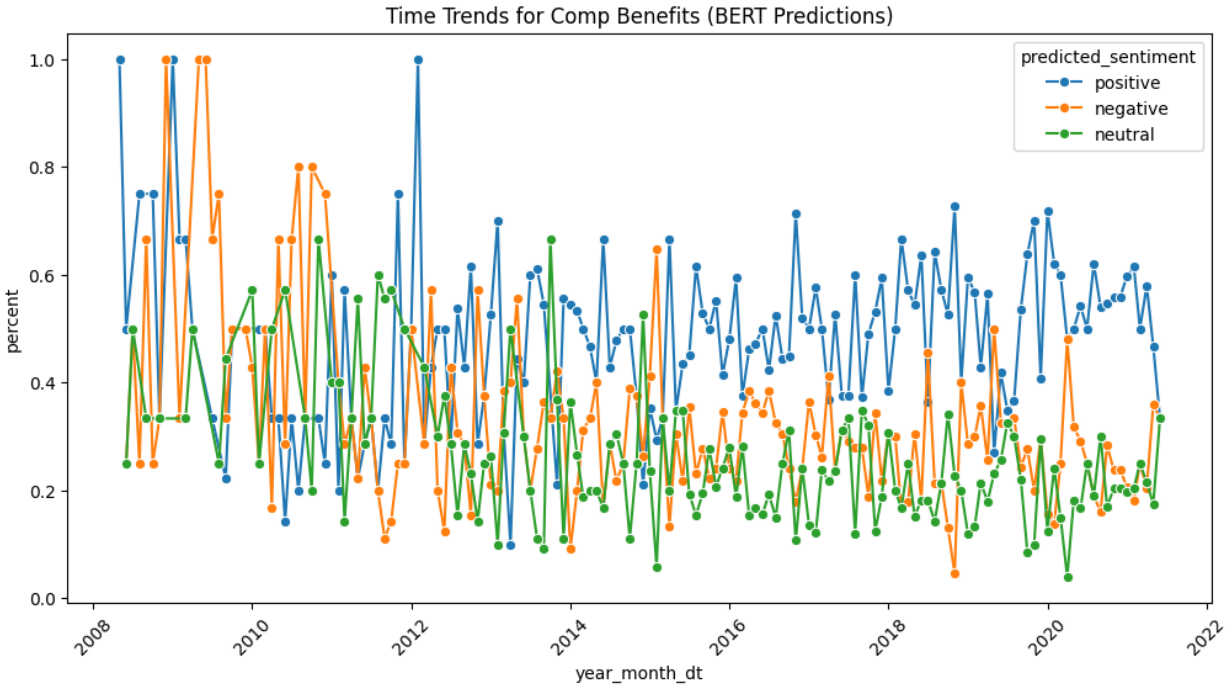
Furthermore in the analysis, in order to better understand how employee opinions evolved over time, the predicted sentiments were analyzed in relation to the review dates. For this purpose, each review was assigned to a specific month and grouped by aspect and sentiment class. The counts were then normalized into percentages in order to make trends comparable across different aspects and time periods.
The results were visualized using line plots, where the x-axis represents time (year and month) and the y-axis shows the relative percentage of each sentiment.

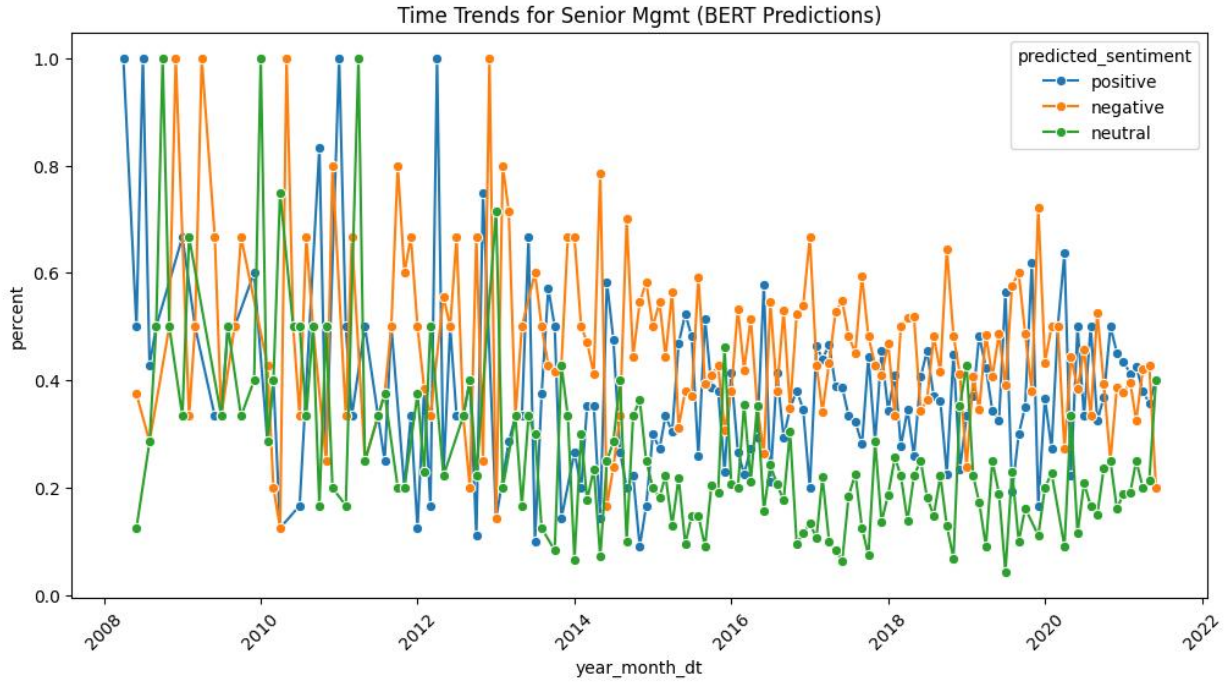





Time Trends for Diversity Inclusion (BERT Predictions)
percent
year_month_dt
predicted_sentiment
positive
negative
neutral
1.0
0.8
0.6
0.4
0.2
2020-10
2020-11
2020-12
2021-01
2021-02
2021-03
2021-04
2021-05
2021-06


Time Trends for Career Opp (BERT Predictions)
percent
year_month_dt
predicted_sentiment
negative
neutral
positive
1.0
0.8
0.6
0.4
0.2
0.0
2008
2010
2012
2014
2016
2018
2020
2022





The final analysis was a risk assessment performed at the company level, based on the sentiment predictions produced by the BERT model. To measure this, a simple risk score was calculated by dividing the number of negative reviews by the number of positive ones. A higher score indicates that negative opinions are more dominant compared to positive feedback, and therefore the company may face higher reputational or organizational risks. The table below presents the Top 10 firms with the highest risk scores.

Firm	Negative	Neutral	Positive	Risk Score
The-Range	9	0	0	9.00
Mitie	17	1	2	5.67
ISS-Facility-Services	19	7	4	3.80
Link-Group	11	1	2	3.67
Creative-Support	6	0	1	3.00
HM-Prison-Service	3	2	0	3.00
Grange-Hotels	3	0	0	3.00
Rapport-London	3	0	0	3.00
Four-Seasons-Health-Care	8	0	2	2.67
Debenhams	37	11	15	2.31

The results showed that The-Range had the highest risk score, with only negative reviews and no positive ones at all. Other companies such as Mitie and ISS-Facility-Services also scored high, reflecting a strong imbalance between negative and positive feedback. On the other hand, Debenhams had the largest number of negative reviews overall, but since it also had more positive ones than most other companies in the list, its risk score was comparatively lower.

Qualitative & Error Analysis

In order to test the practical applicability of the fine-tuned DistilBERT model, an inference function was implemented to analyze new reviews. The function first cleans and splits the input text into sentences, then matches each sentence against a predefined set of aspects using keyword detection. If an aspect is identified, the sentence is tokenized and passed through the trained model, which predicts the sentiment label along with a probability score. The results are aggregated into a structured table, linking each sentence to the corresponding aspect and sentiment.

When applied to the example review “The management was poor but salary and benefits were great. The schedule is flexible though.”, the system correctly identified three relevant aspects: work-life balance, compensation and benefits, and senior management. However, all of them were classified as negative with a relatively low probability (0.39). This outcome highlights both strengths and weaknesses of the current approach. On the positive side, the aspect extraction step worked as expected, successfully detecting the presence of multiple aspects within a single review. On the other hand, the sentiment classification struggled with cases of mixed opinions: while the review expresses dissatisfaction with management, it also clearly mentions positive views on salary, benefits, and schedule. The model was not able to capture this contrast and instead assigned a uniform negative label to all aspects.

This limitation can be attributed to several factors. First, the training process was conducted on a reduced sample, which limited the model’s ability to generalize to more complex patterns. Second, many sentences in the dataset contain overlapping mentions of multiple aspects, which makes aspect-specific sentiment detection more challenging. Finally, the relatively low probability values indicate a lack of strong confidence in the predictions, suggesting that the model is uncertain in such scenarios.

Overall, the inference experiment demonstrates that the system is functional and capable of linking text to aspects, but it also exposes important areas for improvement. Future work could explore more granular aspect segmentation (e.g., at the clause level rather than the sentence level), the use of larger and more

representative training sets, or the integration of aspect-specific attention mechanisms to better separate positive and negative cues within the same sentence.

Sentence	Aspect
0 management poor salary benefit great schedule ...	work_life_balance
1 management poor salary benefit great schedule ...	comp_benefits
2 management poor salary benefit great schedule ...	senior_mgmt

	pred_label	pred_prob
0	negative	0,392567
1	negative	0,392567
2	negative	0,392567

Discussion, Comments/ Notes and Further Work

The results of the experiments demonstrate both the potential and the limitations of applying transformer-based models to large-scale aspect-based sentiment analysis. The analysis pipeline successfully transformed unstructured employee reviews into structured insights, identifying key aspects such as work–life balance, culture and values, compensation, and senior management, and associating them with sentiment categories. This allowed for the generation of business-oriented interpretations, including recommendations for organizational improvements and risk assessment at the company level.

However, several important limitations must be acknowledged. First, the computational requirements of transformer models presented a major challenge. Although DistilBERT was chosen for its efficiency compared to larger architectures, training on the complete dataset proved infeasible given the available resources. As a result, the model was trained on a sample of 20.000 reviews. While this enabled a complete end-to-end workflow, it inevitably restricted the model's ability to generalize and limited the representativeness of the findings. The discrepancy between strong training performance and weaker validation results further reflects the consequences of this constraint.

Second, the dataset itself presents challenges. The reviews extend up to 2022, meaning that more recent employee opinions are not captured. Workplace culture, management practices, and employee priorities evolve rapidly, particularly in the aftermath of global events such as the COVID-19 pandemic and shifts toward hybrid or remote work. Without updated data, the analysis cannot fully reflect the current state of employee sentiment. In addition, inconsistencies between textual content and numerical ratings indicate that user ratings may not always align with written feedback, suggesting a potential source of noise in the supervision signal.

Third, the use of keyword-based aspect extraction, while transparent and interpretable, is also limited. Keywords may fail to capture nuanced references to aspects when employees use synonyms or indirect phrasing, and some sentences may be incorrectly linked to aspects due to overlapping vocabulary. Although this method aligns neatly with the dataset's numerical categories, it reduces the flexibility of the system to discover emergent themes that may not be explicitly included in the predefined aspects.

Looking ahead, several directions for future work are recommended. Most importantly, the model should be trained on the full dataset rather than a reduced sample. In addition, an updated dataset should be

collected to include reviews from 2023 onward, enabling the analysis to reflect the latest trends and organizational practices. Expanding the temporal coverage would also make it possible to conduct longitudinal analyses, tracking how employee sentiment evolves year by year.

Another promising extension would be to conduct the analysis at the company level. By aggregating predictions for individual firms, it would be possible to compare companies against one another, identify leaders and laggards across specific aspects, and compute risk scores with greater precision. This would add substantial practical value for both job seekers and corporate stakeholders.

Further methodological improvements could also be considered. More advanced aspect extraction methods, such as embedding-based clustering or supervised multi-label classification, could replace simple keyword matching.