Large Language Models for Detecting Bias in Job Descriptions

Tristan Everitt*
Dublin City University

Paul Ryan**
Dublin City University

Brian Davis†
ADAPT Research Centre

Kolawole J. Adebayo‡
ADAPT Research Centre

This study explores the application of large language (LLM) models for detecting implicit bias in job descriptions, an important concern in human resources that shapes applicant pools and influences employer perception. We compare different LLM architectures—encoder, encoder-decoder, and decoder models—focusing on seven specific bias types. The research questions address the capability of foundation LLMs to detect implicit bias and the effectiveness of domain adaptation via fine-tuning versus prompt-tuning. Results indicate that fine-tuned models are more effective in detecting biases, with Flan-T5-XL emerging as the top performer, surpassing the zero-shot prompting of GPT-4o model. A labelled dataset consisting of verified gold-standard, silver-standard, and unverified bronze-standard data was created for this purpose and open-sourced[1] *to advance the field and serve as a valuable resource for future research.*

1. Introduction

Organisations strive to promote diversity and inclusivity, driven by the benefits to company culture, stereotype reduction, and compliance with legal standards. An industry report revealed a statistically significant correlation between diversity metrics and financial performance. Specifically, the findings indicated that organisations ranking highest in cultural diversity and gender diversity were 35% and 15% more likely, respectively, to surpass median financial returns (Hunt, Layton, and Prince 2015).

In human resources, bias affects both employers and employees in explicit and implicit forms (Fridell 2017). Explicit bias is conscious and controllable, but can be illegal in employment contexts. Implicit bias is subtle, unconscious, and harder to address (Fiske and Lee 2008; Cunningham and Cunningham 2022; Storm et al. 2023). Implicit bias in job descriptions is a major concern as it shapes the applicant pool and influences applicants' decisions. Bias in the language of job descriptions can affect how attractive a role appears to different individuals and can impact employer perception. The challenge is to efficiently identify and mitigate these biases.

The application of large language models (LLMs) for detecting bias in job descriptions is a promising yet underexplored area. This study investigates the effectiveness of LLM

* DCU School of Computing, Dublin, Ireland. E-mail: tristan@ep9.io
** DCU School of Computing, Dublin, Ireland. E-mail: paulmartinryan@gmail.com
† ADAPT Research Centre, Dublin, Ireland. E-mail: brian.davis@adaptcentre.ie
‡ ADAPT Research Centre, Dublin, Ireland. E-mail: kolawole.adebayo@adaptcentre.ie
1 Dataset Repository: https://huggingface.co/2024-mcm-everitt-ryan

architectures[2] with fewer than 10 billion parameters in detecting implicit bias. The chosen model sizes and architectures were selected to investigate how different designs perform given the resource constraints.
A comprehensive evaluation is conducted on the models' performance across diverse in-context learning scenarios (Brown et al. 2020; Wang et al. 2023; Wei et al. 2022), focusing on their adaptability and generalisability in various settings. Specifically, we assess the models' performance both with and without fine-tuning for domain adaptation, providing insights into their ability to identify implicit bias.
We conceptualise the task of identifying implicit bias in job descriptions as a multi-label classification problem, where each job description is assigned a subset of labels from a set of eight categories—age, disability, feminine, masculine, general exclusionary, racial, sexuality, and neutral. This study investigates two primary research questions:

1. *Can foundation LLMs accurately detect implicit bias in job descriptions without specific task training?* We evaluate the performance of three topical decoder-only models under four distinct prompt settings, assessing their ability to extract relevant information from job descriptions and identify implicit bias.
2. *Does domain adaptation via fine-tuning foundational LLMs outperform prompt tuning for detecting implicit bias in job descriptions?* We fine-tune models with varying architectures as text classifiers on task-specific data and compare their performance to that of prompt-tuned models.

Central to the research is the creation of a labelled job descriptions dataset. We employed the services of two graduate students knowledgeable on the task to work on the annotation process. The manual annotation of real job descriptions, which took 76 hours, was undertaken to produce a subset of gold-standard manually verified genuine labelled data. Synthetic data was generated using large-scale language models. A subset of the synthetic data was manually annotated over 92 hours to produce silver-standard, manually verified labelled data, which was then used to augment minority classes. The remainder of the synthetically generated data constitutes bronze-standard unverified labelled data.

The paper is presented as follows. Section 2 contains a brief review of related literature. Sections 3 and 4 detail the dataset and experimental methodology. Section 5 describes how we conducted the research experiments. Section 6 presents our results and an analysis of the outcomes. We conclude the research in Section 7.

2. Related Work

In the domain of job descriptions, the use of phrases and their context can impact an applicant's perception of the employer and make them more or less attractive to certain individuals. Gender-biased language in job descriptions, for instance, has been studied for its impact on the number of women applying for leadership positions (Horvath and Sczesny 2016). Another study examined the effect of gender-specific wording and its role in sustaining gender disparities in typically male-dominated occupations (Gaucher, Friesen, and Kay 2011). Research has shown that age-specific wording in job descriptions significantly attracts younger applicants (Burn et al. 2022). The wording within job descriptions not only shapes perceptions around specific biases such as gender and

2 Encoder, encoder-decoder, and decoder architectures.

age but can also attract individuals with communal narcissistic tendencies, highlighting how job description wording can resonate with individuals (Fatfouta 2023).

A simple method to identify bias is lexicon or keyword matching, which identifies specific words, word stems, or phrases predetermined to be associated with biased language. Regular Expressions have been discovered to remain a potent tool for identifying illegal age discrimination in Dutch job descriptions, given the predictability of age-coded words (Pillar, Poelmans, and Larson 2022). However, the basic find-and-replace approach does not scale well as individual words can carry several meanings depending on context, thereby requiring the use of Natural Language Processing (NLP) techniques. A study employed part-of-speech tags, lemmas, relative word position, and linguistic lexicons for subjectivity and modality for detecting biased language in Wikipedia articles (Recasens, Danescu-Niculescu-Mizil, and Jurafsky 2013). Two principal types of bias within Wikipedia are identified: epistemological bias, which manifests subtly through linguistic subtleties, and framing bias, which is explicit and conveyed through subjective language. Similarly, another study used part-of-speech tags, sentiment analysis, and the identification of verbs often associated with biased expressions to identify biased language in Wikipedia articles (Hube and Fetahu 2018). The study used "seed" words to identify potential biased statements and, with word embedding models, created a lexicon of words that frequently indicate bias in articles. Furthermore, a study employed part-of-speech tags and sentiment analysis, alongside named entity types and token characteristics, as well as word embeddings, to develop textual features (Frissen, Adebayo, and Nanda 2023).

The advent of large pre-trained language models has revolutionised the field of natural language processing (NLP), offering new avenues for improving the performance of existing methodologies (Devlin et al. 2019; Radford et al. 2019). A key question that arises is whether the context-aware capabilities of these models can surpass traditional NLP techniques in detecting subtle biases in job descriptions. Furthermore, innovative prompting strategies, such as *chain-of-thought* prompting, have shown promise in improving performance on tasks that require reasoning (Wei et al. 2022). Recent studies have shown that sufficiently large large language models can perform well on multiple tasks without requiring task-specific training (Radford et al. 2019). Additionally, few-shot settings have proven effective for in-context learning, enabling models to improve on a task when provided with relevant examples (Brown et al. 2020).

Several studies have explored the potential of large language models for various NLP tasks. For instance, a study on GPT-3, a 175B-parameter model, evaluated its performance under few-shot settings and demonstrated its ability to learn from task-specific exemplars within the prompt (Brown et al. 2020). Another study investigated the use of ChatGPT for data annotation and found that it can outperform human annotators in terms of accuracy, while also reducing time and financial costs (Gilardi, Alizadeh, and Kubli 2023). These findings are particularly relevant to our study, as they establish the effectiveness of in-context learning for text classification.

Despite these advances, the question remains whether LLMs can effectively detect implicit bias in job descriptions. Our study aims to address this research gap by investigating the ability of LLMs to identify such subtle biases, and explore the effectiveness of different prompting strategies and domain adaptation techniques for improving their performance. By leveraging the context-aware capabilities of LLMs, we seek to develop a more accurate and efficient approach to detecting implicit bias in job descriptions.

A comprehensive review of the literature reveals that the use of large language models for detecting bias in job descriptions is a relatively unexplored area of research. To date, only one study has investigated the application of large language models for

this purpose, focusing specifically on age and gender-related biases in Australian job descriptions (Mao, Tan, and Moieni 2023). However, this study has several limitations that our research aims to address. Firstly, the study did not include encoder-decoder architectures, which have been shown to be effective in various NLP tasks. Secondly, the study uses a decoder model, GPT-2 (Radford et al. 2019), that has since been surpassed in terms of performance by more recent models. Finally, it does not investigate the impact of different prompting strategies on the detection of bias in job descriptions.

Our study seeks to fill these gaps in the literature by making several key contributions. Firstly, we include a diverse sample of job descriptions from multiple countries, allowing us to examine the generalisability of our findings across different cultural and linguistic contexts. Secondly, we address a broader range of biases, including racial, sexuality, and disability-related biases, in addition to age and gender-related biases often tackled in prior works. Finally, we incorporate encoder-decoder architectures and recent state-of-the-art decoder models in place of GPT-2, thereby enabling us to investigate the impact of architecture and model sizes in addition to prompting methods. This results in a comprehensive understanding of the promises and limitations of applying LLMs for bias detection and fostering inclusive hiring in the job market space.

3. Dataset

A requirement of this study is to undertake a multi-label classification task to identify seven distinct types of bias, as well as a neutral label. However, since we are unaware of any existing dataset that includes labelled data for this specific purpose, it is necessary to create a new dataset tailored to the needs of this study. Moreover, given that bias can manifest differently across various cultures and legal frameworks, it is important to include job descriptions from a wide range of countries. This section outlines the procedures undertaken to create the dataset.

3.1 Data Gathering, Cleaning, & Preprocessing

We utilised a public job descriptions dataset (Techmap.io 2020–2023), which avoided web scraping across various international websites (Appendix `C-A1`). The dataset consisted of job descriptions from Ireland (October 2020, 2021, 2022), approximately 3.4 million global samples from September 2021, and 33,000 USA postings from May 2023.

The dataset required extensive preprocessing to extract the necessary information. We performed the following steps:

1. Data Extraction. Relevant details such as country, position name, and raw HTML were extracted from the dataset.

2. Remove Duplicates and Language Detection. Removing duplicates reduced the dataset from 3.4 to about 2.6 million samples from 83 countries. The United States constituted nearly a third of the samples[3]. Language detection (Stahl 2023) resulted in 56 languages within the dataset, with English at 69.1%, German at 15.8%, and Russian at 10.6%. As the focus of the study is on English bias and non-inclusive language, non-English samples were discarded, further reducing the dataset to about 1.76 million samples.

3 USA 31.4%; DEU 16.2%; GBR 15.5%; RUS 6.1%; AUS 5.8%

3. HTML Preprocessing. We removed broken links and invalid HTML tags that interfered with an HTML-to-Text parser (Hedley 2023) for text extraction [4]. We also removed excess white spaces, added missing full-stops, and corrected punctuation spacing where applicable.

4. Text Cleaning and Filtering. We used regular expressions (Appendix D) to address problematic phrases that remained after the initial cleaning process. This step resolved issues with extra white spaces, punctuation spacing, duplicated special characters e.g., having €€€€ instead of €, emojis, accents, and diacritical marks. This ensures that we obtain well formatted sentences and remove noises that do not contribute to the context.

5. Potential Bias Filtering. To reduce the dataset further, we used a list (Appendix C–A4) of 641 biased terms aggregated from previous studies (Gaucher, Friesen, and Kay 2011; Burn et al. 2022; OFCCP 2024; Bruce 2009; Ongig Team 2024; Gill 2020; Frissen, Adebayo, and Nanda 2023). We selected only samples containing one or more of these terms and discarded the rest. We divided the samples into sections (phrases) of up to 400 words, rounded to the nearest complete sentence and only retained those containing between 3 and 400 words; a character count between 10 and 3000; and having not more than 20 phrases per sample. This process cleaned out about 4% of the original samples.

6. Deduplication. Due to multiple listings on various platforms, updates, revisions, and agency postings, duplicate and near-duplicate samples are inevitable. An embedding-based approach (Douze et al. 2024; Reimers and Gurevych 2019) identified and removed semantically similar texts by creating an embedding vector for each sample and calculating the squared Euclidean (L2) distance between vectors. This distance, calculated by summing squared vector values, ranges from zero (identical vectors) to infinity (most different). Samples with a distance less than 0.3 were considered duplicates, retaining one copy and discarding the rest.

3.2 Dataset Statistics

The top ten countries are shown in Table 1. A total of 19 million potentially biased terms were identified across the samples, with a breakdown by category shown in Figure 1. Therefore, the samples at this stage are considered potentially biased, as they contain at least one of the 641 biased terms.

3.3 Data Labelling, Anonymisation, Augmentation

After cleaning the dataset and reducing it to potentially biased phrases, the data needs to be labelled, as it is currently unsuitable for training.

Data Labelling The main challenges of labelling the dataset are:

1. Bias and non-inclusive language can often be subtle and require domain knowledge to accurately identify their various forms.

4 Example: `<li>` instead of `<li>`; or `/xe2/x80/x99` instead of a right single quotation mark.

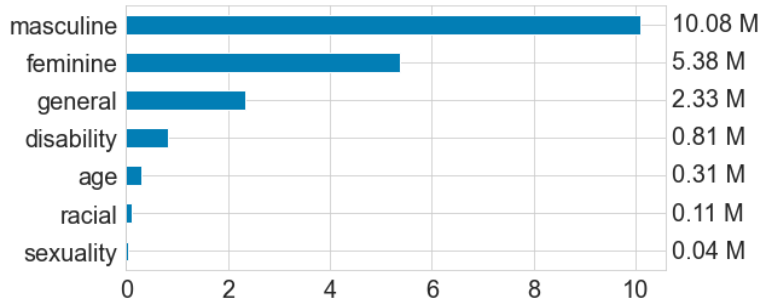

Figure 1
Potential Bias Terms (in Millions)

Table 1
Country Distribution of Samples

Country	Count	Percentage
United States	643,648	38.0%
United Kingdom	594,561	35.11%
Australia	128,427	7.58%
Ireland	102,314	6.04%
Canada	34,297	2.03%
New Zealand	31,846	1.88%
Singapore	19,635	1.16%
India	19,584	1.16%
Germany	19,035	1.12%
Hong Kong	13,938	0.82%

2. Annotating 2.5 million phrases across 1.69 million samples is a massive task given our resources.
3. Human labelling can be slow, error-prone, and may introduce the annotator's own unconscious bias.

To address the first challenge, we reviewed the 641 biased terms to improve our understanding of identifying bias. We added 34 new terms and rationales (Appendix C–A4) explaining why these terms can be considered biased. For the 641 existing terms, GPT-4 Turbo (OpenAI 2023) provided provisional rationales, which we then sampled a small number and refined manually. This process served two purposes:

1. Manually verifying synthetic rationales and cross-checking them against existing literature increased domain knowledge (Gaucher, Friesen, and Kay 2011; Burn et al. 2022; OFCCP 2024; Bruce 2009; Ongig Team 2024; Gill 2020; Frissen, Adebayo, and Nanda 2023).
2. Allowed us to collect short biased phrases that would later be used to synthesise biased samples.

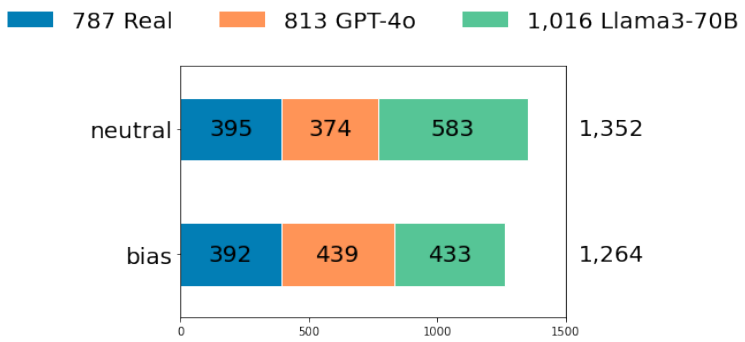

Figure 2
Annotated Samples: Bias vs Neutral

To mitigate the second challenge of labelling the dataset, a small random subset of the 2.5 million phrases was manually annotated. To avoid excluding other countries due to the high representation of the US and the UK, a two-round sampling was conducted instead of random sampling. The first round balanced the dataset by country, and the second ensured balanced representation of bias categories by focusing on the number of bias terms.

The third challenge of labelling data cannot be entirely overcome, but best efforts were made to mitigate it. Samples were annotated by presenting them along with details of the country, job role, detected biased terms, and their provisional/verified rationales (Tkachenko et al. 2020-2022).

After manually reviewing 787 real samples, the bias categories remained skewed. Age bias was most prevalent with 161 samples, while feminine bias was the least common with 31 samples. Options considered included further manual labelling and training a classifier to identify additional samples. However, minority class classification remained suboptimal. To address the class imbalance, the chosen approach involved synthetic oversampling along with manual verification, which will be discussed in more detail in Section 3.3 under Data Augmentation. Manually verified samples, both real and synthetic, illustrating the comparison between biased and neutral samples are presented in Figure 2 and the distribution of bias across categories is shown in Figure 3.

Data Anonymisation To prevent personally identifiable information (PII) from leaking into model training, we implemented a data anonymisation process. Initially, regular expressions were used, but they failed to catch all details. We then exported the dataset to a human-readable format and used version control to track changes. GPT-4o was used to replace PII with placeholders such as *[Name Redacted]* and *[Email Redacted]*. To save costs and improve speed, the system returned <*SKIP*> if no PII was found, avoiding unnecessary text generations. An iterative review ensured the process had not inadvertently altered non-PII text. Any errors were reverted. Finally, random sampling was performed to verify that the automated cleaning process had not missed any PII.

Data Augmentation In this study, GPT-4o and Meta Llama3 70B (OpenAI 2024; AI@Meta 2024) were utilised to generate 3,480 synthetic samples. Of these, 1,829 were version controlled, manually verified, and augmented to achieve a better balance among

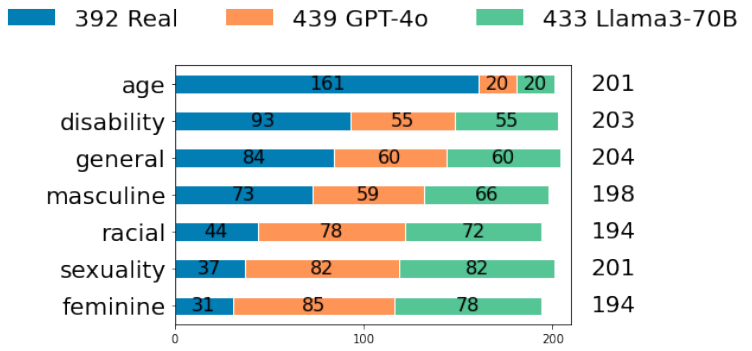

Figure 3
Annotated Samples by Bias Category

the classes. The remaining 1,651 unverified samples were used solely to supplement the training data. Therefore, it was crucial to generate high-quality samples to avoid introducing artefacts or creating instances that overlap with the majority classes. Three sequential iterations of data augmentation were conducted, with each attempt adjusted based on findings from the previous iteration. However, only the final iteration contributed to the dataset, as the first two were unsatisfactory attempts.

Iteration 1 introduced specific biases into the samples by prompting the model to demonstrate biased language within a particular category. However, it led to unrealistic outputs with excessive exclusionary language and multiple unintended biases instead of the targeted bias. Additionally, recurring themes such as family-oriented narratives and binary gender norms dominated some categories, thus limiting variety.

Iteration 2 of data generation involved refining prompts by incorporating the 675 verified/provisional rationales from Section 3.3 (See Appendix E–A for prompts). This approach guided the model on how to include specific biased terms within its output. However, this approach had several unintended consequences. The use of rationales to inject biases into the samples resulted in a skewed dataset, with a disproportionate number of samples falling into categories with the most terms. This led to repeated themes similar to those observed in the first iteration, and generated overly biased samples that failed to generalise. One notable issue that arose from this approach was the misclassification of certain phrases. For example, a classifier would categorise any mention of language in a sample as racially biased, even if it was not inherently so. This issue was traced to the term *native*, which the rationale suggested might be non-inclusive due to its implication of preference for certain individuals. As a result of this issue, phrases such as *Native English Speaker* were included in samples, but the classifier focused on the language aspect rather than the exclusionary nature. This caused phrases such as *Proficient in English* to be misclassified as racially biased. This highlights the importance of considering the context and nuances of language when generating biased samples. Furthermore, the samples generated with the rationales were mainly exclusionary, lacking neutral examples for contrast. This limited the ability of the classifier to learn from the data and make accurate predictions. These findings

underscore the need for careful consideration of the data generation process and the importance of including neutral examples in the dataset.

Iteration 3 of data generation, we built upon the second iteration by introducing trichotomy of phrases per bias category: negative, neutral, and positive. This approach allowed us to capture a more nuanced representation of implicit bias in job descriptions.

- *negative*: Phrases that were discriminatory, exclusionary, or subtly favoured certain groups over others. For example, "He should be adept at problem-solving." (gender-coded language).
- *neutral*: Phrases that were unbiased, equitable, and did not favour any particular group. For example, "The ideal candidate should have excellent problem-solving skills".
- *positive*: Phrases that, due to tokenism or misguided good intentions, may unintentionally introduce bias and deter applicants who feel excluded. For example, "We are focused on hiring … to enhance our team diversity.".

An exception to the trichotomy were masculine and general biases which consisted of a dichotomy of *negative* and *neutral*. We did not observe positive discrimination or tokenism towards masculinity in real data, and therefore excluded it from the trichotomy.

A total of 2,136 phrases (Appendix C–A5) were developed across seven bias categories and their groupings (negative, neutral, positive). These phrases included examples from real samples, synthetic variations of those examples, and purely synthetic creations. After manual review and adjustment, these phrases were used to generate full synthetic samples targeting a particular implicit bias or no bias at all.

After revising the data generation process, we reviewed a new set of samples to assess their quality and suitability for training data. Our evaluation revealed that the revised samples addressed previous concerns, exhibiting more subtle bias without overlapping majority classes. Only a few samples required adjustments, which increased our confidence that the unverified samples would be suitable for the training data. This suggests that the revised data generation process was effective in producing high-quality samples that accurately represent implicit bias in job descriptions. However, we noted an issue when generating samples using the GPT-4 model (OpenAI 2024), where artefacts were occasionally introduced by the removal of offensive terms, such as *homosexual*, despite the intention to include them. This problem was more common with GPT-4, but occurred inconsistently in both GPT-4 and Llama3 models, likely due to the models' intentions to avoid offensive content (Rebedea et al. 2023). Interestingly, we found that when the models explained why content was deemed offensive or biased, they became more lenient, retaining the offensive material. This suggests that providing models with the opportunity to justify their decisions can help to mitigate the introduction of artefacts and improve the overall quality of the generated samples. Our findings have implications for the development of AI models that aim to detect and mitigate implicit bias in job descriptions and highlight the importance of allowing models to explain their decisions, which can help to improve their performance and reduce the introduction of artefacts.

3.4 Composition of the Final Dataset

The final dataset consists of 4,267 samples, organised into verified and unverified groups. The verified set includes 2,616 manually annotated samples, with 787 real samples (Gold) and 1,829 synthetic samples (Silver). The remaining 1,651 samples are

Table 2
Gold (verified), Silver (verified), and Bronze (unverified) samples∗

Type	age	disability	feminine	general	masculine	neutral	racial	sex.
Gold	161	93	31	84	73	395	44	37
Silver	40	110	163	120	125	957	150	164
Bronze	440	451	405	517	422	0	469	466
Total	641	654	599	721	620	1352	663	667

∗*Gold are real job descriptions, while Silver and Bronze are synthetic.*
∗*Biased (non-neutral) samples can have multiple labels.*

unverified annotated synthetic data (Bronze). A detailed breakdown of sample distribution by label and data collection methods is provided in Table 2.

4. Methodology: Model Architectures

This section covers the model architectures and testing methodologies used to evaluate the effectiveness of LLMs in detecting implicit bias in job descriptions. This reflects the study's focus on comparing methods with and without domain adaptation.

4.1 Model Architecture Overview

The models selected for our study are given.

4.1.1 Encoder Architecture

- BERT (Bidirectional Encoder Representations from Transformers): Developed by Google, BERT is designed to capture the context of words in search queries. Its architecture enables the model to learn contextualised representations of words by jointly conditioning on both left and right context (Devlin et al. 2019).
- RoBERTa (A Robustly Optimised BERT Approach): Built on top of BERT's architecture, RoBERTa is a variant developed by Facebook AI. RoBERTa introduces several key modifications, including longer training with larger batches, more data, and dynamic masking, which improves its performance and generalisation capabilities (Liu et al. 2019).

4.1.2 Encoder-Decoder Architecture

We selected one prominent encoder-decoder model, Flan-T5. Developed by Google, Flan-T5 combines the strengths of an encoder to understand input data and a decoder to generate relevant outputs. Additionally, Flan-T5 incorporates instruction fine-tuning, which enables the model to improve its performance and generalisation to unseen tasks (Raffel et al. 2020; Chung et al. 2024).

4.1.3 Decoder Architecture

We selected three prominent decoder-based models, all of which are autoregressive models that generate text by predicting the next word in a sequence.

1. **Phi-3:** Developed by Microsoft (Abdin et al. 2024).
2. **LLama 3:** Developed by Meta (Grattafiori et al. 2024; AI@Meta 2024).
3. **Gemma 2:** Developed by Google (Gemma Team 2024).

Additionally, OpenAI's GPT-4 autoregressive model (OpenAI 2024) was used for several purposes in this study: data preprocessing, data augmentation, and as a prompting baseline.

4.2 Model Testing

To evaluate model performance in detecting bias in job descriptions, we used two methods: fine-tuning and prompting. We fine-tuned the smaller Phi3 Mini (3.8B) model as an exception to compare its performance to the larger Phi3 Small (7B), which was tested with prompting only. This approach enabled us to assess the effectiveness of fine-tuning a smaller model versus prompting a larger one of similar architecture.

4.2.1 Fine-Tuning

We applied the Low-Rank Adaptation (LoRA) approach to fine-tune the models, which enabled us to reduce the parameter count of the over billion-parameter models to 4-bit precision using the QLoRA approach (Hu et al. 2022; Dettmers et al. 2023). This allowed us to train the models on a single GPU. QLoRA was applied to all models except BERT and RoBERTa, which were fine-tuned using the standard approach. The decoder models used for fine-tuning were standard base models, except Phi3, which was only available as an instruction-tuned model.

4.2.2 Prompting (In-Context Learning)

We evaluated the instruction-tuned decoder models using four prompting approaches:

- Zero-Shot (pZS): Models were prompted without providing examples and without task-specific training.
- Few-Shot (pFS): Models were provided with a small number of example inputs and corresponding outputs, with the expectation that the model could generalise from these examples when given unseen inputs (Brown et al. 2020).
- Chain-of-Thought (pCoT): Models were guided through a series of reasoning steps, with the expectation that breaking a complex problem into logical steps would enhance the reasoning performance (Wei et al. 2022). We utilised the *Zero-Shot CoT* method (Kojima et al. 2022).
- Self-Consistency (pSC): Multiple diverse outputs were generated for the same prompt, and the final answer was determined by selecting the most consistent response among these outputs (Wang et al. 2023). We applied three iterations of chain-of-thought reasoning with a majority vote for each label.

5. Experimental Setup

To investigate the research questions, we conducted a series of experiments designed to facilitate a comparative analysis, while controlling most of the relevant conditions.

Table 3
Dataset Distribution Before/After Adding Unverified Synthetic Data

Set	Samples (Before)	Samples (After)	Difference
Training	1,439 | 55.01%	3,090 | 72.42%	+1,651
Validation	593 | 22.67%	593 | 13.90%	0
Testing	584 | 22.32%	584 | 13.69%	0
Total	2,616	4,267	+1,651

5.1 Dataset Splitting

The manually verified samples were shuffled. The validation and test sets were organised to include exactly 80 manually verified samples per label, while the training set contained at least 34 manually verified samples per label for the development of n-shot prompts. This resulted in a distribution of 55.01% for training (1,439 samples), 22.67% for validation (593 samples), and 22.32% for testing (584 samples). The training set was then supplemented with 1,651 unverified synthetic samples containing zero or more labels. This resulted in an overall dataset split of 72.42% for training (3,090 samples), 13.90% for validation (593 samples), and 13.69% for testing (584 samples), making up a total of 4,267 samples (see Table 3).

The fine-tuning experiments were trained on the full training split, while the prompting n-shot experiments operated only on the verified samples of the training split. Both groups of experiments were evaluated using the validation split for prompt/parameter tuning and the test split for final results.

5.2 Baseline Models for Comparison

We selected BERT (base-uncased) as our primary baseline and GPT-4o for zero-shot prompting as a secondary baseline.

5.3 Binary Vector Encoding for Multi-Label Classification

Considering that this is a multi-label problem, a label is represented as an 8-dimensional binary vector, where each bit corresponds to the presence or absence of a particular class.

$$\mathbf{y} = [y_0, y_1, y_2, y_3, y_4, y_5, y_6, y_7]$$

where $y_i \in \{0, 1\}$ for $i = 0, 1, 2, \ldots, 7$. Each position y_i corresponds to a specific class:

y_0 : age	y_2 : feminine	y_4 : masculine	y_6 : sexuality
y_1 : disability	y_3 : general	y_5 : racial	y_7 : neutral

Table 4
Parameters, learning rate η, memory, power, time

Model	**Size** (10^9)	η	$\mathbf{GPU}_{\mathbf{GiB}}$	W	$T_{\mathbf{hrs}}$
BERT base	0.11	3×10^{-5}	2.85	114	0.12
BERT large	0.34	3×10^{-5}	6.78	279	0.27
RoBERTa base	0.13	3×10^{-5}	3.13	101	0.13
RoBERTa large	0.36	3×10^{-5}	7.17	312	0.15
Flan T5 XL	2.85	1×10^{-3}	44.50	244	1.47
Phi3 Mini	3.82	1×10^{-4}	7.44	330	1.82
Llama3 8B	8.03	1×10^{-4}	12.61	355	4.43
Gemma2 9B	9.24	1×10^{-4}	15.35	269	4.18

5.4 Fine-Tuning Setup

Fine-tuning for multi-label classification differs across architectures. In encoder-based models (BERT, RoBERTa), a randomly initialised linear layer is added on top of the pooled output as the classification head. The loss function used is binary cross-entropy, which independently evaluates the probability of each label being present. In encoder-decoder models (Flan-T5), the process follows a sequence-to-sequence approach. The input text is tokenised and processed by the encoder, while the labels are converted into a textual sequence (e.g., concatenated label names) and tokenised. The decoder generates token sequences representing the labels. Cross-entropy loss is applied to compare the predicted and actual token sequences of labels. For decoder-based models (LLaMA, Gemma, Phi3), which are autoregressive, multi-label classification requires adding a sequence classification head. A linear layer maps the hidden states to logits, using the hidden state of the final token to generate the pooled logits for classification. Similar to encoder-based models, the loss function used is binary cross-entropy.

As the objective of the study is to compare models rather than optimise hyper-parameters, the approach was to keep the number of hyperparameter changes to a minimum for a more controlled comparison. For all models, the learning rate was adjusted (see Table 4). For the 4-bit precision models, the LoRA rank (r) and scaling factor (α) parameters were adjusted, keeping them equal to maintain the scaling weights at 1.0. LoRA configurations are presented in Table 5. All other hyper-parameters were kept constant across models according to software defaults, except for RoBERTa large, where a warm-up ratio of 0.1 was set due to its initially poor performance.

Overfitting is a common problem with large language models, and regularisation techniques were applied to mitigate it. However, to ensure a fair comparison across all models, customised regularisation techniques for each model were avoided. Instead, a more balanced approach was taken. While some models reduced overfitting through more aggressive dropout and weight decay, others experienced significant performance deterioration. Therefore, dropout rates of 0.1 and a weight decay of 0.001 were chosen as a good balance across models. A batch size of 8 was used, as having a small size can introduce noise in the gradient updates, serving as an additional regulariser while helping with memory constraints with the larger models such as Flan-T5 XL. Training epochs were set to a maximum of 3. As a result, some overfitting is expected but is

Table 5
Low-Rank Adaptation Parameters

Model	$\textbf{Size}_{4bit}(\mathbf{10^9})$	$r = \alpha$	$\theta_{\textbf{LoRA (million)}}$	Layers
Flan T5 XL	1.78	32	70.78 | 2.42%	All Linear
Phi3 Mini	1.97	32	25.24 | 0.67%	All Linear
Llama3 8B	4.12	16	42.04 | 0.56%	All Linear
Gemma2 9B	5.21	32	108.18 | 1.16%	All Linear

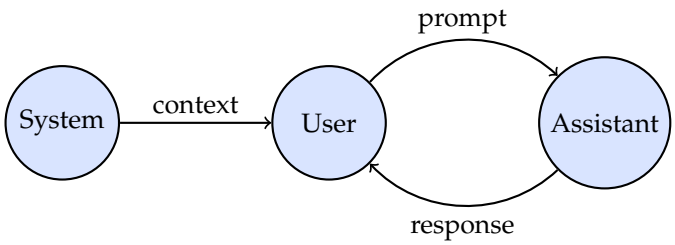

Figure 4
Interaction Between System, User, and Assistant Roles.

acceptable since all models will be tested with the same unseen data and evaluated accordingly.

The experiments used an NVIDIA L40S 48GB GPU, with a container image standardising the software component versions (Appendix C–C4).

5.5 Prompting (In-Context Learning)

In-context learning involves presenting a model with a task description or prompt, along with context, and expecting it to generate a relevant response. This method is well-suited to decoder (autoregressive) models, which can naturally generate coherent, extended text in response to prompts and produce relevant content even for tasks they haven't been specifically fine-tuned for. We did not apply in-context learning to encoder-only or encoder-decoder models, applying prompting solely to decoder models (Llama, Gemma, Phi3).

The interaction with the LLM involves three roles: the **`System`**, which sets the tone and context for the conversation; the **`User`**, who provides input or queries; and the **`Assistant`**, which generates responses to the input. This conversational format is visually represented in Figure 4.

The structure and wording for each prompt setting and for each evaluated model was arrived at through manual iterative experimentation. The structure and components of the *user* prompt are presented in Prompt 1.

The generated text returned in the *assistant* message is parsed using regular expressions to produce the predicted label. Prompting flexibility allows the *neutral* label class to be treated as the special case that it is: it should be true only if all other label classes are false, and false if at least one is true. The prompts are designed to minimise any

PROMPT 1: User Prompt Structure

```
TASK: A short component that describes the task, sometimes used as a
system prompt.
 INSTRUCTION: A longer component that directs the LLM to analyse a sample, lists
the seven bias categories and directs the model towards strictness.
  EVIDENCE: Input text using the format:
Job Posting: \n ${text} \n ===END===
  CLOSING INSTRUCTION: Asks if the ${text} contains implicit bias, specifies the
format, directs a 'no bias' response as the neutral label, and states no further
explanation is needed.
```

response that breaks this logic. While structured generation techniques (Willard and Louf 2023) could have been employed to ensure more rigid control over the outputs, they were not deemed necessary. The outputs were already structured in a consistent and acceptable manner.

The performance of each model is evaluated under four prompt settings: Zero-shot, Few-shot, Chain-of-thought and Self-consistency Chain-of-thought. The structure and wording of each prompt was optimised individually using the *validation* dataset. Final prompt evaluation was done using the *test* dataset.

5.5.1 Zero-shot (pZS)

The LLMs under evaluation are prompted with each sample in the test dataset. The zero-shot prompt for model Meta-Llama-3-8B-Instruct is shown in Prompt 2, with a truncated sample for brevity. For the prompts used, see Appendix B, Prompts 3 and 4.

5.5.2 Few-shot (pFS)

Few-shot prompts use non-synthetic samples and labels from the *train* dataset split. Sequential *user* and *assistant* messages are constructed, ending with a final *user* message containing the actual job description being tested. For Llama-3, the *system* and *instruction* components appear only in the initial *user* message, while subsequent ones contain only *evidence* and *closing instruction*. Phi-3 includes all four components in each *user* message. For the prompts used, see Appendix B, Prompts 5, 6, and 7.

5.5.3 Chain-of-thought (pCoT)

The chain-of-thought technique employed here is a form of zero-shot, as it includes no exemplars in the prompt. The prompt includes reasoning steps to guide the model in checking each bias type, with a minimal prompt found to be most effective for this purpose. Extra clarification is added only for the *general* category. For the prompts used, see Appendix B, Prompts 8, 9, and 10.

5.5.4 Self-consistency (pSC)

The previous three prompting strategies used greedy search decoding, which is deterministic. Self-consistency prompting uses sampling with temperature and nucleus sampling (*top_p*) chosen through grid search for each model (Table 6). Temperature controls randomness in text generation, while nucleus sampling selects from a distribution based on cumulative probability. The same prompts used for chain-of-thought prompting are applied here. Three outputs are generated, each converted to an 8-dimensional binary vector. The final output is determined by a position-wise majority vote.

PROMPT 2: Zero Shot (pZS) - Meta-Llama-3-8B-Instruct

SYSTEM You are an analyst of job postings. Your task is to detect implicit bias, if present.

USER **Analyse the job posting for implicit bias. The bias categories are age, . . . Return the appropriate label only when absolutely sure that the particular bias category is present in the text.**
Job Posting:
Role: Graduate Trainee Recruitment Consultant. Location: Bristol. Package: £20-24K Basic Salary . . .
===END===
Does the job posting contain any implicit bias? Please respond in the format of 'Labels: <labels>' where the possible labels are age, . . . If no bias is detected please return 'Labels: neutral'. No further explanation is required.

ASSISTANT Labels: general

Table 6
Self-Consistency Parameters

Parameter	Gemma2-9B	Llama3-8B	Phi3-7B-8k
Temperature	0.70	0.15	0.20
top_p	1.00	0.90	0.80

5.6 Label Extraction

Testing of the chosen models entails extraction of a *predicted label* from the model response. The label extraction protocol differs based on whether the model is used for in-context learning or fine-tuning.

5.6.1 Prompting (In-Context Learning)

For in-context learning experiments (Llama, Gemma, Phi3), words representing the bias terms or *neutral* are extracted from each prompt response. In these experiments, the models rely on in-context learning to generate responses. As the responses are short - maximum nine words - and very consistent in quality, we used regular expressions to extract the bias terms (Appendix E–D). To construct the label, each position in the 8-bit label is set to 0 or 1, depending on whether each bias type is absent or present in the response.

5.6.2 Fine-tuning

For encoder (BERT, RoBERTa) and decoder (Llama, Gemma, Phi3) models, the sigmoid function and binarisation of its output were used to create the predicted labels. Logit outputs were converted into probabilities, with a threshold of 0.5 applied for binarisation.

$$\hat{y} = \begin{cases} 1, & \text{if } \frac{1}{1+e^{-x}} \geq 0.5 \\ 0, & \text{otherwise} \end{cases}$$

For the T5-Flan model, due to its sequence-to-sequence nature, the presence of each label was checked directly against the generated output sequence.

6. Results & Discussion

6.1 Evaluation Metrics

To assess the performance of the models, we considered using K-fold validation for the encoder models, which would have provided a more comprehensive evaluation. However, due to the lengthy training times required for the multi-billion parameter decoder models, we decided instead to use a straightforward hold-out validation method. The hold-out validation method involves splitting the data into training and validation sets, with the validation set used to evaluate the models' performance. This method was applied across all experiments to ensure consistency in the evaluation process, allowing for a direct comparison of the models' performance and their generalisation abilities.
To measure the performance of our experiments, we used precision, recall, and F-score (Pedregosa et al. 2011). Due to the multi-label nature of the problem, we used sample-wise averaging whereby these metrics are calculated for each sample and then averaged across all samples. A fourth metric, exact match ratio, is also applied.

Precision. The metric computes the proportion of correctly predicted true positive samples to the total predicted positives.

$$\frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_j + FP_i}$$

Recall. The proportion of correctly predicted true positives to all the positives in the category.

$$\frac{1}{N}\sum_{i=1}^{N}\frac{TP_i}{TP_i + FN_i}$$

F_β***-Score***. This metric balances the precision and recall scores. As we want recall and precision to be equally important, we used $\beta = 1$.

$$\frac{1}{N}\sum_{i=1}^{N}\frac{(1+\beta^2)\cdot \text{Precision}_i \cdot \text{Recall}_i}{\beta^2 \cdot \text{Precision}_i + \text{Recall}_i}$$

Exact Match Ratio (EMR). This metric computes the proportion of samples that have all their predicted labels exactly matching the true labels.

$$\frac{1}{N}\sum_{i=1}^{N}\mathbb{I}(\hat{y}_i = y_i)$$

Table 7
Fine-Tuning vs Prompting Performance.

Model	Type	F_1	Precision	Recall	EMR
BASELINES					
BERT base uncased	FT	0.67	0.67	0.69	0.62
GPT-4o-2024-05-13	pZS	0.59	0.61	0.59	0.56
BERT large uncased	FT	0.70	0.70	0.72	0.67
RoBERTa base	FT	0.71	0.71	0.73	0.66
RoBERTa large	FT	0.64	0.63	0.65	0.59
Flan T5 XL	FT	**0.74**	**0.74**	**0.75**	**0.70**
Gemma2-9B	FT	0.72	0.71	0.74	0.64
	pZS	0.56	0.55	0.60	0.47
	pFS	0.55	0.52	0.60	0.42
	pCoT	0.56	0.54	0.62	0.44
	pSC	0.56	0.54	0.62	0.44
Llama3-8B	FT	0.71	0.70	0.74	0.65
	pZS	0.46	0.46	0.46	0.44
	pFS	0.54	0.54	0.61	0.41
	pCoT	0.48	0.48	0.48	0.45
	pSC	0.48	0.48	0.48	0.45
Phi3 3.8B 4k	FT	0.67	0.66	0.69	0.61
Phi3-7B-8k	pZS	0.56	0.55	0.59	0.47
	pFS	0.46	0.45	0.49	0.40
	pCoT	0.58	0.57	0.63	0.47
	pSC	0.59	0.57	0.63	0.47

Bold boxes mark the highest value(s) per metric; lighter boxes mark the lowest.

6.2 Results and Analysis

The comparative performance between fine-tuning (*FT*) and prompting experiments (*PT*) demonstrates that fine-tuning consistently outperforms prompting in the task of multi-label job description bias detection. Fine-tuned models achieve higher and more reliable F_1, precision, recall, and exact match ratio, as shown by their consistent clustering towards higher precision and recall values. This suggests that fine-tuning is a more effective approach for detecting bias in job descriptions, as it allows the model to learn specific patterns and relationships in the data that are relevant to the task.

Table 7 and Figure 5 present the overall model performance, while Figure 6 compares the category-wise performance of the leading models against baseline models. Additionally, detailed F_1 results are provided in Table 8, with precision and recall results shown in Table 9.

6.2.1 Encoder-Decoder vs Decoder Dichotomy

Interestingly, as shown in (Table 7, Fig. 6), Flan T5 XL, an encoder-decoder model, outperforms the decoder-only models in our experiments. We note that it is more

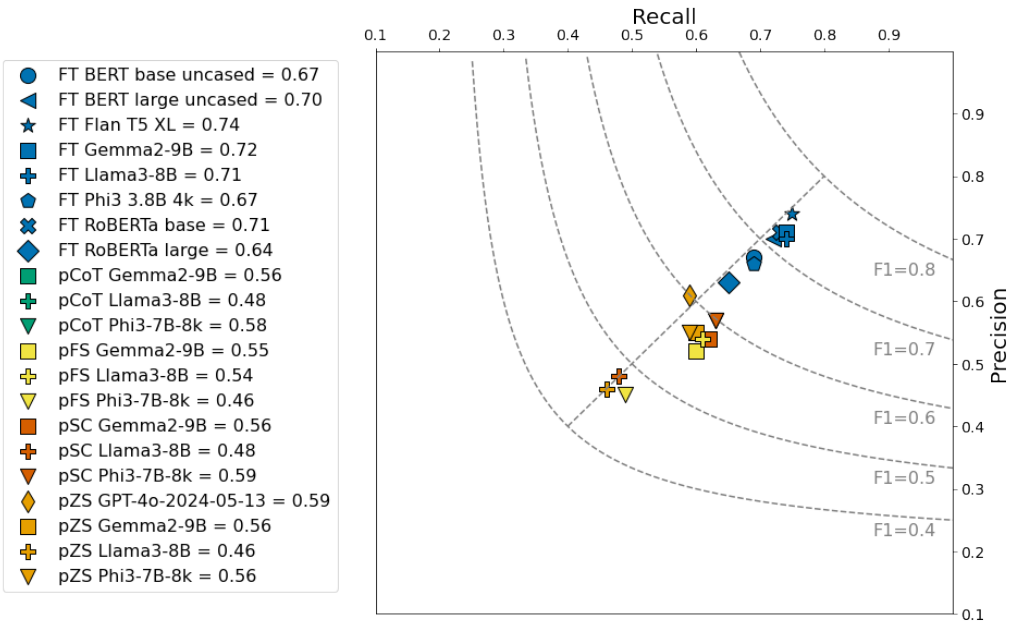

Figure 5
Model Performance: Precision vs Recall

computationally expensive to train than the encoders (Table 4). However, it has fewer parameters and requires less compute compared to most decoder models employed in our study.

This suggests that the encoder-decoder architecture may be better suited for multi-label job description bias detection tasks. One possible hypothesis is that the encoder-decoder architecture allows for a more nuanced representation of the input text, which is beneficial for detecting subtle biases. It benefits from leveraging the strengths of both encoder and decoder architectures. The encoder allows for a more nuanced representation of the input text, while the decoder enables the model to generate more accurate and informative outputs. Additionally, the encoder-decoder architecture may be able to capture longer-range dependencies in the text, which is important for understanding the context and nuances of job descriptions.

Gemma2 9B *FT* is the second-best performer, and appears to be a good trade-off in terms of compute and training time. This suggests that Gemma2-9B-FT is a good choice for applications where accuracy is critical, but computational resources are not a concern. However, as our results show, Gemma2-9B-FT is still less competitive to Flan-T5 XL with approximately 2.85B parameters, which suggests that the encoder-decoder architecture may be more effective for this task. One possible hypothesis is that the decoder-only architecture may be more prone to overfitting, particularly when dealing with complex and nuanced tasks like multi-label job description bias detection.

6.2.2 Performance Comparison and Occlusion Analysis of Encoder Models

The BERT large model shows improved performance compared to the baseline (BERT base) but still falls short of both RoBERTa base and Llama3 *FT*. Although RoBERTa large initially underperformed, its results improved with the introduction of a warmup ratio,

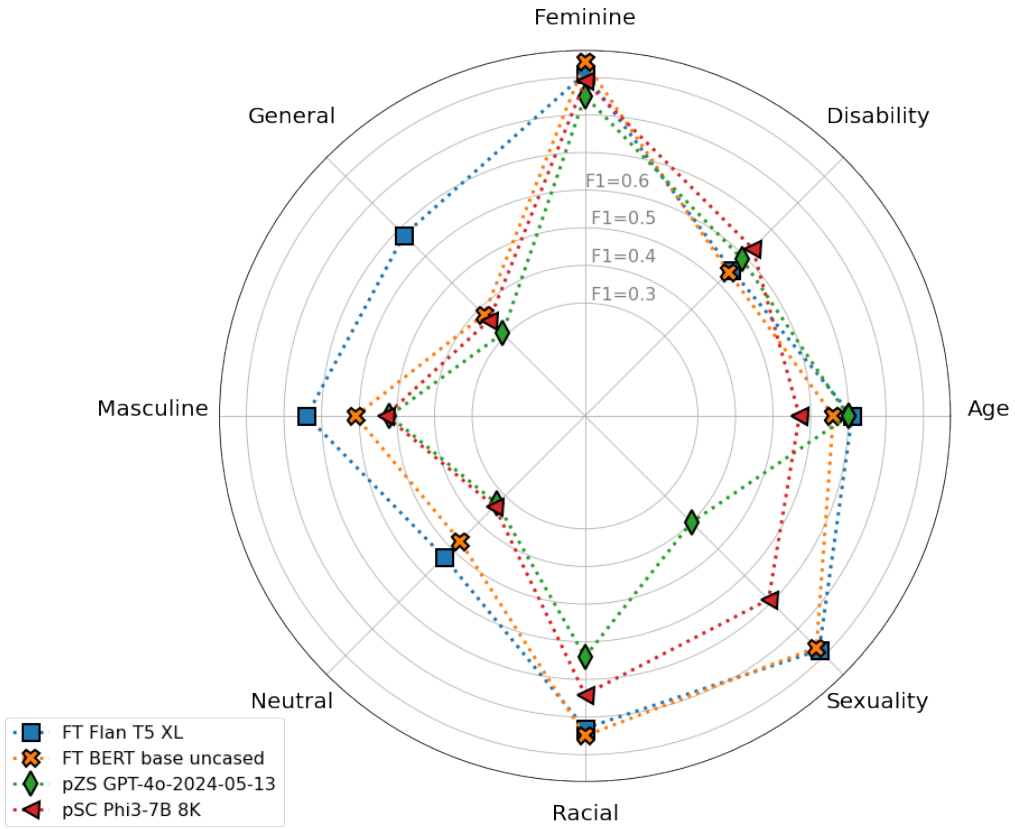

Figure 6
Top Performers' Comparison Against Baseline Models

outperforming the *PT* experiments. However, it remained the lowest performer among the *FT* experiments. This suggests that the BERT large model is a good choice for multi-label job description classification tasks where accuracy is important, but computational resources are limited. The results also suggest that the performance of the BERT large model can be improved with careful tuning of hyperparameters, such as the warmup ratio.

Analysis of RoBERTa. A notable drop in performance, highlighted in Table 8, is observed for the *general* label. The base model achieves an F_1 score of 0.62, whereas the large model scores only 0.26. To further investigate these performance differences, we present examples where the models' predictions diverge from the ground truth labels. These samples were selected based on notable discrepancies in confidence scores (probabilities) between the base and large RoBERTa models, with predicted labels assigned when confidence scores are equal to or greater than 0.5.

We employed a token-level ablation method, specifically an *occlusion analysis*, to identify influential phrases contributing to the models' predictions. This technique involves systematically masking each word or phrase in the input text with a neutral placeholder token (e.g., 00000) and recalculating the model's confidence scores. By

Table 8
Category-wise Performance: F_1

Model	Type	Age	Dis.	Fem.	Gen.	Mas.	Neu.	Rac.	Sex.
BERT base	FT	0.66	0.54	0.94	0.38	0.61	0.47	0.85	0.87
GPT-4o	pZS	0.70	0.59	0.85	0.31	0.52	0.33	0.64	0.40
BERT large	FT	0.63	0.56	0.94	0.60	0.67	0.47	0.87	0.85
RoBERTa base	FT	0.63	0.63	0.93	0.62	0.67	0.50	0.82	0.92
RoBERTa large	FT	0.61	0.60	0.96	0.26	0.64	0.48	0.85	0.86
Flan T5 XL	FT	0.71	0.55	0.91	0.68	0.74	0.53	0.83	0.88
Gemma2-9B	FT	0.65	0.66	0.95	0.60	0.66	0.49	0.86	0.85
	pZS	0.58	0.49	0.88	0.44	0.43	0.35	0.70	0.57
	pFS	0.54	0.38	0.81	0.36	0.51	0.27	0.67	0.62
	pCOT	0.56	0.53	0.83	0.37	0.50	0.39	0.69	0.57
	pSC	0.57	0.51	0.83	0.38	0.52	0.41	0.69	0.57
Llama3-8B	FT	0.63	0.64	0.92	0.67	0.63	0.50	0.84	0.84
	pZS	0.33	0.42	0.84	0.36	0.31	0.34	0.66	0.28
	pFS	0.56	0.63	0.69	0.31	0.42	0.40	0.73	0.55
	pCOT	0.45	0.50	0.82	0.33	0.35	0.35	0.71	0.22
	pSC	0.44	0.52	0.82	0.33	0.33	0.35	0.71	0.22
Phi3 3.8B 4k	FT	0.55	0.60	0.93	0.57	0.61	0.44	0.85	0.84
Phi3-7B 8k	pZS	0.64	0.47	0.87	0.27	0.50	0.36	0.73	0.64
	pFS	0.28	0.31	0.83	0.24	0.43	0.36	0.62	0.56
	pCOT	0.56	0.62	0.89	0.35	0.53	0.33	0.72	0.69
	pSC	0.57	0.63	0.89	0.36	0.53	0.34	0.74	0.69

Bold boxes mark the highest value(s) per metric; lighter boxes mark the lowest.

observing changes in the output scores resulting from the omission of specific tokens, we can pinpoint which words have the most significant impact on the model's decisions.

In the examples below, certain phrases are removed (shown as crossed out) and sometimes replaced with text (shown in bold) to indicate a modification that would substantially lower the confidence score of the occlusion target label. This method highlights the models' sensitivity to particular linguistic cues.

- **Truth:** general; **Occlusion Target:** general; **Confidence:** base=0.97, large=0.02
  *. . . Job Type: Chauffeur . . . Qualifications and Experience: Prior experience as a chauffeur or in a related field is preferred. ~~Must possess a valid driver's license with a clean driving record~~**Must hold a valid driver's licence with a safe and responsible driving record**. A high school diploma or equivalent is required; ~~degree from a reputable institution is desirable~~. Ability to pass a thorough background check is mandatory. Perks and Benefits: We are committed to fair and equitable pay practices. The salary for this position ranges from $45,000 to $60,000 USD per year, based on your experience and skills . . .*
- **Truth:** feminine; **Occlusion Target:** feminine; **Confidence:** base=0.85, large=0.09
  *. . . Two years ago, there were only 2 ~~girls~~**employees** in our HR department. Now we have 4 and we are looking for the 5th person, that is how fast we grow **our team of boys and girls**! . . .*

Table 9
Category-wise Performance: Precision and Recall

Model	Type	Age	Dis.	Fem.	Gen.	Mas.	Neu.	Rac.	Sex.
BERT base	FT	84¦54	79¦41	95¦94	90¦24	72¦54	32¦86	89¦81	94¦81
GPT-4o	pZS	75¦66	62¦56	87¦82	57¦21	74¦40	21¦71	95¦49	100¦25
BERT large	FT	87¦50	91¦40	95¦93	88¦45	82¦56	32¦88	87¦86	97¦76
RoBERTa base	FT	80¦51	87¦50	93¦94	75¦53	78¦59	38¦72	83¦81	96¦89
RoBERTa large	FT	86¦47	88¦45	97¦94	65¦16	76¦55	33¦89	87¦84	97¦78
Flan T5 XL	FT	89¦59	89¦40	92¦90	79¦60	83¦68	37¦93	89¦79	96¦81
Gemma2-9B	FT	72¦60	95¦50	99¦91	84¦46	65¦66	34¦86	90¦82	97¦75
	pZS	72¦49	74¦36	80¦97	35¦59	29¦86	48¦28	81¦62	97¦40
	pFS	61¦49	68¦26	70¦95	25¦68	36¦86	50¦19	61¦74	79¦51
	pCOT	66¦49	82¦39	73¦96	26¦69	35¦90	56¦30	78¦62	97¦40
	pSC	67¦49	81¦38	73¦96	26¦69	36¦91	60¦31	78¦62	97¦40
Llama3-8B	FT	91¦49	97¦47	99¦86	82¦56	62¦65	35¦90	89¦80	95¦75
	pZS	74¦21	64¦31	85¦84	30¦46	94¦19	22¦70	89¦53	93¦16
	pFS	65¦50	60¦66	55¦95	22¦54	73¦30	31¦56	76¦70	97¦39
	pCOT	78¦31	68¦40	79¦85	37¦30	75¦23	23¦81	89¦59	100¦12
	pSC	76¦31	70¦41	79¦85	37¦30	74¦21	23¦81	90¦59	100¦12
Phi3 3.8B 4k	FT	89¦40	97¦44	99¦89	65¦51	95¦45	30¦90	93¦79	95¦75
Phi3-7B 8k	pZS	62¦66	72¦35	81¦95	23¦35	54¦47	29¦47	91¦61	70¦60
	pFS	93¦16	88¦19	73¦96	20¦31	71¦31	23¦81	95¦46	89¦41
	pCOT	47¦70	66¦59	84¦95	28¦49	58¦49	30¦35	86¦62	79¦61
	pSC	47¦71	67¦59	84¦95	28¦50	58¦49	32¦36	89¦62	79¦61

Note: [Precision¦Recall], values are shown without leading zeros or decimals for readability.

- **Truth:** age; **Occlusion Target:** feminine; **Confidence:** base=0.92, large=0.26
  *Job title: Graduate Sales Manager . . . this role offers a recent graduate the opportunity to gain real-life experience from very early on, all with the support of ~~a nurturing~~**an encouraging** office environment. . . .*
- **Truth:** neutral; **Occlusion Target:** feminine; **Confidence:** base=0.05, large=0.60
  . . . is a thriving community hospital that proudly provides acute care services . . . As a key member of the Women & Children's Health team, the Obstetrics Registered Practical Nurse is responsible for assessing, analyzing, prioritizing, planning, and evaluating care in collaboration with ~~women and~~ families in both normal and complex situations. The nursing care includes antenatal care, postpartum care, breastfeeding support, infant care, and care following perinatal loss. . . . Breastfeeding Certificate Program required or willingness to complete within 1 year. . . . and emotional needs of the post-partum women and neonates required . . .

By analysing these examples, we observe that the base RoBERTa model generally assigns higher confidence to the correct labels than the large RoBERTa model. However, this is not always the case. In the *Graduate Sales Manager* example, the base model incorrectly assigns a high confidence score (0.92) to the *feminine* label, despite the ground truth being *age*. This misclassification is influenced by the word *nurturing*, which the base model strongly associates with the *feminine* label.

In the *Chauffeur* example, the only synthetic example presented, the large model did not recognise the potential ambiguity in the term *clean driving record*, which could

unintentionally exclude candidates with minor infractions that do not reflect their actual driving skills or safety and are still suitable for the role. Additionally, the term *reputable institution* may suggest prestige, creating unnecessary ambiguity or barriers since a high school diploma or equivalent is already requested. The requirement of a *thorough background check* could also introduce ambiguity, but the occlusion analysis showed that neither model recognised this issue.

In the *HR department* example, the occlusion analysis showed that the word *girls* significantly contributed to the *feminine* label. When *girls* was replaced with *employees*, the base model no longer predicted the *feminine* label. However, adding the phrase *our team of boys and girls* did not reintroduce the *feminine* classification, suggesting that it is not simply the presence of *girls* but its contextual use that influences the classification. This indicates that the base model is sensitive to the contextual application of gender-specific terms rather than just their occurrence. In contrast, the large model did not detect the cue that the job description is specifically seeking a fifth *girl*.

In the *community hospital* example, the large model misclassified the sample as *feminine*, and the occlusion analysis identified the word *women* as a significant contributing factor. By eliminating the word, the large model's confidence in the *feminine* label was substantially reduced and no longer classified it as such. However, despite the presence of several words associated with females—such as *Women's and Children's Health*, *breast-feeding*, and *postpartum*—these terms are contextually appropriate and do not represent feminine bias in the text. This suggests that the large model may be overly sensitive to some gender-specific terms, misclassifying neutral or appropriately gendered content as biased.

The occlusion analysis shows that specific words or phrases can significantly impact both models' predictions and performance on particular labels. These findings suggest that larger models do not inherently guarantee improved performance, and smaller models may misclassify due to influential linguistic cues. Thus, careful hyperparameter tuning and a deeper understanding of how language cues affect predictions remain important. Further investigation into these influences may improve the performance of language models in multi-label classification for job descriptions.

6.2.3 Prompting (In-Context Learning) Experiments

Prompting experiments show significant variability in performance across different prompting methods and categories. GPT-4o (pZS), Phi3-7B-8k (pZS, pCoT), and Gemma2-9B (pFS, pCoT, pSC) showed moderate precision and recall. GPT-4o (pZS) was the best *PT* experiment in terms of the exact match ratio, despite its poor performance in detecting *sexuality* bias compared to Phi3-7B-8k (pSC), the second leading *PT* experiment by F_1 score (Fig. 6). Gemma2-9B *PT* performed competitively to Phi3-7B-8k (pSC) while the Llama3-8B models (pSC, pZS) and Phi3-7B-8k (pFS) performed poorly compared to the other models in our experiments. This suggests that the choice of prompting method and category can have a significant impact on the performance of the model.

Category-wise Performance. All experiments, whether fine-tuned or prompted, performed well on the *feminine* category, indicating strength in this area. Fine-tuned experiments performed well on *racial* and *sexuality*. Other models in our experiments showed broader score distributions across categories, with *sexuality* showing wide variability, followed by *age*. Interestingly, most models in our experiments struggled with *neutral*,

disability, and *general* to some degree. This suggests that the models are able to detect bias in certain categories, but may struggle with others.

Challenges in Interpreting Results. The *general* category is particularly challenging to interpret due to its subjective nature; for instance, the absence of a diversity statement or salary transparency can be perceived as *general* bias. Interestingly, *age* and *disability* did not perform as well, despite being majority classes during manual annotation of real job descriptions. We believe this suggests that the models may be biased towards certain categories and that the performance of the models can be improved with careful tuning of hyperparameters and the use of more diverse training data.

7. Conclusions

This research investigated the application of LLMs for the detection of implicit bias in job descriptions. A dataset was constructed for the task, consisting of a mix of gold, silver, and bronze-standard labelled data, with the validation and test sets comprising entirely gold and silver-standard labelled data. This publicly available dataset provides a valuable resource for future researchers in related fields.

Different model architectures were fine-tuned and evaluated for their ability to classify text as containing one or more of seven bias types. Three decoder-only models were tested under four distinct prompt settings and compared against a zero-shot GPT-4o baseline. The *feminine* category consistently achieved high F_1 scores across all experiments, indicating strong performance in this area. Fine-tuned models consistently outperformed non-fine-tuned ones, with Flan-T5-XL emerging as the top performer. Despite its smaller size (2.85B parameters) compared to the larger decoder models (including GPT-4o), the fine-tuned Flan-T5-XL demonstrated notable performance. This suggests that targeted fine-tuning with encoder-decoder architectures can result in more efficient models with lower energy and computational costs when detecting implicit bias in job descriptions.

We restricted our research to open-source models with less than 10B parameters. Future research could investigate larger models to determine if the increased scale enhances their ability to identify the nuanced language of implicit bias. Another future area of research would be in the instruction-tuning of the chosen LLMs on a curated job description dataset.

Acknowledgments

This research is co-funded by the European Regional Development Fund through the ADAPT Centre for Digital Content Technology grant number 13/RC/2106_P2. In addition, Kolawole Adebayo is supported by Enterprise Ireland's CareerFit-Plus Co-fund and the European Union's Horizon 2020 research and innovation programme Marie Skłodowska-Curie Grant No. 847402. Tristan Everitt's contribution was primarily supported by Armac Systems, and Paul Ryan's by eSpatial Solutions, with both part-funded by Technology Ireland ICT Skillnet.

Appendix

Appendices, test results, regular expressions and source code are available at:

- `https://github.com/2024-mcm-everitt-ryan`.

Datasets and the model repository can be accessed at:

- `https://huggingface.co/2024-mcm-everitt-ryan`.

The software developed for data preparation and augmentation has been contributed as plugins to the Hop Orchestration Platform (HOP), a project under the Apache Software Foundation that enables the visual design of data processing workflows. The code is released under the open-source Apache License, Version 2.0, making it accessible to a broader community beyond this work.

- `https://hop.apache.org`.
- `https://github.com/apache/hop`.

References

Abdin, Marah, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, and et al. 2024. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*.

AI@Meta. 2024. Llama 3 model card. https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md, April.

Brown, Tom, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D. Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and et al. 2020. Language models are few-shot learners. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901, Vancouver, BC, Canada, December. Curran Associates, Inc.

Bruce, Stephen. 2009. Non-prejudicial language for ADA job descriptions. *HR Daily Advisor*, March.

Burn, Ian, Daniel Firoozi, Daniel Ladd, and David Neumark. 2022. Help really wanted? the impact of age stereotypes in job ads on applications from older workers. Working Paper 30287, National Bureau of Economic Research, July.

Chung, Hyung Won, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and et al. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Cunningham, George B. and Harper R. Cunningham. 2022. Bias among managers: Its prevalence across a decade and comparison across occupations. *Frontiers in Psychology*, 13, November.

Dettmers, Tim, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient finetuning of quantized LLMs. In A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine, editors, *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115, New Orleans, LA, USA, December. Curran Associates, Inc.

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, USA, June. Association for Computational Linguistics.

Douze, Matthijs, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvasy, Pierre-Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. 2024. The Faiss library. *arXiv preprint arXiv:2401.08281*.

Fatfouta, Ramzi. 2023. What do they really want? effects of the wording of job advertisements on narcissists' perceptions of organizational attraction. *Current Psychology*, 42(1):154–164, January.

Fiske, Susan T. and Tiane L. Lee. 2008. Stereotypes and prejudice create workplace discrimination. In Arthur P. Brief, editor, *Diversity at Work*, Cambridge Companions to Management. Cambridge University Press, April, page 13–52.

Fridell, Lorie A. 2017. Introduction. In *Producing Bias-Free Policing: A Science-Based Approach*. Springer International Publishing, Cham, pages 1–5.

Frissen, Richard, Kolawole John Adebayo, and Rohan Nanda. 2023. A machine learning approach to recognize bias and discrimination in job advertisements. *AI & SOCIETY*, 38(2):1025–1038, April.

Gaucher, Danielle, Justin Friesen, and Aaron C. Kay. 2011. Evidence that gendered wording in job advertisements exists and sustains gender inequality. *Journal of personality and social psychology*, 101(1):109, July.

Gemma Team. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, July.
Gilardi, Fabrizio, Meysam Alizadeh, and Maël Kubli. 2023. ChatGPT outperforms crowd workers for text-annotation tasks. *Proceedings of the National Academy of Sciences*, 120(30), July.
Gill, Kieran. 2020. BBC bans football commentators from saying "racist" phrases such as cakewalk, nitty gritty, sold down the river and uppity as part of racial bias training. *Daily Mail*, September. Accessed: 21-Jul-2024.
Grattafiori, Aaron, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and et al. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, July.
Hedley, Jonathan. 2023. jsoup: Java html parser. https://github.com/jhy/jsoup, December. Release v1.17.2 (9dec1ba).
Horvath, Lisa Kristina and Sabine Sczesny. 2016. Reducing women's lack of fit with leadership positions? effects of the wording of job advertisements. *European Journal of Work and Organizational Psychology*, 25(2):316–328.
Hu, Edward J., Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, Virtual Conference, April.
Hube, Christoph and Besnik Fetahu. 2018. Detecting biased statements in wikipedia. In *WWW'18: Companion Proceedings of the The Web Conference 2018*, page 1779–1786, Lyon, France, April. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, Switzerland.
Hunt, Vivian, Dennis Layton, and Sara Prince. 2015. Why diversity matters. *McKinsey & Co.*, January.
Kojima, Takeshi, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213, New Orleans, LA, USA, November. Curran Associates, Inc.
Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *arXiv preprint arXiv:1907.11692*, July.
Mao, Ruochen, Liming Tan, and Rezza Moieni. 2023. Developing a large-scale language model to unveil and alleviate gender and age biases in australian job ads. In *2023 IEEE International Conference on Big Data (BigData)*, pages 4176–4185, Sorrento, Italy, December.
OFCCP. 2024. Best practices and resources. *U.S. Department of Labor, Office of Federal Contract Compliance Programs*, April.
Ongig Team. 2024. Diversity and inclusion blog. *Ongig*, July. https://blog.ongig.com/diversity-and-inclusion. Accessed: 21-Jul-2024.
OpenAI, 2023. *GPT-4-Turbo*. Language model developed by OpenAI.
OpenAI, 2024. *GPT-4o*. Optimised version of GPT-4 developed by OpenAI.
Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, and et al. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830.
Pillar, Anna, Kyrill Poelmans, and Martha Larson. 2022. Regex in a time of deep learning: The role of an old technology in age discrimination detection in job advertisements. In Bharathi Raja Chakravarthi, B. Bharathi, John P. McCrae, Manel Zarrouk, Kalika Bali, and Paul Buitelaar, editors, *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*, pages 13–18, Dublin, Ireland, May. Association for Computational Linguistics.
Radford, Alec, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI*, February.
Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.
Rebedea, Traian, Razvan Dinu, Makesh Narsimhan Sreedhar, Christopher Parisien, and Jonathan Cohen. 2023. NeMo guardrails: A toolkit for controllable and safe LLM applications with programmable rails. In Yansong Feng and Els Lefever, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages

431–445, Singapore, December. Association for Computational Linguistics.
Recasens, Marta, Cristian Danescu-Niculescu-Mizil, and Dan Jurafsky. 2013. Linguistic models for analyzing and detecting biased language. In Hinrich Schuetze, Pascale Fung, and Massimo Poesio, editors, *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1650–1659, Sofia, Bulgaria, August. Association for Computational Linguistics.
Reimers, Nils and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November. Association for Computational Linguistics.
Stahl, Peter. 2023. Lingua. https://github.com/pemistahl/lingua, August. Release v1.2.2 (47d5287).
Storm, Kai Inga Liehr, Lea Katharina Reiss, Elisabeth Anna Guenther, Maria Clar-Novak, and Sara Louise Muhr. 2023. Unconscious bias in the HRM literature: Towards a critical-reflexive approach. *Human Resource Management Review*, 33(3):100969, September.
Techmap.io. 2020–2023. Public datasets on Kaggle. https://www.kaggle.com/techmap/datasets. Accessed: 2024-07-11.
Tkachenko, Maxim, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. 2020-2022. Label Studio: Data labeling software. Open source software available from https://github.com/heartexlabs/label-studio.
Wang, Xuezhi, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*, Kigali, Rwanda, May.
Wei, Jason, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837, New Orleans, LA, USA, December. Curran Associates, Inc.
Willard, Brandon T. and Rémi Louf. 2023. Efficient guided generation for LLMs. *arXiv preprint arXiv:2307.09702*, July.