Exploring Synthetic Data Generation Techniques for Employment Type Classification in Job Advertisements

Anonymous ACL submission

Abstract

The classification of employment types in online job advertisements (OJAs) is crucial for labor market analysis and recruitment. This study addresses the limitations of manual data annotation by leveraging synthetic data generation (SDG) techniques using large language models (LLMs). We evaluate four SDG methods—plain prompting, sampling, precise attributes, and adjective attributes—to generate synthetic job ads and assess their impact on classification model performance. Our analysis focuses on the balance between dataset size, data diversity and label-fit, and we explore the use of Natural Language Inference (NLI) filtering to enhance data quality. Results show that models trained on synthetic data can effectively classify real-world job ads, achieving competitive performance. However, we observed significant volatility in outcomes, which we could not fully explain. By making our code and data publicly available, we provide the research community with opportunities to further investigate SDG techniques. By publishing our best models, we offer researchers tools capable of achieving up to 96% F1 on a real-world dataset for classifying German OJAs by employment type.

1 Introduction

Classifying employment types in online job advertisements (OJAs) is crucial for labor market analysis and recruitment. Krüger (2023) categorized OJAs into four types but faced data scarcity and imbalance issues. This study addresses these challenges using synthetic data generation (SDG) techniques with large language models (LLMs). This study evaluates SDG methods for generating synthetic job ads and their impact on employment type classification models, focusing on balancing data diversity and label-fit. We also explore using Natural Language Inference (NLI) filtering to enhance data quality.

The main contributions of this paper are:

1. We compare four prompting methods for SDG: plain prompting, sampling, precise attributes, and adjective attributes.
2. We investigate the effects of these methods on data diversity and label-fit, reflecting on measurement methods and identifying research needs.
3. We assess the effectiveness of NLI-based filtering in improving synthetic data quality and model performance.
4. We benchmark models trained on synthetic data against those trained on real-world data, showing SDG's potential in employment type classification.
5. Our results show seemingly arbitrary performance volatility. We offer our code, data, and models publicly for further investigation and improvement.
6. We release distilBERT models with up to 96% F1 score for employment type classification, providing robust tools for researchers working with German OJAs.

2 Motivation and Background

We build on Krüger (2023), which classified German OJAs into the employment type categories *Apprenticeships*, *Other Minor Positions*, *Leading Positions*, and *Regular Workers*. They faced challenges due to data scarcity and label imbalance, with some categories appearing fewer than ten times in 15,000 labeled OJAs. This necessitated merging labels and highlighted the resource-intensive nature of manual labeling. Therefore, a more feasible and dynamic approach is required.

Recently, SDG in Natural Language Processing is increasingly used to address data scarcity issues (Delmas et al., 2024; Li et al., 2023; Schmidhuber and Kruschwitz, 2024; Josifoski et al., 2023; Veselovsky et al., 2023) due to the rise of generative LLMs. The idea is to prompt LLMs to generate text conditioned on various aspects such as the label space, text type, or genre. This data can then be used to train downstream Language Models. Contemporary research has shown that this technique is also promising to OJA research (Clavié and Soulié, 2023; Magron et al., 2024; Borchers et al., 2022). Prompting LLMs to generate job ads appears to consistently output relatively realistic job ads. This is presumably due to the large amount of OJAs available publicly on the internet, which results in these data being included in the often publicly scraped training data of LLMs. Furthermore, using LLMs to generate synthetic data to train a downstream task (like text classification) specific model has proven to yield better results than using the LLM as a zero-shot predictor for the specific task directly (Schick and Schütze, 2021; Meng et al., 2022; Ye et al., 2022; Josifoski et al., 2023). Also, downstream models can be a lot cheaper computationally (Ye et al., 2022; Schick and Schütze, 2021), which has major practical and ethical (Bannour et al., 2021; Strubell et al., 2019) advantages.

As a recent technique, SDG is still under research. One key advantage is the potential to generate practically unlimited training data. While generating an infinite amount of data is impractical and unnecessary for simple tasks, the ability to create large volumes of data can significantly enhance model performance[1]. However, one particular aspect that has been found to be relevant in this regard is text diversity. Since data generation with LLMs, even with sampling techniques for randomization, is a statistical process, repeatedly using the same prompt will eventually produce outputs with certain biases, resulting in redundancy. When the dataset then becomes too similar, the ability of the downstream model to generalize will be affected. In their research, Ye et al. (2022) show that with increasing the size of the synthetic dataset, the performance of the downstream model increases, up to a certain "threshold" point at which the performance plateaus (or even drops). We argue that the reason behind this is that the data has become too similar, causing the model to overfit. We hypothesize that the more diverse the generated data is, the higher this threshold can be. Therefore, finding ways to diversify synthetically generated datasets has been brought up by researchers as a promising approach to improve SDG (Yu et al., 2024; Ye et al., 2022; Schick and Schütze, 2021; Clavié and Soulié, 2023). One pitfall in this regard has been brought up by (Ye et al., 2022), who mention that a more diverse dataset will only be beneficial as long the quality or correctness, which in this paper we will call label-fit (see Section 3.2 for a formal definition), is not impaired. Generating random words would create a more diverse dataset, but in order to train a functioning downstream model, the data will need pertain their label-fit. In their study, Ye et al. (2022) find there to be a balance between diversity and label-fit. In our study we want to test different prompting methods to generate synthetic OJAs for employment type classification and investigate the interaction between dataset size, diversity, label-fit and the performance of the downstream model. We also investigate how applying a NLI filter influences the performance of SDG.

3 Related Work

3.1 Diversifying SDG

We present recent approaches to diversifying SDG similar to our research. Ye et al. (2022) find that more sampling leads to higher diversity but less stable label-fit, using Self-BLEU to measure diversity and human evaluation for label-fit. Yu et al. (2024) use what they call attributes to diversify prompts, meaning that they introduce a template to the prompt where certain attributes that the desired output should have can be specified. For their work on topic classification of newspaper articles, these attributes are the *subtopic*, *length*, *style*, and *location* of the articles generated by the LLMs. They measure diversity by *Vocabulary Size*, *Average Pairwise Sample Similarity* and *Inter-Sample N-gram Frequency*. They do not directly measure label-fit, but perform manual analysis of biases in their data. They find that their technique creates somewhat more diverse data compared to a simple prompting technique, but much less diverse than the public gold standard datasets. They also conclude that designing prompts with diverse attributes contributed positively to the performance of the downstream model.

[1] The absolute amount of data required for good performance depends on the problem's complexity.

For skills matching, Magron et al. (2024) generate synthetic training sentences containing skills, whereby they seek to diversify their dataset by varying the lengths of skill combinations for each sentence. They also prompt the model to vary the openings of the descriptions and avoid certain phrases. They claim to measure diversity and quality of their generated data based on *Perplexity*, *Skill-Sentence Similarity* and *Explicitness*, but do not mention which metric specifically measures diversity. They do, however, conclude that higher diversity of training data leads to a better skill matching performance.

3.2 Measuring Diversity and Label-fit

In Section 3.1 we have already discussed how other work in SDG has quantified **text diversity**. They all use different metrics. We argue that this is a consequence of the fact that quantifying text diversity is a non trivial task with various conceptual and operational challenges. Beyond SDG, text diversity measurement is discussed in broader research areas like Natural Language Generation (NLG) and Machine Translation (MT). We summarize key aspects from this literature.

Tevet and Berant (2021) review commonly used diversity metrics and cluster them into the four categories *Perplexity, N-gram-based metrics, Embedding-based metrics* and *Human evaluation*. They also make an important point that to our knowledge has not been considered in works on diversity in SDG: there are be different *types* of diversity (Tevet and Berant, 2021). They use the division of *form* and *content* diversity, but acknowledge that these can be divided further into, for example in the case of form diversity, *syntactic* and *lexical* diversity (Tevet and Berant, 2021). We argue that designing research on diversity in SDG should first identify the specific type of diversity being studied and then select appropriate quantifying metrics or at least reflect on it.

With regards to the metrics and types of diversity introduced above, it can be said that **Perplexity**, which is commonly used (Tevet and Berant, 2021; Hashimoto et al., 2019), measures the LLM rather than the dataset, making it unsuitable for evaluating texts obtained from different sampling and prompting strategies in SDG. **N-gram-based metrics** like Self-BLEU (Zhu et al., 2018) measure form diversity well but poorly assess content diversity (Tevet and Berant, 2021). Lexical diversity also counts as an N-gram metric. **Embedding-based metrics** evaluate diversity by embedding sentences in a latent space, performing similar in form diversity but better in content diversity (Tevet and Berant, 2021). **Human evaluation** captures diversity most effectively (Tevet and Berant, 2021) but is resource-intensive.

We argue that in SDG, focusing on form diversity is reasonable as content diversity is often limited by factors like text type or class set. OJAs, for example, have predetermined content. Research on quantifying text diversity is ongoing, with no single perfect metric. Therefore, we use a combination of methods for our analysis, listed in Section 5.4.

Since previous literature uses various to describe **label-fit** and similar concepts (quality[2], correctness, density), we first create a definition of it. Consider $\mathcal{L} = \{l_1, l_2, \ldots, l_i\}$, a set of labels. For a text t to be conditionally generated for a specific label (e.g., l_1) and used in training a downstream classification model, it must possess distinguishing features characteristic of l_1 and not simultaneously associated with other labels in $\mathcal{L}$. To the best of our knowledge, there exists very limited literature on how to quantify label-fit. Ye et al. (2022) measure it in two different ways. Firstly, they train a classification model based on a standard training dataset, which might be suitable for their purpose, but cannot be applied in a real-world scenario because such a dataset is not available in contexts of data sparsity. Secondly, they perform human evaluation, which is an option but is also resource-intensive. To the best of our knowledge, the only work to automatically quantify label-fit agnostic to existing training data is by Lai et al. (2020), who call it density. They measure the number of data points (texts) that fall within a unit volume in the embedding space, accounting for high-dimensional space through a dimension-normalized volume calculation. However, the authors did not provide code or data to replicate their findings or method. Hence, for this work, we opted to perform a human evaluation to quantify label-fit.

3.3 NLI Filtering

Improving label-fit involves filtering out data with poor label alignment. Bartolo et al. (2021) showed that various filtering methods improve question answering models, though not directly applicable here. Chen and Liu (2022) build on the common

[2]Note, that quality as measured frequently in MT (for example Alihosseini et al. (2019)) is different from label-fit, because it is measured w.r.t reference data.

idea to use NLI (Bowman et al., 2015) as a Zero-Shot method by reformulating NLP problems as premise and hypothesis pairs (Wei et al., 2021). They used synthetic text as the premise and label space as the hypothesis, showing that NLI filtering generally improves results (Chen and Liu, 2022). We adopted a simple NLI filtering approach for this work.

4 Research Questions

This study addresses aspects of synthetic data generation (SDG) for employment type classification in OJAs based on the current state of research as outlined above. We focus on the following research questions:

1. **Effectiveness of Synthetic Data**: Can models trained on synthetic job ads classify real-world ads effectively?
2. **Optimal Data Generation Strategies**: What strategies generate training data with optimal diversity and label-fit?
3. **NLI Filter:** Does integrating an NLI filter significantly improve model performance?
4. **Data Diversity and Label-Fit**: Is there a correlation between data diversity, label-fit, and model performance? Does enhancing diversity while preserving label-fit expand the plateuing threshold?
5. **Diversity Metrics**: How do different diversity metrics impact experimental outcomes?

5 Methodology

For our experiments, we generate job ad data conditioned to a label space from the task of employment type classification and use this data to fine-tune a downstream text classification model, whose performance we test on manually curated test sets. More specifically, we seek to test different methods to generate synthetic data with respect to the diversity and label-fit of the dataset as well as the performance of the downstream model. For each method, we also generate datasets of different sizes to investigate the plateauing effect of synthetic data generation (SDG). Additionally, for each dataset, we employ a filtering step and calculate each metric with and without the filter.

More formally, if we let $\mathcal{D} = \{(X, Y)\}$ be a dataset containing text and label pairs, we can define that:

- $\mathcal{D}^{test} = \{(X, Y)\}$ is a manually curated test set.
- $\mathcal{M}$ is a set of methods for generating synthetic data conditioned to a label space.
- $\mathcal{N}$ is a set of size parameters, indicating how much synthetic data is generated.

For each combination $(m_i, n_j) \in \mathcal{M} \times \mathcal{N}$, we generate synthetic datasets $\mathcal{D}^{g}_{m_i,n_j}$ and filtered datasets $\mathcal{D}^{gf}_{m_i,n_j}$. We fine-tune a text classification model $\mathcal{C}_{m_i,n_j}$ on each dataset and denote models trained on unfiltered and filtered datasets as $\mathcal{C}^{\text{unf}}_{m_i,n_j}$ and $\mathcal{C}^{\text{f}}_{m_i,n_j}$, respectively. Then, for each $\mathcal{C}^{\text{unf}}_{m_i,n_j}$ and $\mathcal{C}^{\text{f}}_{m_i,n_j}$ we calculate a Performance ($\mathcal{P}$) of the model $\mathcal{C}_{m_i,n_j}$ on $\mathcal{D}^{test}$, e.g., F1-score. For each $\mathcal{D}^{gf}_{m_i,n_j}$ we also calculate a Diversity Score ($\mathcal{DS}$) and manually evaluate the Label-Fit ($\mathcal{LF}$). We chose to assess these metrics only on the filtered data, because they are very resource intensive to measure, requiring significant computational power ($\mathcal{DS}$) and human effort ($\mathcal{LF}$).
We will detail the experimental pipeline in the following sections, specifying the metrics used to measure $\mathcal{P}$, $\mathcal{DS}$, and $\mathcal{LF}$.

5.1 Parameters

The experiment pipeline operates with two primary parameters:

1. **Size**: This parameter dictates the total number of job ads in the training set, divided equally across all classes (rounded down for parity). For instance, a size setting of 500 results in 55 ads per class. The size range [500, 1000, 2500, 5000, 7500] was selected based on prior studies (Krüger, 2023; Ye et al., 2022).
2. **Method**: This refers to the technique used for creating prompts fed into the LLM. Four distinct methods are employed:
   (a) **Plain**: The baseline method, where the prompt straightforwardly requests a job ad for a specific class, e.g., "A job ad for an internship."
   (b) **Sampling**: Similar to Plain, but with a higher 'top k' sampling parameter (Plain = 5; Sampling = 50), encouraging dataset diversity at the potential cost of quality. This method is based on the findings in Ye et al. (2022).

(c) **Precise Attributes (Prec)**: This method diversifies prompts with detailed instructions about the ad, varying by class. These include ad length, language style, content elements, and industry sector (or other relevant class-specific details such as the formalized name of the apprenticeship), adhering to German WZ08 taxonomy standards (Kla, 2008). This method is based on the ideas presented in Yu et al. (2024). Rather than their approach of using a LLM to derive relevant attributes, we manually reflected on possibly relevant attributes for our text type. The template and all options can be found in appendix B.

(d) **Adjective Attributes (Adj)**: Here, prompts are enhanced with 2 to 5 adjectives, randomly selected from a list of 30, describing possible language styles of OJAs. This method also is based on the ideas presented in Yu et al. (2024), but is simpler. Instead of having to manually construct a set of attributes (with or without the help of LLMs), we simply had to come up with a set of adjectives that can describe the style of text type, which is quicker and requires less effort.

Each method was conceived to explore different aspects of job ad generation, with the ultimate goal of enhancing the diversity and quality of the synthetic dataset for effective model training.

5.2 Dataset Generation

The dataset generation aligns with the parameters delineated in 5.1. We utilized the Falcon-40b model[3], because at the time of conducting the experiments it was the state-of-the-art open source[4] option that included German text in its training data. The only alternative, a larger 180b model, was not feasible due to GPU constraints. For efficient inference, we utilized the VLLM library[5], incorporating techniques like continuous batching and paged attention for enhanced performance ((Kwon et al., 2023)).

[3] https://huggingface.co/tiiuae/falcon-40b

[4] The term *open source* can be debated, we refer to (Liesenfeld and Dingemanse, 2024) for an in depth discussion

[5] ://github.com/vllm-project/vllm

5.3 Filtering

A NLI model was employed for dataset filtration, assessing each job ad against the hypothesis "class label name wanted" using the multilingual mDeBERTa model's[6] zero-shot classification pipeline (Yang et al., 2020). Ads not ranking their actual class within the top three predictions were excluded. Both filtered and unfiltered datasets were used for training downstream models to evaluate the filtering's impact on performance. In a preliminary test phase, we found that this approach seemed to yield decent results for all label categories except *regular full-time positon* from which the model filtered out ads disproportionately. This category is special in the sense that it is the norm and therefore less specific and salient than the other label categories, which may be the reason why the model performed worse for this class. Therefore, for experiments, we skipped the filtering for ads from the *regular full-time positon* category.

5.4 Data Analysis

In this step, we quantify the diversity and label-fit of all datasets $\mathcal{D}^{gf}_{m_i,n_j}$. Label-fit, assessed through human judgment, was measured by annotating a sample of 50 ads from each dataset based on whether the ads possess distinctive features characteristic of their respective labels. The ads are categorized into five groups as per Table 5, using broad guidelines inductively developed from initial data analysis. To quantify diversity, we use the following metrics. **Diversity Metrics**:

1. **Unique Lemmas**: Counting unique lemmas to measure lexical diversity.

2. **Self-BLEU**: We calculate Self-BLEU (Zhu et al., 2018) to measure diversity of lexical patterns as well as syntactic diversity to some extend.

3. **BERT Vendi-Score**: As an embedding-based method, we choose to calculate the Vendi-Score (VS) (Dan Friedman and Dieng, 2023), which measures dataset diversity based on the Shannon entropy of a similarity matrix. To create such a matrix we calculated the cosine distance based on the embeddings of the pooler output of a BERT model.

[6] https://huggingface.co/MoritzLaurer/mDeBERTa-v3-base-xnli-multilingual-nli-2mil7

5.5 Training and Testing

Each synthetic dataset undergoes training and testing five times to mitigate random variation. The training process involves fine-tuning a distilBERT (Sanh, 2019) model on both filtered and unfiltered datasets, following hyperparameters from Krüger (2023) (see Appendix C). Test data encompasses two distinct test datasets that are constant across all runs:

- **Qual-Testset**: A manually annotated dataset with 20 ads per class, measuring performance on real-world data.
- **Ausklasser-Testset**: Adopted from Krüger (2023), consisting of apprenticeship and non-apprenticeship OJAs, allowing performance comparison with models trained on real-world data. We aggregate all predictions to this binary label space the same way they did in their experiments.

6 Results and Discussion

In this section, we summarize and interpret the most important results. Supplementary metrics can be accessed in Appendix E. We specifically discuss our results concerning our research questions from Section 4. Figure 1 shows the F1[7] scores on the Qual-Testset for all $\mathcal{C}^{\mathrm{f}}_{m_i,n_j}$ across five runs each. The results show, concerning the first research question, that models trained on synthetically generated data can indeed classify real-world job ads well, achieving up to 96% F1 on our Qual-Testset. For the binary Ausklasser-Testset, some of our models even achieved 100% accuracy and are generally competitive with the models trained on 10,000 manually annotated job ads in Krüger (2023). However, the models also appear to be volatile, showing arbitrary behavior concerning method and size combinations. For example, the dataset $\mathcal{D}^{gf}_{\mathrm{Adj},5000}$ achieved only 59% F1 on average, despite having much better results with smaller datasets and in the two adjacent size categories also having slightly better performance when unfiltered.

This observation makes it difficult to answer research questions two and three. Due to the volatility the results cannot be viewed in an overly statistical manner. Specific comparisons of parameters, even with statistical significance testing, may not be meaningful due to the arbitrary nature of some outcomes, indicating the presence of factors we have not yet identified or a large random factor in SDG independent of the specific parameters. Such factors might be aspects of *Fidelity* or *Utility* as described in Yuan et al. (2024). Therefore, we will rather descriptively analyze the results. Table 1 shows that *Plain* has the highest overall mean F1 and a relatively high median, indicating that this method was relatively stable with fewer outliers compared to other methods, which have a larger difference between their mean and median. This might be explained by the fact that Sampling is more random by its nature and within the Prec and Adj methods, there is also some additional randomness in the prompting. It is plausible that, for example, certain adjectives from the list of adjectives in Adj did cause the model to output low quality data. If these adjectives were sampled frequently in dataset creation, the quality of the dataset would be lower compared to when they were sampled fewer times. However, given the limited number of adjectives and repeated sampling, it is statistically unlikely that any single adjective would have impact on the overall results as large as in the case described above for the $\mathcal{D}^{gf}_{\mathrm{Adj},7500}$ dataset. The random distribution and repeated appearances of each adjective mitigate the influence of individual adjectives on dataset quality. Analyzing the results further, the size factor played an important role as Figure 1 shows. Overall, results show that increasing the size parameter has improved scores initially, but all methods appear to have plateaued. Comparing our size to the results in Krüger (2023), models trained on synthetically generated data do not require more training for comparable performance. In the case of Prec, we observe that there were several outliers in the lower dataset size settings, but the results became much more stable with increased data.

Filtering had a slightly positive effect on both mean and median (3% F1 and 1% F1 gain respectively), but again the results are very volatile, because often, the effect was rather small, while sometimes it seemed to have a huge impact in both directions. For example in the Prec 500 setting, the models performed very well on the unfiltered data, but much worse on the filtered data. This variability can be attributed to the extent of data filtering; if too much data is removed, the remaining dataset may be insufficient to train a model that generalizes well.

[7] All reported F1 scores, precision, and recall refer to the macro-averaged metrics unless otherwise specified.

Figure 1: **F1 on Qual-Testset by Method and Size.** Proportion of our $\mathcal{LF}$ annotation labels per method.

Method	Mean F1	Median F1
Plain	0.89	0.90
Sampling	0.85	0.91
Prec	0.87	0.92
Adj	0.88	0.91

Table 1: Overall Mean and Median F1 Scores per Method on Qual-Testset

For our fourth and fifth research question, we have to consider the $\mathcal{DS}$ and $\mathcal{LF}$ metrics. Comparing the results of Unique Lemmas, Self-BLEU and BERT Vendi-Score across different metrics and sizes (see Table 2 for an overview and Appendix E for more in-depth results), we find that metrics are relatively stable across sizes, indicating that a given method will behave similar with respect to other methods regardless of size. Each metric singles out *Sampling* as producing the most diverse datasets. Since *Sampling* is known to increase diversity in text generation, this result is expected and shows that our metrics work as intended. Surprisingly however, the order of the other three methods differ depending on the metric. This proves that our fifth research question is highly relevant and the insight from our literature review in Section 3.2 that there is no single ground truth metric for measuring text diversity holds truth. It shows that SDG research using diversity is highly dependant on the metric chosen. As we have pointed out in Section 3.2, different metrics can measure different *aspects* of text diversity. We believe that our results show that this idea needs to be made more prominently within SDG diversity research. This behavior of our $\mathcal{DS}$ measures also means that we cannot answer our fourth research question. Any form of correlation between $\mathcal{DS}$, $\mathcal{P}$ and $\mathcal{LF}$ w.r.t $\mathcal{M}$ and $\mathcal{N}$ would always be dependant on the $\mathcal{DS}$ we choose.

Method	Mean BERT-VS ↑	Mean Self-BLEU ↓	Mean Unique Lemmas ↑
Plain	1.412	61.18	22227
Sampling	**1.526**	**35.99**	**36179**
Prec	1.430	60.32	21988
Adj	1.478	60.426	18568

Table 2: **Diversity scores.** Averaged $\mathcal{DS}$ per method. The arrow indicates whether a higher or lower score means that the data is more diverse.

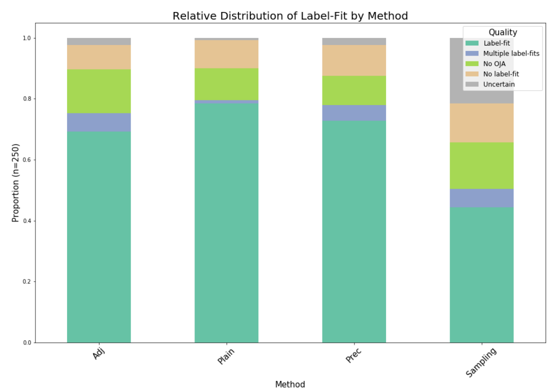

Figure 2: **Label-fit by method.** Proportion of our $\mathcal{LF}$ annotation labels per method.

At the same time, Figure 2 shows that *Plain* has the highest label-fit, whilst seemingly plateauing at a lower F1 score than the other methods. This could hint at datasets being generated by this method being less diverse. With this logic, BERT-VS and Self-BLEU would be more useful to measure diversity than counting unique lemamas, because they measure *Plain* as being less diverse than *Adj* and *Prec*. We do acknowledge, though, that such a claim requires further investigation, because the volatility described earlier hints at further factors still unknown to us are contributing to the $\mathcal{P}$. We also acknowledge that we randomly sampled fifty ads per dataset to manually annotate and our annotation process was relatively simple (single-blind). This makes the results of our label-fit less stable. Since our testsets are also rather small, our results are to be taken with caution.

Finally, during the manual data annotation, we had some intriguing qualitative observations, which we want to briefly summarize in the following. Most prominently, we see in Figure 2 that there is a portion of ten to twenty percent of ads labeled as not being OJAs. This, however, is to a large extend caused by a design choice in annotations. Most of these cases are actually partly a regular OJA, but then at some point turn into something else. The model seems to get confused and starts to generate other text genres related to jobs or job ads. Most frequently were job applications letters, forum posts or newspaper articles. In most of these cases, however, the job ad did start normally and did also contain a correct label-fit. Therefore, these data might still help the downstream model learn to distinguish between employment type labels to some extend.

7 Conclusion and Outlook

In this study, we explored synthetic data generation (SDG) methods to enhance the classification of employment types in online job advertisements (OJAs). Our experiments focused on four main strategies: plain prompting, sampling, precise attributes, and adjective attributes, while investigating the impact of dataset size, diversity, and label-fit on downstream model performance. Additionally, we examined the efficacy of a NLI filter in improving the quality of the synthetic data.

Our findings indicate that models trained on synthetically generated data can classify real-world job ads effectively, achieving competitive performance compared to models trained on manually annotated data. However, the results exhibited volatility, with significant fluctuations in performance depending on the method and dataset size combination. Our best performing model, trained on $\mathcal{D}^{gf}_{\text{Adj},7500}$, configuration achieved 96% F1 score on our Qual-Testset and 99% F1 on the Ausklasser-Testset. Despite this, the plain prompting method demonstrated the highest overall stability and mean F1 score, suggesting that simpler methods may yield more consistent results.

Data diversity and label-fit were measured using multiple metrics, revealing that the sampling method consistently produced the most diverse datasets. Nonetheless, the choice of diversity metric significantly influenced the evaluation, highlighting the need for careful consideration when selecting metrics for SDG research. Our label-fit analysis showed that while plain prompting achieved the highest label-fit, it did not necessarily correlate with the best performance, suggesting that a balance between diversity and label-fit is crucial.

Filtering synthetic data using NLI had a slightly positive overall impact on model performance, but its effect varied across different methods and dataset sizes. This suggests that while NLI filtering can improve data quality, its benefits may be context-dependent and require further optimization.

Overall, our most important finding is the volatility of our results. This indicates that there were additional factors influencing the outcomes of our results. Future work seek to identify those by performing more in-depth analyses on factors such as variance of label performance, the variance in different attributes in *Adj* and *Prec*, qualitative analysis of unexpected results like the poor performance of $\mathcal{D}^{gf}_{\text{Adj},5000}$ and in what way statistically as well as qualitatively the NLI filter influenced the datasets. Furthermore, our work shows the importance to develop more unified ways to measure text diversity and label-fit in SDG research.

8 Limitations

This section discusses the limitations of our study. Most importantly, we reported metrics across five runs during the training of each $\mathcal{C}^{\mathrm{unf}}_{m_i,n_j}$ and $\mathcal{C}^{\mathrm{f}}_{m_i,n_j}$ to mitigate randomness. However, during the generation of each $\mathcal{D}^{g}_{m_i,n_j}$ the sampling also introduces randomness. Therefore, if we want to analyze the impact of our input parameters, it would be better to also generate each $\mathcal{D}^{g}_{m_i,n_j}$ several times, which then each time goes through the rest of the pipeline. This, however, was beyond of the scope of this paper due to the major increase in computational cost this would entail.

There are two major limitations when it comes to our $\mathcal{P}$-measures. First, our $\mathcal{D}^{test}$ are relatively small, which generally makes results less reliable. Furthermore, our Qual testset was constructed by manually searching OJAs online, because we believed, based on the heavy label imbalance in OJA data w.r.t employment type, annotating data would result in heavy manual annotation effort. Therefore, our data comes from a relatively small time span, in which the OJAs went online. This could have introduced biases. Second, $\mathcal{P}$-measures are all calculated based on the same configuration w.r.t. several parameters, such as Hyperparameters or the choice of the pretrained model, which likely influence the performance $\mathcal{P}$. Especially using more sophisticated techniques could substantially improve results even further.

We also see limitations in the way we treat our label space. As mentioned in REF APPENDIX we derived the labels based on labor market expert opinions on what they thought were beneficial for OJA research. However, it can be debated whether we capture all different types of employment exhaustively. More importantly, it can be debated whether the categories we opened up are clearly distinguishable in all cases. For example, a PhD position may be full or part time. Also, in real world data it can occur that employers state some flexibility, for example by looking for an intern **or** a working student, which we do not account for in the way we treat the employment type classification. As our qualitative analysis shows the LLMs sometimes did generate instances like that, indicating that they can potentially be leveraged for a more sophisticated system.

One important consideration regarding our *Prec* method is that we did not consider the plausibility of our randomly sampled attribute combinations. For example, some employment types like *voluntary social year* might be extremely uncommon in certain industry sectors as they are typically associated with specific types of employers and organizations from the social sector. Prompting such unrealistic combinations might have negatively impacted data generation. Similarly, our list of adjectives for *Adj* did not have any scientific foundation, because we could not find any in the literature we considered, which included linguistic literature on discourse analysis, register analysis or genre linguistics as well as literature on corporate identity from economics. It is likely that some of the adjectives negatively impacted label-fit. We believe that studying aspects of text style and how to describe it could benefit SDG.

There are also limitations w.r.t. the way we measure $\mathcal{LF}$. Firstly, we only annotate a relatively small sample from our data. Secondly, we to the best of our knowledge there exist no public guidelines to aid such annotation for label-fit in synthetic data. We believe that sharing our experience annotating, however, can help other researchers in SDG that seek to manually examine their data. Developing a shared and more refined approach to annotation should be a goal in this research area.

References

2008. *Klassifikation der Wirtschaftszweige*. Statist. Bundesamt, Wiesbaden.

Danial Alihosseini, Ehsan Montahaei, and Mahdieh Soleymani Baghshah. 2019. Jointly measuring diversity and quality in text generation models. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 90–98.

Nesrine Bannour, Sahar Ghannay, Aurélie Névéol, and Anne-Laure Ligozat. 2021. Evaluating the carbon footprint of NLP methods: a survey and analysis of existing tools. In *Proceedings of the Second Workshop on Simple and Efficient Natural Language Processing*, pages 11–21, Virtual. Association for Computational Linguistics.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. 2021. Improving question answering model robustness with synthetic adversarial data generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8830–8848.

Conrad Borchers, Dalia Gala, Benjamin Gilburt, Eduard Oravkin, Wilfried Bounsi, Yuki M Asano, and Hannah Kirk. 2022. Looking for a handsome carpenter! debiasing gpt-3 job advertisements. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 212–224.

Samuel Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Yanan Chen and Yang Liu. 2022. Nli-based filtering for data augmentation in topic classification. In *2022 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, pages 103–110. IEEE.

Benjamin Clavié and Guillaume Soulié. 2023. Large language models as batteries-included zero-shot esco skills matchers. *Recsys in HR @ Recsys*.

Dan Dan Friedman and Adji Bousso Dieng. 2023. The vendi score: A diversity evaluation metric for machine learning. *Transactions on machine learning research*.

Maxime Delmas, Magdalena Wysocka, and André Freitas. 2024. Relation extraction in underexplored biomedical domains: A diversity-optimised sampling and synthetic data generation approach. *Computational Linguistics*, pages 1–49.

Tatsunori B Hashimoto, Hugh Zhang, and Percy Liang. 2019. Unifying human and statistical evaluation for natural language generation. 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 1689–1701.

Martin Josifoski, Marija Sakota, Maxime Peyrard, and Robert West. 2023. Exploiting asymmetry for synthetic training data generation: Synthie and the case of information extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1555–1574.

Kai Krüger. 2023. Ausklasser-a classifier for german apprenticeship advertisements. In *Proceedings of the Communication Papers of the 17th Conference on Computer Science and Intelligence Systems*, volume 36. IEEE Piscataway, NJ, USA.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. *Preprint*, arXiv:2309.06180.

Yi-An Lai, Xuan Zhu, Yi Zhang, and Mona Diab. 2020. Diversity, density, and homogeneity: Quantitative characteristic metrics for text collections. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1739–1746.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. 2023. Synthetic data generation with large language models for text classification: Potential and limitations. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Andreas Liesenfeld and Mark Dingemanse. 2024. Rethinking open source generative ai: open washing and the eu ai act. In *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, pages 1774–1787.

Antoine Magron, Anna Dai, Mike Zhang, Syrielle Montariol, and Antoine Bosselut. 2024. Jobskape: A framework for generating synthetic job postings to enhance skill matching. In *1st Workshop on Natural Language Processing for Human Resources*. Association for Computational Linguistics.

Yu Meng, Jiaxin Huang, Yu Zhang, and Jiawei Han. 2022. Generating training data with language models: Towards zero-shot language understanding. *Advances in Neural Information Processing Systems*, 35:462–477.

V Sanh. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *Proceedings of Thirty-third Conference on Neural Information Processing Systems (NIPS2019)*.

Timo Schick and Hinrich Schütze. 2021. Generating datasets with pretrained language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6943–6951.

Maximilian Schmidhuber and Udo Kruschwitz. 2024. Llm-based synthetic datasets: Applications and limitations in toxicity detection. *LREC-COLING 2024*, page 37.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

Guy Tevet and Jonathan Berant. 2021. Evaluating the evaluation of diversity in natural language generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 326–346, Online. Association for Computational Linguistics.

Veniamin Veselovsky, Manoel Horta Ribeiro, Akhil Arora, Martin Josifoski, Ashton Anderson, and Robert West. 2023. Generating faithful synthetic data with large language models: A case study in computational social science. *arXiv preprint arXiv:2305.15041*.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Yiben Yang, Chaitanya Malaviya, Jared Fernandez, Swabha Swayamdipta, Ronan Le Bras, Ji-Ping Wang, Chandra Bhagavatula, Yejin Choi, and Doug Downey. 2020. Generative data augmentation for commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1008–1025.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022. Zerogen: Efficient zero-shot learning via dataset generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11653–11669.

Yue Yu, Yuchen Zhuang, Jieyu Zhang, Yu Meng, Alexander J Ratner, Ranjay Krishna, Jiaming Shen, and Chao Zhang. 2024. Large language model as attributed training data generator: A tale of diversity and bias. *Advances in Neural Information Processing Systems*, 36.

Yefeng Yuan, Yuhong Liu, and Liang Cheng. 2024. A multi-faceted evaluation framework for assessing synthetic data generated by large language models. *arXiv preprint arXiv:2404.14445*.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 1097–1100, New York, NY, USA. Association for Computing Machinery.

A Employment Type Classification

Label Name	Label Nr	Translation
Praktikum	0	Internship
Freiwilliges Soziales Jahr	1	Voluntary Social Year
Volontariat	2	Voluntary Service
reguläre Vollzeitstelle	3	Regular Full-Time Position
Ausbildungsstelle	4	Apprenticeship Position
Promotionsstelle	5	Doctoral Position
Teilzeitstelle	6	Part-Time Position
Werksstudentenstelle	7	Working Student Position
Traineestelle	8	Trainee Position

Table 3: Class label overview

Table 3 shows the labels we derived for employment type classification. The choice of these labels was motivated by consultations with experts in labor market research, but does not claim to be the exhaustive ground truth to employment types. Also, not all labels are easily translatable to English, because some of them are specific to the Germany.

B Data Generation

Data generation was executed using the falcon-40b[8] model using default parameters from the huggingface transfomers pipeline, except for top-k sampling (5, 50 as described above) and max_token=512 configuration. We wrapped the model in the VLLM[9] library, where we set the *dtype* parameter to *half*, which means using 16-bit floating-point precision, and *tensor-parallel-size* to two as we ran our code on two NVIDIA RTX A6000 GPUs.

Below, we list the (translated) templates and explain the variables. For the original templates as well as a full list of all possible input values, we refer to the source code. All parameters we randomly sampled with the constrains specified below, except for the *input_class*. Here, we took the overall number of ads to be generated (depending on the *size* parameter) and divided it by the number of input classes (nine) such that each generated dataset had an even label distribution.

B.1 Plain

```
{
    "prompt": "A job ad for a {
        input_class}",
    "input_class": "The employment type
        categories we use in this paper
        .",
    "top_k": 5
}
```

[8] https://huggingface.co/tiiuae/falcon-40b

[9] https://github.com/vllm-project

B.2 Sampling

```
{
    "prompt": "A job ad for a {
        input_class}.\n Job ad:\n"
    "input_class": The employment type
        categories we use in this paper.
    top_k: 50
}
```

B.3 Prec

```
{
    "prompt": "A job ad for a {
        input_class}.\n
    {mainModule}\n
    {lenModule}\n
    {infoModule}\n
    {styleModule}\n
    Job ad:\n"
    "input_class": The employment type
        categories we use in this paper.
    "mainModule": This was dependant on
        the type of input classes. For
        most input classes, the prompt
        here was: 'industry sector of
        the searching company: {industry
         sector}'. The industry sector
        was sampled the German industry
        sector taxonomy \textit{
        Klassifikation der
        Wirtschaftszweige 08}. However,
        for the apprenticeships we
        instead specified the type of
        apprenticeship instead.
        Apprenticeships are highly
        formalized in Germany and there
        is a finite amount of official
        apprenticeship programs
        available. For the PhD class we
        instead used a list of research
        subject sampled from WikiData.
    "lenModule" : We specified the
        length the ad should have.
        Lengths were always a
        descriptive word (e.g.: \textit{
        long}, \textit{short}) as well
        as a range of words (e.g. \
        textit{100 to 150 words}).
```

```
    "infoModule": We sampled from a list
        of zones typically found in
        OJAs (e.g.: \textit{company
        description}, \textit{job tasks},
         \textit{contact information}).
    "styleModule": One of four styles
        the language of the job ad
        should have. Simlar to Adj, but
        less creative.
    top_k: 5
}
```

B.4 Adj

```
{
    "prompt": "A job ad for a {
        input_class}.\n
    Characterstics: {sampled adjectives
        }\n
    Job ad:\n"
    "input_class": The employment type
        categories we use in this paper.
    "sampled adjectives": Two to five
        randomly sampled adjectives
        describing the style of OJAs
        from a list of 30 adjectives.
    top_k: 5
}
```

C Training Parameters

For the downstream training, we fine-tune a German distilBERT[10] model with the hyper-parameters specified in Table 4. All other hyper-parameters were set to default. For the test metrics, we calculated macro F1, Precision and Recall.

Hyperparameter	Value
num_train_epochs	4
learning_rate	0.0001
per_device_train_batch_size	8
per_device_eval_batch_size	8
warmup_steps	False

Table 4: Hyperparameters used for LLM training with HuggingFace

D Label-fit Annotation

Label-fit annotation was done by three annotators in a single blind annotation process. We randomly sampled 50 texts from the filtered datasets for each $\mathcal{D}^{gf}_{m_i,n_j}$. Each time, the annotator was given the choice between five labels as detailed in Table 5.

Label Name	Label Nr	Explanation
label-fit	0	The job ad fits the label.
no label-fit	1	The job ad does not fit the label.
double label-fit	2	The job ad contains features for two or more labels, including the input label (e.g., "We seek an intern or an apprentice").
no job ad	3	Instances where the model fails to generate a job ad, producing an unrelated text type.
unsure	4	Cases where annotators are uncertain, requiring further review.

Table 5: **Label-Fit Category Descriptions.** These instructions were given to the annotators.

E Supplementary Results

Tables 6 and 7 show the average F1 performances across all method/size combinations on the two testsets respectively. Figures 3 to 5 plot the results aggregated for filtering, methods and size respectively.

[10] https://huggingface.co/distilbert/distilbert-base-german-cased

Method	**Size**	**Filtered F1 Score**		**Unfiltered F1 Score**	
		Mean	**Median**	**Mean**	**Median**
Plain	500	0.84	0.88	0.86	0.83
	1000	0.88	0.88	0.89	0.89
	2500	0.92	0.93	0.92	0.92
	5000	0.92	0.92	0.86	0.87
	7500	0.91	0.91	0.90	0.90
Sampling	500	0.70	0.73	0.53	0.53
	1000	0.86	0.89	0.91	0.91
	2500	0.93	0.93	0.83	0.83
	5000	0.93	0.93	0.93	0.93
	7500	0.93	0.93	0.95	0.95
Prec	500	0.58	0.70	0.90	0.90
	1000	0.84	0.89	0.81	0.81
	2500	0.88	0.90	0.95	0.95
	5000	0.93	0.93	0.93	0.93
	7500	0.93	0.94	0.94	0.94
Adj	500	0.86	0.88	0.87	0.87
	1000	0.90	0.91	0.90	0.90
	2500	0.88	0.91	0.93	0.93
	5000	0.94	0.95	0.59	0.59
	7500	0.95	0.95	0.96	0.96

Table 6: F1 Score Statistics on Qual-Testset by Method and Size

Method	Size	Filtered F1 Score		Unfiltered F1 Score	
		Mean	Median	Mean	Median
Plain	500	0.93	0.95	0.88	0.86
	1000	0.90	0.90	0.93	0.96
	2500	0.96	0.97	0.92	0.91
	5000	0.95	0.95	0.93	0.95
	7500	0.96	0.97	0.95	0.95
Sampling	500	0.85	0.95	0.65	0.65
	1000	0.94	0.95	0.92	0.92
	2500	0.98	0.99	0.85	0.85
	5000	0.95	0.95	0.94	0.94
	7500	0.96	0.96	0.96	0.96
Prec	500	0.72	0.90	0.94	0.94
	1000	0.91	0.91	0.77	0.77
	2500	0.88	0.91	0.91	0.91
	5000	0.94	0.94	0.96	0.96
	7500	0.94	0.95	0.92	0.92
Adj	500	0.94	0.94	0.87	0.87
	1000	0.97	0.97	0.86	0.86
	2500	0.96	0.97	1.00	1.00
	5000	0.95	0.95	0.33	0.33
	7500	0.98	0.97	0.99	0.99

Table 7: F1 Score Statistics on Ausklasser-Testset by Method and Size

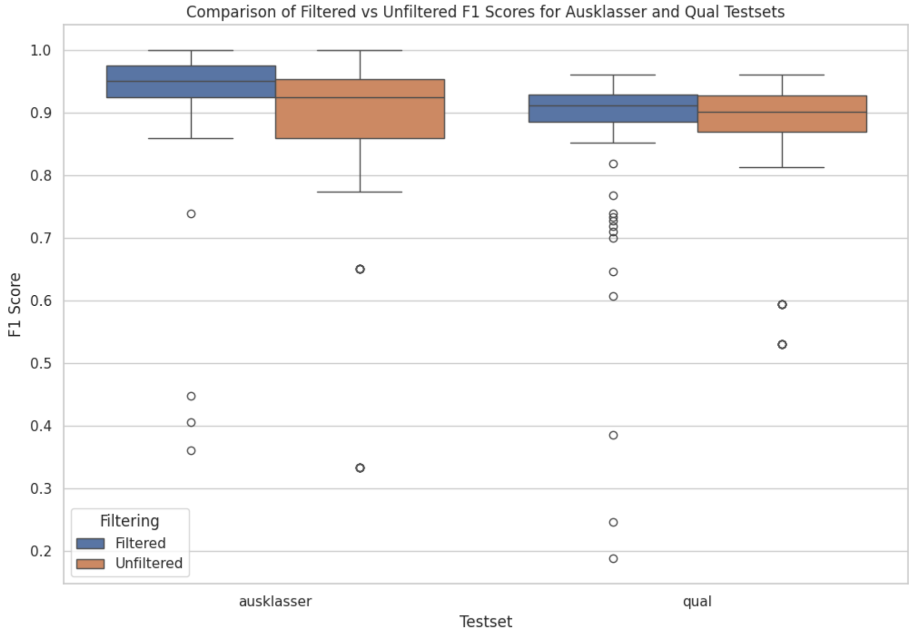

Figure 3: **Filtered versus unfiltered.** Compares the runs on $\mathcal{D}^{g}_{m_i,n_j}$ against $\mathcal{D}^{gf}_{m_i,n_j}$. as per F1 score on the Qual-Testset and Ausklasser-Testset

.

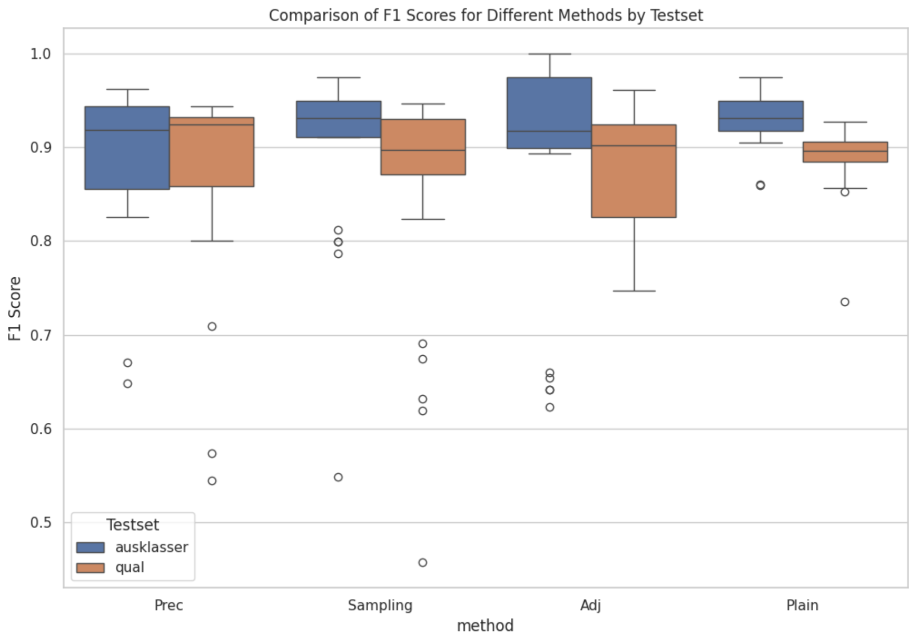

Figure 4: **Method Comparison.** Compares F1 scores across methods on Ausklasser- and Qual-Testset.

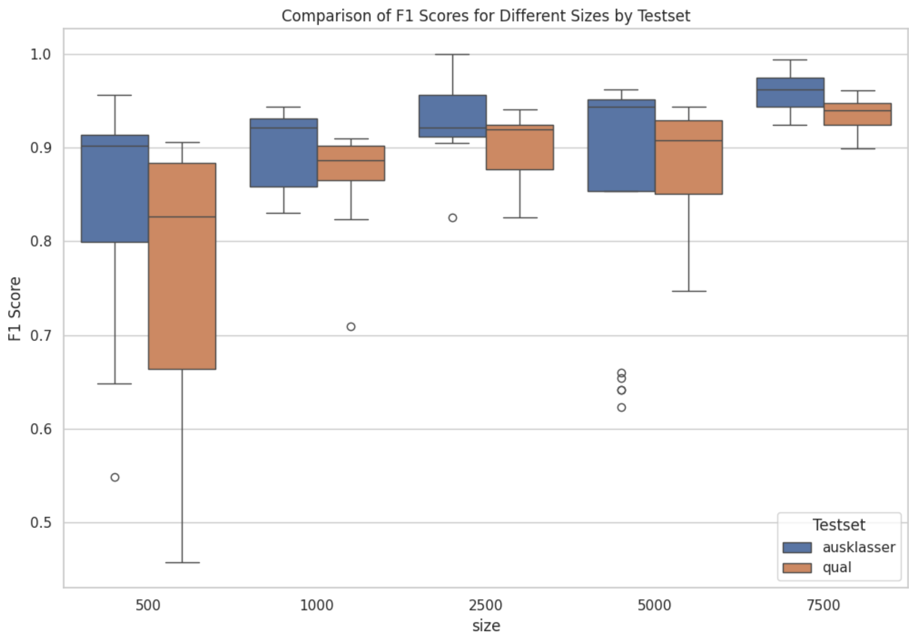

Figure 5: **Size Comparison.** Compares F1 scores across sizes on Ausklasser- and Qual-Testset.