Università di Brescia

Sapienza Università di Roma

University of Brescia
Department of Information Engineering

Sapienza University of Rome
Department of Computer, Control
and Management Engineering

National PhD Program in Artificial Intelligence

Academic Year 2024–2025 (XXXVIII cycle)

Analysis and Detection of Social Biases in Deep Neural Language Models

Supervisor:
Prof. Alfonso Emilio Gerevini

Candidate:
Michele Dusi
Matr. 2075001

Co-supervisors:
Prof. Ivan Serina
Prof. Luca Putelli

Abstract

The rising adoption of deep Neural Language Models (NLMs) has caused concerns about the presence of social biases and their potential societal impact. While a substantial body of work has documented biased behaviors in model outputs, the question of where such biases originate within the language modeling pipeline remains only partially understood.

This thesis introduces a method for detecting and quantifying social biases in pre-trained language models that aims to balance interpretability and statistical rigor. The proposed approach tests whether the information encoded in embedded representations of protected attributes (e.g., gender, nationality, religion) can be used to predict stereotyped attributes through a simple supervised classification task. Requiring only a minimal labeled dataset, this method provides an accessible way to probe representational biases. Experimental results on several Transformer-based models reveal consistent associations between protected and stereotyped properties. In addition, a complementary visualization-based technique is introduced to support qualitative inspection of bias patterns.

Building on this methodological framework, the thesis also conducts a systematic analysis of recent literature on bias in language models, with the aim of exploring the context and mechanisms through which the phenomenon manifests. A central motivation is to move beyond the common assumption that bias is solely a reflection of training data; therefore, the resulting analysis is organized around four complementary perspectives. First, it examines the role of training and fine-tuning corpora in shaping measurable bias. Second, it analyzes how bias can

emerge, evolve, or be amplified during the training process itself. Third, it considers model-internal factors, investigating how biases are encoded and propagated through parameters, representations, and architectural components. Finally, it discusses evidence on model scale and complexity, assessing whether certain forms of bias appear or intensify only beyond specific thresholds of capacity or model size.

Overall, this thesis provides a structured synthesis of current knowledge on bias origins in language models, while offering a practical tool for their empirical assessment. The ultimate goal is to provide actionable insights that support the development of fairer and more inclusive NLP technologies.

Keywords: NLP Bias, AI Fairness, Neural Language Models, Stereotype Detection, AI Ethics.

Table of Contents

Chapter 1

Introduction and Research Vision

Over the past decade, deep Neural Language Models (NLMs) have evolved from experimental research artifacts to foundational components of contemporary digital infrastructure. Transformer-based architectures, trained on large corpora and optimized through increasingly sophisticated training procedures, now carry the weight of search engines, conversational agents, content generation systems, recommendation platforms, and decision-support tools. As a result, language models no longer operate solely within research environments. Instead, they shape access to information and mediate communication, influencing the social, economic, and institutional contexts.

This expanded role has intensified inspection of their behavior. Among the most pressing concerns is the presence of social biases, namely, systematic associations or disparities related to protected or socially relevant attributes such as gender, ethnicity, religion, or nationality. The question is no longer *whether* language models can exhibit biased behavior, which has been well-documented across various studies (Blodgett et al., 2020; Garrido-Muñoz et al., 2021; Ghosh & Wilson, 2025; Meade et al., 2022; Mehrabi et al., 2021; Sheng et al., 2021), but *how* such biases arise, how they are represented internally, and under what conditions they manifest.

1.1. Problem Statement and Motivation

In a nutshell, language models learn distributional patterns from large text corpora. When these corpora contain historical inequalities, stereotypical associations, or asymmetric representations of social groups, models may internalize

these patterns as statistically salient features. In fact, unfair behaviors have been documented in NLMs by a wide range of studies, and most of the scientific community converges on the idea that these biases primarily originate from the training data (Caliskan et al., 2017; Garrido-Muñoz et al., 2021; Gehman et al., 2020; Guo & Caliskan, 2021; Lu et al., 2020; Sheng et al., 2021; B. H. Zhang et al., 2018; Zhao et al., 2018a).

However, more recent research reveals that there may be additional mechanisms at play. Bias may emerge through complex interactions between data distributions (Gupta et al., 2022; Subramanian et al., 2021; Valentini et al., 2022), optimization objectives (He et al., 2022; B. Zhang et al., 2018), architectural constraints (Adiga et al., 2025; Gaci et al., 2022), representation learning dynamics (Feng et al., 2023; Gonçalves & Strubell, 2023), and scaling effects (Y. Chen et al., 2025; Zhao et al., 2025; Zhou et al., 2023). In this sense, the phenomenon is better characterized as an emergent property of the entire language modeling pipeline, rather than as a direct effect of individual data points. It can be observed at multiple levels: in output probabilities, in contextual generation behavior, in latent representations, and in the geometry of embedding spaces. Moreover, biases may evolve during pre-training and fine-tuning, or interact with model capacity and scale in non-trivial ways.

While the data-centric perspective has provided valuable insights, it oversimplifies the problem. In fact, by attributing bias predominantly to distorted data distributions, the field may overlook other mechanisms through which bias can emerge. For instance, the optimization process may amplify certain associations that are weakly present in the data, or architectural features may predispose models to encode specific types of correlations.

This thesis is motivated by the need to move beyond a strictly data-centric account of bias. We adopt a broader analytical perspective, systematically examining empirical studies and theoretical analyses. Through such perspective, we seek to contribute to a more integrated understanding of where bias originates and how it persists in neural language models. At the same time, we aim to

develop and apply novel methodologies for detecting and quantifying the phenomenon in model representations, in order to provide tools for further research and mitigation efforts.

1.2. Main Contributions

The main contributions of this thesis to the study of social bias in NLMs span theoretical, methodological, and empirical dimensions.

First, the thesis synthesizes empirical findings through a multi-level framework examining data, training dynamics, architecture, representation learning, and scaling, to provide a more comprehensive account of bias formation and persistence.

Building on this broader perspective, we propose a supervised probing methodology for detecting and quantifying bias in pre-trained language models. Our approach evaluates whether information encoded in representations of protected attributes (e.g., gender, nationality, religion) can be used to predict stereotyped attributes through a supervised classification task. Designed to balance interpretability and statistical rigor, the method requires only minimal labeled data.

Finally, we conducted a cross-model empirical study across multiple Transformer-based pre-trained models, revealing consistent associations between protected and stereotyped properties within the embedding spaces. Together, these contributions support a more comprehensive and empirically grounded understanding of bias in NLMs.

1.3. Document Structure

This document is structured as follows:

- **Chapter 2** provides background information on models in Natural Language Processing, by describing their evolution and the main architectures used today.
- **Chapter 3** introduces the problem of bias in language models, presenting a framework to analyze and categorize different types of biases.
- **Chapter 4** analyzes the current state of the art in bias detection and mitigation techniques for NLP, highlighting their strengths and limitations, with a focus on methods that consider training data influences.
- **Chapter 5** discusses the original research contributions, presenting some methods aimed at understanding bias encoding in language models.
- **Chapter 6** presents the experimental setting and application of the proposed methods to a variety of pre-trained language models.
- **Chapter 7** discusses and comments the implications of experimental results within their broader context.
- **Chapter 8** concludes the thesis by summarizing the key findings and suggesting directions for future work in the field of bias analysis in language models.

Chapter 2

Background on Natural Language Processing

Understanding how Neural Language Models (NLMs) work is essential for analyzing the issues of fairness that are the focus of this thesis. This chapter provides a technical overview of the main concepts underlying modern NLP systems, with a particular focus on the transformer-based architectures that dominate the contemporary NLM landscape.

We first introduce the self-attention mechanism, the fundamental component that revolutionized NLP by allowing models to selectively focus on relevant parts of the input. This mechanism was subsequently incorporated into the Transformer architecture, which derived from the encoder-decoder architecture and set the new standard for many language processing tasks.

The chapter then proceeds with the description of the two Transformer-derived architectures: encoder-only models such as BERT, optimized for text comprehension tasks, and decoder-only models such as GPT, designed for autoregressive text generation.

2.1. The Self-Attention Mechanism

In the 2010s, **machine translation** and other sequence-to-sequence tasks were typically addressed using encoder-decoder architectures by Cho et al. (2014). Those are neural network models composed of two main components: an encoder that processes the input sequence and a decoder that generates the output sequence (Figure 1).

> *Example.* In machine translation, the input is a sequence of words in one language, such as 🇬🇧 They threw me to the wolves. The corresponding output –

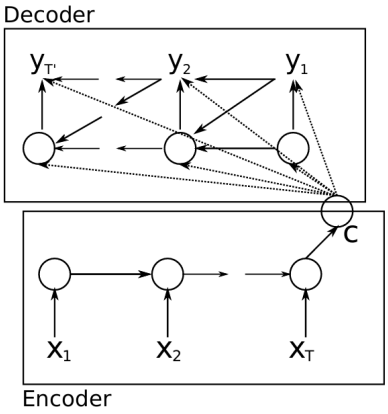


Figure 1: The encoder-decoder architecture schema, as it was presented in the original paper by Cho et al. (2014).

> which is the objective of the machine translation task – is a sequence of words in another language, such as 🇫🇷 Ils m'ont jeté dans la gueule du loup (Cho et al., 2014).

At the time, encoder-decoder architectures were often based on recurrent neural networks (RNNs) or convolutional neural networks (CNNs). These types of networks had limitations in handling long-range dependencies in the input sequence, which is crucial for tasks like machine translation. Vaswani et al. (2017) changed the paradigm by introducing the **Self-Attention Mechanism** — inspired by the **Attention Mechanism** by Bahdanau et al. (2015) — as a way to compute the interaction among all words in the input sequence, regardless of their distance from each other. In the same paper, the authors also introduced the **Transformer** architecture and the *Multi-Head Attention*, which are now the basis of most modern NLMs.

In the following sections, we describe these latter architectures and mechanisms and their different implementations.

2.1.1. Computing Self-Attention

The **Self-Attention Mechanism** was proposed by Vaswani et al. (2017) as an evolution of the Attention Mechanism computing the interaction among all words within the same sentence.

> *Example.* Given the sentence The animal did not cross the street because it was too tired, we try to understand whether the pronoun it refers to the word street or the word animal. For a human this question is trivial: it refers to animal. However, it is not simple for an algorithm. Given that it and animal are *linked together*, we want to assign a higher value to their connection rather than to the connection between it and street. Example by Alammar (2018).

This central intuition leads to the Self-Attention mechanism. As the model processes each word (each position in the input sequence), Self-Attention allows it to look at words in other positions in the input sequence and assigns a degree of importance related to the current word in analysis. As a general process, for each word in the input sequence the Self-Attention estimates an "attention vector", which contains the relative importance of every other word in the sentence to the analyzed word. Next, this vector is used to compute a weighted average of the representations of all words in the sequence; the resulting average is finally used as the new representation of the analyzed word.

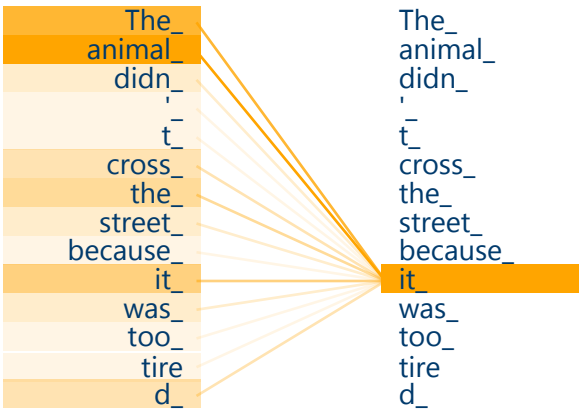


Figure 2: Importance of the words in the sentence for the word it computed by an attention layer. The word animal is more important (a brighter orange) than the word street. Example by Alammar (2018).

More formally, the first step in computing this representation is to create three vectors from each input word embedding:

(1) a **query** $\bar{q}_t$ (2) a **key** $\bar{k}_t$ (3) and a **value** $\bar{v}_t$

Let $\bar{x}_t \in \mathbb{R}^d$ be the embedding of the word at position t in the input sequence, with d being the original embedding dimension. These three vectors are created as follows: the input word embedding $\bar{x}_t \in \mathbb{R}^d$ is multiplied by three matrices that are trained during the training process: $W_q, W_k, W_v \in \mathbb{R}^{d \times b}$, where b represents the dimension of the internal representation subspace.

$$\begin{aligned} \bar{q}_t &= \bar{x}_t W_q \\ \bar{k}_t &= \bar{x}_t W_k \\ \bar{v}_t &= \bar{x}_t W_v \end{aligned} \tag{1}$$

Next, these three vectors are used to create a final representation $\bar{z}_t$ of the word $\bar{x}_t$. First, the scalar value α_{it} is computed, namely, a value that represents the influence (or *attention*) of a word $\bar{x}_i$ on the considered word $\bar{x}_t$:

$$\alpha_{it} = \text{softmax}\left(\frac{\bar{q}_t \bar{k}_i^\top}{\sqrt{b}}\right) \tag{2}$$

Then, the value α_{it} is exploited to compute $\bar{z}_t$:

$$\bar{z}_t = \sum_{i=1}^{n} \alpha_{it} \bar{v}_i \tag{3}$$

Example. As shown in Figure 3, let us analyze this process by a simple example using the words thinking and machines. First, the algorithm converts thinking and machines into vectors $\bar{x}_1$ and $\bar{x}_2$. Following Equation 1, we compute:

1. the queries $\bar{q}_1$ and $\bar{q}_2$,
2. the keys $\bar{k}_1$ and $\bar{k}_2$,
3. and the values $\bar{v}_1$ and $\bar{v}_2$.

Subsequently, we compute the two scalar products of the query $\bar{q}_1$ and the key vector of each word ($\bar{k}_1$ and $\bar{k}_2$), dividing them by $\sqrt{b}$ (Equation 2). We apply the softmax function to both results and multiply them by their relative

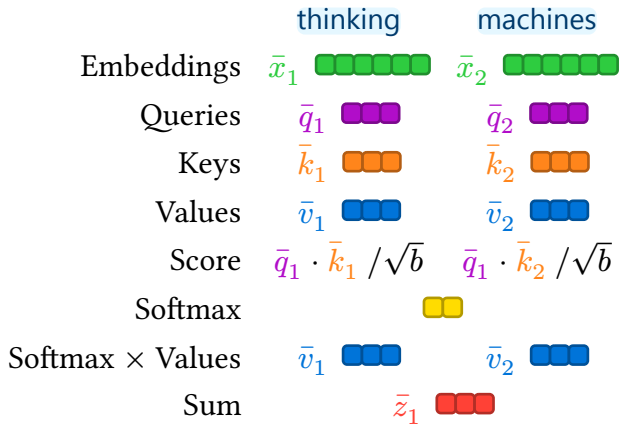


Figure 3: Steps of the Self-Attention procedure using the sentence thinking machines.

> value representations. Finally, we sum the two products to obtain the final representation $\bar{z}_1$ for the word $\bar{x}_1$. Example by Alammar (2018).

For simplicity, we analyzed the self-attention mechanism using vectors. However, it is better to express such formulas using matrices (as shown in Figure 4), as they are computed in parallel.

Consider the matrix X composed of the row vectors $(\bar{x}_1, ..., \bar{x}_n)$, corresponding to the embeddings $\bar{x}_i \in \mathbb{R}^d$ of the words in the sequence of length n:

$$X \in \mathbb{R}^{n \times d} \quad \text{where} \quad X_{[t]} = \bar{x}_t, \text{ for } t \in [1, n] \subset \mathbb{N} \tag{4}$$

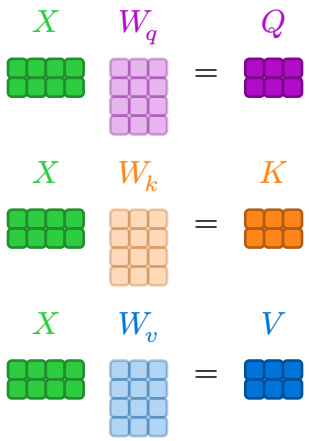
Equation 1 is redefined as:

$$\begin{aligned} Q &= XW_q \\ K &= XW_k \\ V &= XW_v \end{aligned} \tag{5}$$

with $Q, K, V \in \mathbb{R}^{n \times b}$. Following Equation 3 and Equation 2, we obtain a more compact definition of Self-Attention:

$$A = \text{attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{b}}\right) \cdot V \tag{6}$$

where $A \in \mathbb{R}^{n \times b}$ is a matrix that contains the final representation of the input in each row (i.e., $A_{[t]} = \bar{z}_t$ for $t \in [1, n]$).



$$\text{attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{b}}\right) \cdot V = \text{softmax}\left(\frac{\quad}{\sqrt{b}}\right) \cdot$$

Figure 4: Matrix operations to compute the Self-Attention mechanism with $X \in \mathbb{R}^{2\times4}$ and Q, K, $V \in \mathbb{R}^{2\times3}$. The final output is a matrix $Z \in \mathbb{R}^{2\times3}$ that contains the final representation of each word in its rows. Example by Alammar (2018)

2.1.2. Multi-Headed Attention

Vaswani et al. (2017) introduced also the *Multi-Headed Attention Mechanism*: multiple self-attentions are used in parallel to compute multiple representations. This approach has some advantages:

- it expands the model capabilities to focus on different positions;
- prevents the current word from dominating over the others;
- offers various representation subspaces (with different key, query, and value matrices).

As shown in Equation 7, the i-th Self-Attention mechanism takes the name of (i -th) **head**:

$$\text{head}_i = \text{attention}\left(XW_i^Q, XW_i^K, XW_i^V\right) \tag{7}$$

where $\text{head}_i \in \mathbb{R}^{n\times b}$, for $i \in [1, m]$.

The m parallel heads are then concatenated and weighted by a weight matrix $W^0 \in \mathbb{R}^{bm\times b}$, that is trained jointly with the model. The resulting representation is the $\text{multi-head}(Q, K, V) \in \mathbb{R}^{n\times b}$:

$$\text{multi-head}(Q, K, V) = \begin{bmatrix} & \dots & \\ \text{head}_1 & \dots & \text{head}_m \\ & \dots & \end{bmatrix} W^0 \tag{8}$$

2.1.3. Masked Self-Attention

The mechanism of **Masked Self-Attention**, introduced by Vaswani et al. (2017), was developed to prevent a word from attending to subsequent words. This attention can be used in decoder architectures to prevent the current word from learning from future words during training. Given the embedding matrix X of the word sequence, and considering the word $X[t] = x_t$, the Masked-Self Attention is almost identical to the self-attention except that all the tokens with an index $i \in [1, n]$ such that $i > t$ are masked (i.e., are set to zero) by applying a masking operator, as shown in Equation 6.

To implement this masking, the authors use an **Attention Mask** $\mathcal{A} \in \mathbb{R}^{n \times n}$, a matrix with values of 1 if $i \leq t$, or 0 otherwise. Given the attention mask, we set each value with a corresponding attention mask of 0 to $-\infty$ (subsequently, the softmax function converts $-\infty$ to 0):

$$\text{attention}(Q, K, V) = \text{softmax}\left(\text{mask}_{\mathcal{A}}\left(\frac{QK^\intercal}{\sqrt{b}}\right)\right) \cdot V \tag{9}$$

The Masked Self-Attention is essential to prevent information leakage from future input positions during training, conditioning each prediction only on the preceding context and enforcing causality.

Example. For instance, while training for a sequence-to-sequence translation task, we may want to translate the English sentence I am eating an apple to Italian Sto mangiando una mela. When the model is learning to translate eating to mangiando, we want to mask the rest of the sentence una mela to prevent the model from learning from those words, ensuring that the generation of mangiando is conditioned only on Sto. This training change

> is critical because the model will not have future words available during the generation phase.

2.2. Transformer Architecture

The different mechanisms proposed by Vaswani et al. (2017) and described in the previous sections of this thesis, were introduced by the authors as the building blocks of a new deep learning architecture called **Transformer**, which in the following years became the State-of-the-Art in *Natural Language Processing* (NLP) tasks. The Transformer architecture employs the *multi-head attention mechanism* as the unit component for the model's encoder and decoder stack, to obtain (in its original purpose) better performance in machine translation.

At a high level, the first component of the Transformer architecture, the **Encoder**, converts an input sentence to a corresponding hidden representation. The second component, the **Decoder**, exploits the hidden representation in an auto-regressive procedure to compute the translated sentence word by word iteratively.

In the following years, Devlin et al. (2019) and Radford et al. (2018) decomposed the architecture into two main components: the Encoder and the Decoder, obtaining **Bidirectional Encoder Representations from Transformers** (BERT) (Devlin et al., 2019), and **Generative Pre-trained Transformer** (GPT) (Radford et al., 2018), respectively. BERT allows classification tasks to be performed starting from a textual input by exploiting the hidden knowledge of the model, while GPT allows generative tasks to be performed starting from an initial textual prefix.

Later, various authors trained the GPT architectures into **Large Language Models** (LLMs), which demonstrate outstanding performances in different NLP tasks, show remarkable linguistic knowledge and exhibit forms of factual knowledge, common sense, and even programming skills.

The following subsections describe the main steps and components of the Transformer architecture.

2.2.1. Tokenization

In order for the Transformers to be able to process textual inputs, word sequences must be first transformed into numbers. This process, namely *Tokenization*, involves breaking the text into smaller text units called *tokens* via a statistical algorithm and associating each token with a unique numerical representation. Tokens do not have a one-to-one correspondence with words; they can represent whole words, subwords, or even individual characters, depending on the tokenization strategy employed.

In fact, tokenization methods vary across NLP models. For instance, Transformers and GPT implement *Byte-Pair Encoding* (BPE) tokenizer, which is a technique derived from text compression; BERT uses the *WordPiece* tokenizer to handle Out-of-Vocabulary (OOV) words by splitting them into meaningful subwords; other models employ other tokenizers.

2.2.2. Embedding of the Input Sequence

The goal of word embeddings in NLP is to convert each token/word into a real-valued vector such that semantically similar words have similar representations. In transformer architecture, the *embedding layer* is responsible to do that: it converts each token into a fixed-length real-valued vector (of size 512 in the original paper) using a learnable weight matrix, the embedding matrix $E \in \mathbb{R}^{v \times e}$, where v is the vocabulary size and e the embedding size.

2.2.3. Encoder and Decoder Stack

After completing the tokenization and embedding of the input sentence, we can divide the transformer architecture into two blocks: the encoder and the decoder stack (Figure 5).

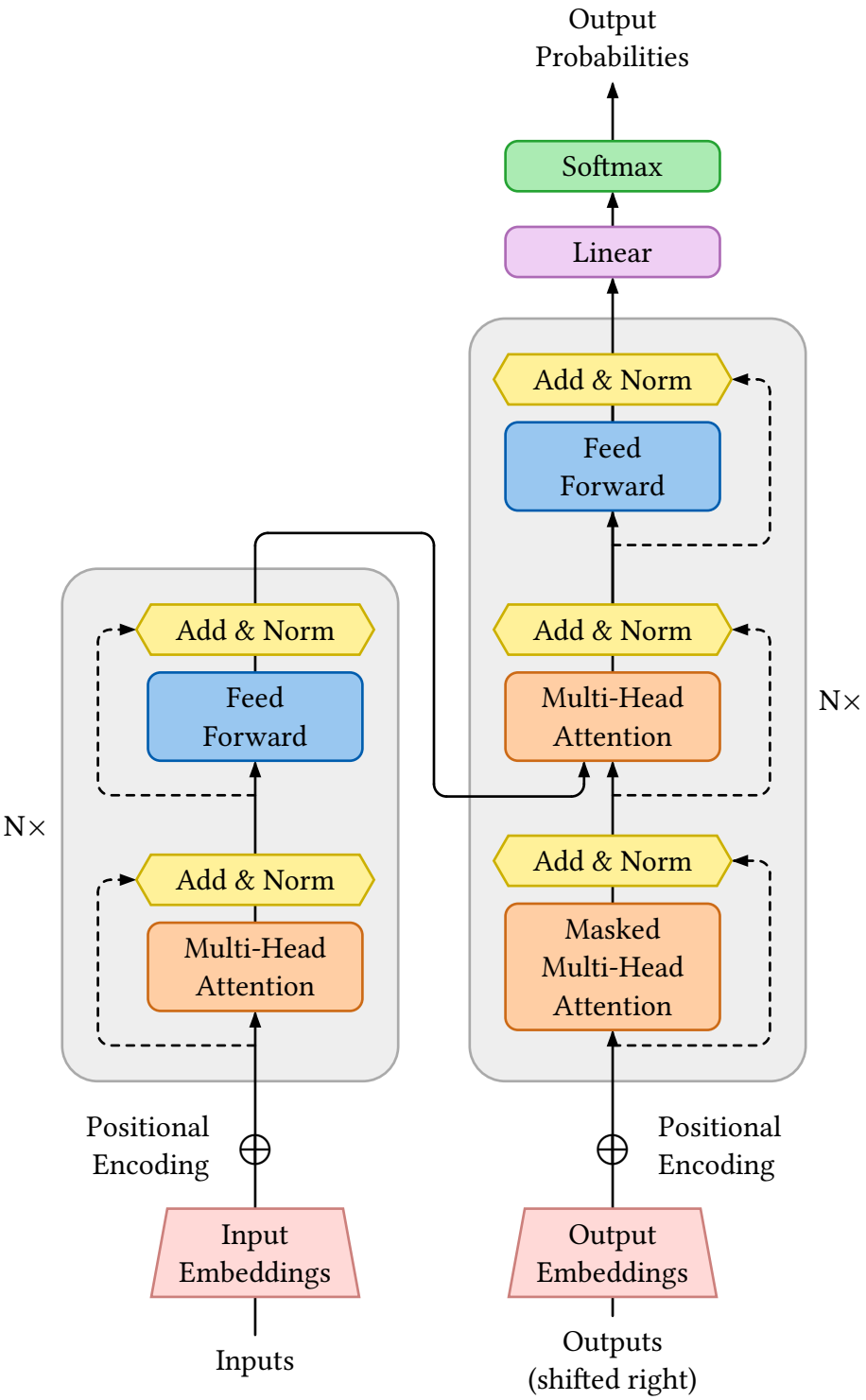


Figure 5: The *Transformer*-model architecture. Graphical schema by (Vaswani et al., 2017).

The encoder stack contains N **encoder blocks** ($N = 6$ in the original paper), each of one containing a multi-head self-attention sub-layer followed by a feed-forward network (applied independently in each position):

- The self-attention sub-layer allows every position in the encoder to depend on all positions in the previous layer, and it uses the multi-head attention mechanism described in Subsection 2.1.2.
- The fully connected feed-forward network consists of two linear transformations with a ReLU activation between them.

A residual connection followed by a layer normalization (the "Add & Norm" block in Figure 5) is applied to both the Multi-Head Attention and the FF sub-layers. That is, the output of each sub-layer is computed as $\mathrm{LayerNorm}(X + \mathrm{Sublayer}(X))$, where the LayerNorm shifts and scales the input to have zero mean and unit variance.

The decoder stack contains N **decoder blocks** (where N is a hyperparameter). A decoder block includes a sub-layer of masked multi-head self-attention that processes the newly generated tokens, followed by the multi-head self-attention sub-layer, which combines the hidden representation of the encoder stack with the previous output, and terminates with a feed-forward network:

- The masking in the first sub-layer of attention, combined with the fact that the output embedding is offset by one position, ensures that the predictions for the current token depend only on the previous ones.
- The second attention sub-layer, however, uses queries from the decoder's previous sub-layer combined with keys and values from the encoder's hidden representation, enabling attention across all input positions.

Similarly to the encoder, a residual connection is applied to each sub-layer, followed by a normalization step.

Through a fully connected neural network, the output of the last decoder block is projected into a much larger vector called **logits vector**, with length equal to the vocabulary size. Finally, a $\mathrm{softmax}$ layer turns those scores into probabilities (all positive, all add up to 1) (last block in Figure 5).

After these computation steps, thus, the Transformer model outputs a **probability distribution** over the vocabulary for the next token in the sequence. In order to generate a sequence of tokens (that is eventually converted back to text), the most likely token is selected and added to the input of the decoder for the next step. The process is iteratively repeated, until a special end-of-sequence token is generated or a predefined maximum length is reached.

2.2.4. Pre-training and Fine-tuning

After describing the inference process of the Transformer architecture, we can now analyze how the model is trained, i.e. how the inner parameters of the model are set in order to perform the desired task at inference time. This section will briefly mention the two main training phases of the Transformer-based models: pre-training and fine-tuning. Further details will be provided in the following sections for the specific GPT and BERT architectures.

The **pre-training** is the first training step of a Language Model; nowadays, LLMs are usually pre-trained using massive text corpora (e.g., Common Crawl, Wikipedia, books, etc.) in an unsupervised fashion. This phase allows the model to learn general language patterns, such as grammar, syntax, and semantics, by predicting the next token in a sequence given the previous tokens (auto-regressive models) or by predicting masked tokens (masked language models).

Resulting models are often referred to as **foundation models** (Bommasani et al., 2021), as they provide a solid base of linguistic knowledge that can be adapted to various downstream tasks.

Conversely, the subsequent phase is called **fine-tuning**, a continuation of the pre-training on a specific task to adapt the model to the desired application. Fine-tuning typically uses a smaller, labeled dataset specific to the task (e.g., sentiment analysis, question answering, etc.). The idea is to leverage the general language understanding acquired during pre-training and specialize it for the specific requirements of the target task, often resulting in significant performance improvements compared to training a model from scratch.

2.3. Bidirectional Encoder Representations from Transformers (BERT)

BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) distinguishes itself from the original Transformer architecture by using only the **encoder** component. Unlike auto-regressive models that process text unidirectionally, BERT's *bidirectional* attention mechanism allows it to condition on both left and right context simultaneously, enabling richer contextual representations.

This peculiarity reflects into the pre-training choices. BERT is pre-trained on two unsupervised tasks:

- **Masked Language Modeling** (MLM), where random tokens are masked and the model learns to predict them using bidirectional context (i.e. knowing both left and right surrounding words);
- and **Next Sentence Prediction** (NSP), where the model learns to determine whether two sentences are consecutive.

This pre-training strategy is usually followed by a task-specific fine-tuning.

2.3.1. RoBERTa

RoBERTa (Robustly Optimized BERT Pretraining Approach) (Liu et al., 2019) demonstrated that BERT was significantly undertrained and introduced several key modifications to the pre-training procedure.

- The model removes the Next Sentence Prediction objective entirely, finding it unnecessary and potentially detrimental to performance.
- RoBERTa trains with larger batch sizes, longer sequences, and dynamically changes the masking pattern applied to training data across epochs rather than using static masks.
- Additionally, it trains on more data (160 GB versus BERT's 16 GB) for longer durations.

- Finally, it uses byte-level *BPE* tokenization (the same as used by GPT) instead of character-level *WordPiece*.

2.3.2. ELECTRA

ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately) (Clark et al., 2020) introduces a fundamentally different pre-training objective that proves more sample-efficient than MLM.

Rather than masking tokens, ELECTRA trains a discriminator model to distinguish between real input tokens and tokens replaced by a small generator network (trained simultaneously with MLM). This detection task requires the model to make predictions for all input tokens, rather than just the masked ones (typically 15% in BERT), providing a stronger training signal per example.

The generator-discriminator setup resembles *Generative Adversarial Networks* (GANs) by Goodfellow et al. (2014), but uses maximum likelihood training and not adversarial training. Like GANs, only the discriminator is used for downstream tasks, while the generator is discarded.

This approach achieves comparable performance to BERT and RoBERTa while requiring significantly less computation, making it particularly valuable when computational resources are limited.

2.3.3. DistilBERT

DistilBERT (Sanh et al., 2019) addresses the computational cost of deploying large pre-trained models through **knowledge distillation**, maintaining approximately 97% of BERT's language understanding capabilities while reducing model size by 40% and increasing inference speed by 60%. The distillation process trains a smaller *student model* (6 layers instead of BERT's 12) to mimic the behavior of the pre-trained BERT *teacher model* by minimizing a combination of distillation loss (matching the teacher's output distributions), masked language modeling loss, and cosine embedding loss (aligning hidden state directions).

Unlike BERT's original training, DistilBERT does not use the NSP objective and employs a combination of triple loss functions during distillation. The architecture retains BERT's token-type embeddings and positional embeddings but reduces the number of layers, demonstrating that much of BERT's knowledge can be compressed into a smaller model suitable for resource-constrained environments without architectural innovations beyond layer reduction.

2.4. Generative Pre-Trained Transformer (GPT)

As shown in the previous section, the transformer architecture comprises the decoder and the encoder stacks to solve a machine translation problem. Liu et al. (2018) removed the encoder stack and used the decoder stack of the transformer architecture to perform text generation, naming this architecture *transformer decoder*. Following this work, Radford et al. (2018) introduced the **Generative Pre-trained Transformer** (GPT), training the transformer decoder architecture on an unlabeled large text corpus, followed by a fine-tuning procedure on a target task. In GPT, the decoder component employs a left-to-right auto-regressive language modeling, i.e. the model predicts the next token given all previous tokens. Each predicted token is then appended to the previous input, and the procedure continues until a specified number of tokens have been generated or a stopping condition is met, such as encountering a unique end-of-sequence token.

2.4.1. GPT-NeoX and Pythia

GPT-Neo (Black et al., 2021) is an open-source suite of models ranging from 125M to 2.7B parameters, designed by EleutherAI to replicate the performance of OpenAI's GPT-3 (Brown et al., 2020). The suite has been followed and improved by GPT-NeoX (Andonian et al., 2023), a library including larger models like GPT-NeoX-20B.

All GPT-Neo and GPT-NeoX models were trained on *The Pile* dataset (Gao et al., 2020), a diverse 825 GB corpus designed specifically for training LLMs.

Pythia by EleutherAI (Biderman et al., 2023) is again a series of decoder-only models, ranging from 70M to 12B parameters, all pre-trained on identical data in the same order, with checkpoints saved at regular intervals throughout training (154 checkpoints per model). This controlled experimental setup enables researchers to study training dynamics, memorization, few-shot learning emergence, and other phenomena as functions of model scale and training progression.

Chapter 3

Background on AI Fairness

Recent years have witnessed a growth of scientific literature addressing fairness in Artificial Intelligence systems, driven by the emergence of novel models and growing interdisciplinary interest. As a consequence, fairness has become an essential requirement in system development, allowing the creation of diverse methodologies to evaluate and ensure it.

In this chapter, we provide a general background on the concept of fairness in AI. We start by situating fairness within the broader landscape of AI ethics, then we briefly discuss different definitions and perspectives on what constitutes a fair system. Finally, we introduce the specific challenges and approaches to studying fairness in NLP, which will be the main focus of this thesis.

3.1. The AI Ethics Landscape

Fairness is one of several core principles within **AI Ethics**, that is the branch of Philosophy which addresses moral values and principles for the development, deployment, and governance of artificial intelligence systems (Jobin et al., 2019). To study the "ethical" impact of AI means to analyze the technology along multiple interlaced dimensions, such as transparency, accountability, privacy, safety, human oversight, and – of course – fairness.

The increasing deployment of AI systems in high-stakes domains (such as criminal justice, hiring, credit lending, and healthcare) has exposed how algorithmic decisions can perpetuate or amplify existing social inequalities. Recognizing this critical concern, policymakers and international bodies have begun to embed ethical principles into regulatory frameworks and legal initiatives. Notable examples include the **AI Act** (2024), adopted by the **European Union** in 2024 after an intensive legislative process, which establishes a risk-based regulatory

framework requiring high-risk AI systems to meet specific fairness and non-discrimination requirements before deployment. Some of the key provisions were previously included in the **General Data Protection Regulation** (GDPR) (2016), which introduced the right to explanation and data protection principles that indirectly promote fairness in AI systems.

As a consequence of the legislative efforts and social requirements, the development and deployment of fairness research in AI Ethics does not occur in an abstract vacuum, but rather is deeply influenced by political, economic, and cultural factors. A compelling example of how these influences shape scientific research agendas is evident in the trajectory of fairness literature itself. For instance, the majority of scientific production addressing bias in NLP and AI has focused on **gender bias**: 80% of the papers from the last decade address this topic, adducing the argument that "gender bias is one of the easiest to study, given the existence of large datasets" (Ghosh & Wilson, 2025).

This hierarchy of research priorities is not arbitrary: it reflects underlying social pressures, policy concerns, and the advocacy efforts of particular communities. While the technical methodologies employed to study fairness are often agnostic *in theory*, they are necessarily applied to contextualized problems *in practice*. However, this thesis primarily focuses on the technical dimensions of fairness, and thus will largely "abstract away" from these considerations for the remainder of the document. Nevertheless, it remains important to bear this context in mind when interpreting our technical findings.

3.1.1. The quest for a fair system

To grasp the general idea, fairness is the *"absence of any prejudice or favoritism toward an individual or group based on their inherent or acquired characteristics"* (Mehrabi et al., 2021). In practice, a **fair system** is one whose decisions or representations do not systematically disadvantage individuals or groups because of protected characteristics. This does not imply the absence of all errors, but rather the absence of **systematic** and **unjustified** disparities across groups,

given the task and the social context. This might seem like a straightforward definition, easy to operationalize, but in reality, the concept of fairness is complex. It is not clear what it means for a system to be "unjustified" or "systematic", and different stakeholders may have different intuitions about what constitutes a fair outcome.

In recent years, researchers have proposed a wide variety of formal definitions of fairness, each capturing different ethical principles and social values (Verma & Rubin, 2018). When developing an AI system, one might want to guarantee *equalized odds* or *equal opportunity* (Hardt et al., 2016), *demographic parity* (Dwork et al., 2012), *treatment equality* (Berk et al., 2021), and so on (Mehrabi et al., 2021; Verma & Rubin, 2018). To have multiple definitions of fairness is not a problem in itself; one may want to just satisfy the most of them. However, this is not feasible: Kleinberg et al. (2017) mathematically proved that the two fairness constraints of *calibration* and *balancing the positive and negative classes* are incompatible, which means that not every fairness definition can be satisfied at the same time for the same system.

The aforementioned definitions are valid for and usually applied to simple Machine Learning classifiers, in which the input is a precise set of features and the output is a single prediction. And yet, literature does not provide a single easy definition of what a fair system is. The problem becomes even more complex when we consider the case of large language models, as we will see at the end of this chapter.

3.1.2. Intrinsic and extrinsic fairness

Before delving into the specific challenges of fairness in NLP, it is important to set a distinction: fairness of AI systems, and particularly of NLP, can be studied from two complementary perspectives that offer different insights into how and where bias manifests.

Intrinsic fairness examines the internal representations of a model — the hidden layers, embeddings, attention patterns, or latent features — asking

whether sensitive properties (such as gender, race, or religion) are encoded or entangled with stereotyped properties in ways that reflect societal biases. This perspective focuses on *what the model learns* rather than *what it produces*.

Intrinsic analyses are often used to diagnose representational biases at their source, providing interpretable evidence of how stereotypes are encoded in the model's internal geometry. They enable researchers to identify problematic associations before deployment and can guide targeted debiasing interventions. For example, intrinsic analysis might reveal that word embeddings position programmer closer to male-associated terms than to female-associated terms, suggesting an encoded gender stereotype (Bolukbasi et al., 2016).

However, intrinsic fairness alone does not guarantee fair outcomes: a model might have biased representations yet produce fair predictions due to downstream mechanisms, or conversely, might have relatively unbiased representations yet produce unfair outcomes due to task-specific learning dynamics (Gonen & Goldberg, 2019; Nissim et al., 2020).

Extrinsic fairness, by contrast, evaluates the model's observable behavior in real-world applications — its predictions, classifications, generated text, or downstream task performance — asking whether outcomes systematically differ across demographic groups. This perspective focuses on *the consequences of model deployment*.

Extrinsic analyses quantify how representational biases surface in practical applications and manifest as disparate impact across groups. They provide direct measurements of harm in real-world contexts and are often more aligned with regulatory requirements and user-facing concerns, with respect to intrinsic approaches (Blodgett et al., 2020). For instance, extrinsic analysis might measure whether a candidates-screening system ranks qualified male candidates higher than equally qualified female candidates, or whether a sentiment analysis model assigns more negative scores to text mentioning certain ethnic groups.

However, extrinsic unfairness can sometimes be harder to interpret in terms of underlying causes, as it may arise from complex interactions between data,

model architecture, and task-specific learning dynamics. Furthermore, it limits the analysis to specific domains and datasets, potentially overlooking broader representational issues.

3.2. Fairness in Natural Language Processing

In this thesis, most of the dissertation involves and focuses on the problem of **fairness in the NLP field**: the analysis considers words and texts fed to and generated by models, with the purpose of understanding whether their processing can be seen as *fair* or *unfair*. This approach — which mostly considers *intrinsic* fairness — is not unusual in NLP literature. Especially in the first years after the Transformer-based models came out (2017–2021), language model fairness has been assessed on *language representation*, meaning that the biases of the model considered are usually identified and mitigated by studying and changing, respectively, how the text is depicted within the system (Garrido-Muñoz et al., 2021). With the advent of LLMs (2022-2026), complementary approaches based on *extrinsic* fairness started to spread more in the scientific community; however, these will be addressed later in the thesis.

3.2.1. Operationalizing intrinsic fairness in NLP

The effectiveness of the aforementioned perspective is particularly pronounced in models relying on **word embeddings**. These models employ numerical encoding to represent words, ideally capturing their semantic essence. The underlying assumption, coming from the semantic theory of language usage, states that the words appearing in the same contexts tend to have a similar meaning (Harris, 1954), or equivalently that "a word is characterized by the company it keeps" (Firth, 1957). This proposition (namely, the **distributional hypothesis**) conceptualizes the language as a semantic space that the models encode in a vector space. Within this spatial representation, the semantic similarity is trans-

lated into a geometric proximity, facilitating the understanding of relationships between *words* by studying relationships between *vectors*.

It is within this analytical framework that *stereotypes* and *unfair representations* become perceptible as undesired geometric distributions. For instance, the proximity of the word "muslim" to terrorism-related terms may signify an implicit similarity that the model has learnt and that is often derived from a stereotype concealed in data.

While word embeddings provide a technical foundation for studying fairness through geometric distributions, a more structured approach is needed to systematically analyze and measure bias in language. In this thesis, the experimental sections adopt a **property-based framework** (Garrido-Muñoz et al., 2021) that bridges abstract fairness principles with concrete linguistic analysis. This framework allows us to:

- define which human characteristics we care about protecting (protected properties);
- identify which dimensions might exhibit discriminatory patterns (stereotyped properties);
- measure correlations and relationships in the language representation;
- apply targeted evaluations across different model types and languages.

We establish a formalism based on **properties** and **classes** to represent human attributes and their linguistic manifestations. We start by defining some properties applicable to human beings, such as gender, job, religion, behavior, nationality. A **property** (also called attribute) is a sort of variable that can hold a finite domain, typically formed by at least two values; for instance, a person can have the property nationality either as *German* or *Spanish*. Values are sometimes called **classes**, and identify some information about the human they refer to. In this document, we use the underlined notation for properties and the *italic* font style for classes.

To link these concepts to a language, each class can be associated with a series of **terms** from that language. For instance, the *male* class of the gender property

is represented by the words male, he, father, brother, etc., whereas the *female* class of the gender property is represented by the words female, she, mother, sister, etc. Each of these terms indicates one and only one value for the given property.[1]

By means of this framework, we approach the study of stereotypes in natural language, and therefore we can address the fairness requirement on words and sentences. To do that, the fairness idea is built upon the concept of prejudice, which regards the interconnection between two properties. For example, gender is not unfair in itself, but it can be when a prejudicial relationship with the salary property rises. In the same way, words defining the ethnicity of a person are not inherently biased, but they can be when related to words describing the criminality of subjects. As the reader may notice, prejudices come with two attributes, respectively called **protected** and **stereotyped properties**.

The **protected** attribute often defines the human categories that are considered minorities or marginalized groups in the social and juridical fields, whereas the **stereotyped** attribute expresses the dimension in which the discrimination manifests. However, the difference between the two is merely conceptual: any property could theoretically cover the role of the protected attribute or the stereotyped attribute. In other words, given a pair of properties (protected and stereotyped), we observe a **bias** if the two properties are correlated or have some sort of relationship. For instance, if the profession stereotyped property is seen as related to the way we represent the gender protected property, we have a distortion in the representation and, thus, a bias.

3.2.2. Working Definitions and Terminology

To conclude this section, we provide a glossary of key terms and definitions that will be used throughout the thesis, resuming the property-based framework and establishing a consistent vocabulary for our technical analysis.

[1] A property approximates reality to some extent and might not reflect the real-world situation entirely. The pronoun he, for instance, can be used and associated to genders other than the *male* class, and so do many words of the previous example.

Property (or Attribute) — A human characteristic or demographic dimension that we seek to study, such as the gender or the profession of a person. A property comprises a finite set of possible values called **classes**. In our framework, we distinguish two types:

- **protected property**: a demographic characteristic that defines potentially marginalized or vulnerable social groups (e.g., gender, religion, nationality);
- **stereotyped property**: a characteristic that may be systematically associated with protected properties through societal biases. Stereotyped properties represent the dimensions along which discrimination or unfair associations may manifest. Examples include the profession, a personality trait (adjective), or the likelihood of action (verb).

Class (or Value) — One of the discrete categories within a property. For example, *male* and *female* are classes of the gender property, while *christian* and *muslim* are classes of the religion property. Each class is typically represented in language by specific words or expressions (e.g., he, father for *male*).

Word — A linguistic expression (term, phrase, or pronoun) that maps to a specific class within a property. Words serve as the linguistic operationalization of abstract categories. For instance, the *female* class might be represented by terms such as she, woman, daughter, and queen.

Template — A sentence with placeholder(s) designed to insert words of interest while keeping context consistent. Templates are used to generate sentences containing our target words in natural linguistic contexts. For example, the template "[subject] is a very [adjective] person" can be instantiated with different terms to create sentences like "John is a very kind person" or "Marie is a very famous person".

Stereotype — A generalized belief or expectation that certain classes of a protected property are always associated with certain classes of a stereotyped property. For example, when the association between *male* individuals and *technical* jobs is the only association possible for the gender×profession properties, we have a stereotype. In fact, stereotypes may encode real-world statistical pat-

terns (e.g., gender distribution in professions), but they often oversimplify reality, distorting and amplifying these patterns.

Bias — The distortion in learned representations or behaviors of a model that systematically reflect a stereotype. Bias is usually measured as a deviation from an idealized state of fairness, where no systematic associations exist between protected and stereotyped properties. In this chapter, we presented a primary distinction between:

- **intrinsic bias**: related to inner representations of concepts within a model, manifesting in a skewed correlation of classes in the model's internal geometry;
- **extrinsic bias**: related to the model's observable behavior in downstream tasks, manifesting systematic disparities in outcomes across demographic groups.

Fairness (intrinsic) — From the perspective of representational fairness, the property of a model whose learned representations do not exhibit systematic correlations between protected and stereotyped properties. In other words, the model should not internally encode associations that reflect stereotypes.

Chapter 4

State of the Art

This chapter provides a comprehensive overview of the literature on bias in deep Neural Language Models, with a particular focus on fundamental approaches for bias detection, mitigation, and understanding bias acquisition in NLP models. The perspective adopted by this thesis is grounded in explanatory analysis: both the approaches discussed in the sections that follow and the experiments presented in Chapter 6 are designed to move beyond simply acknowledging the existence of bias in models. Instead, they aim to provide concrete insights into the underlying mechanisms and causes that lead to an unfair representation and behavior of NLP models.

To achieve this goal, a significant portion of this chapter draws from a comprehensive survey of the rapidly expanding body of research on bias in NLP. This survey encompasses multiple viewpoints, research positions, ongoing contradictions, and unresolved questions within the field. The survey itself is currently under review for publication, and its findings inform the structure and content of the sections outlined below:

- Section 4.1 addresses **bias detection** techniques for identifying and measuring bias in models;
- Section 4.2 offers a brief view of **bias mitigation** strategies, as they are often intertwined with detection methods and provide important context for understanding the implications of bias measurement;
- Section 4.3 reviews the role of **data** in bias acquisition, including how different data sources, data collection methods, and data preprocessing techniques can contribute to bias in NLP models;

- Section 4.4 explores the influence of **training procedures**, including how different optimization techniques and training regimes can affect the emergence of bias in models;
- Section 4.5 examines the impact of **model architecture** on bias, including how different architectural components (e.g., attention mechanisms, embedding layers) can be affected by and contribute to bias in NLP models;
- Section 4.6 finally investigates the relationship between **model complexity** (e.g., model size, number of parameters) and bias.

4.1. Bias Measurement

The first part of this chapter presents an overview of the methods to *detect*, *measure*, and *quantify* social bias in Language Models (LMs), with an emphasis on modern Large Language Models (LLMs). Discussion will not be limited to the most recent studies, but rather includes seminal works that laid the foundations for this line of research and more recent studies that have advanced our understanding of bias in LMs. Following common distinctions in the fairness literature, we primarily discuss *representational harms* as measured directly in model representations or model scores, and *behavioral* measurements that probe LMs through prompted predictions or generations.

4.1.1. From word embeddings to LM bias measurement.

The work of Bolukbasi et al. (2016) is widely regarded as seminal, because it formalized how social bias can be operationalized in distributed representations through (i) identifying a "bias direction" (e.g., gender) and (ii) quantifying association and proximity patterns in embedding space. This study opened a large body of research reporting the severe downside of blindly training LMs on large text corpora. Similar concerns were expressed by Schmidt (2015) in the same years. This line of work connected NLP bias measurement to decades of social-science methodology: Caliskan et al. (2017) introduced the *Word Embedding Association*

Test (WEAT), explicitly mirroring the Implicit Association Test (IAT), and showed that association scores in embeddings correlate with real-world social regularities.

The first models analyzed by the aforementioned papers were based on a **static** word embedding procedure (*word2vec*, *GloVe*). However, Guo & Caliskan (2021) further analyzed bias measurement in embeddings and its sensitivity to design choices, introducing the *Contextualized Embedding Association Test* (CEAT) to handle contextual embeddings. The transition from static embeddings to contextual encoders required revisiting association-based measurement. May et al. (2019) proposed *Sentence Encoder Association Tests* (SEAT), extending WEAT-style hypothesis testing to contextual sentence encoders.

In parallel, template-based and likelihood-based probes became a dominant strategy for Masked Language Models (MLMs) such as BERT. Zhao et al. (2019) provided an early systematic analysis of how gender bias manifests in ELMo, measuring the different behavior in gender-varying tests through template-based probes. Kurita et al. (2019) proposed comparing conditional probabilities of attributes given target terms in MLMs, aiming to better isolate association effects from lexical frequency. Such approaches established an influential methodological pattern that later carried over to autoregressive LLMs: define minimal pairs or controlled templates differing only in a protected attribute (or its proxy), then measure differences in model likelihoods, ranks, or predicted tokens.

Bias detection was also extended beyond English-centric settings. For instance, Zhou et al. (2019) studied how grammatical gender and typological properties can interact with gender bias measurements across languages. In addition, work such as (Maudslay et al., 2019) used controlled perturbations and distributional analyses to diagnose how gender information is encoded in learned representations.

As the field progressed, its complexity was further highlighted by studies that revealed how different methods can yield divergent results. For instance, the straightforward relationship between bias and word embeddings was chal-

lenged by Gonen & Goldberg (2019), who showed that geometry-based debiasing techniques could fail to remove bias from downstream predictions, leading to the "lipstick on a pig" metaphor. Nissim et al. (2020) further argued against the analogies-based approach to bias detection, on which many early studies relied, exhibiting how such methods can be sensitive to subjective design choices and thus may not be the appropriate tool for capturing biased associations in NLMs.

4.1.2. Stereotypes in LMs' text generation and downstream behavior

With the rise of pre-trained LMs used via prompting, bias measurement shifted from intrinsic association tests toward task-like evaluations and curated bias benchmarks. For instance:

- *StereoSet* benchmark (Nadeem et al., 2021) evaluates whether LMs prefer stereotypical continuations over anti-stereotypical or unrelated ones.
- *CrowS-Pairs* (Nangia et al., 2020) uses sentence pairs that differ minimally in demographic markers to estimate the relative preference for stereotyped statements across multiple bias categories.

These benchmarks helped standardize measurement for pre-trained LMs by framing bias as a **comparative preference** over controlled alternatives, but they also surfaced important limitations: sensitivity to template artifacts, narrow coverage of linguistic phenomena, and challenges in ensuring that anti-stereotypical options are equally plausible.

As autoregressive LLMs later became widely deployed for free-form generation, detection efforts expanded from next-token preference to distributional properties of generated text. Sheng et al. (2020) demonstrated that neural language generation can exhibit systematic gender bias, motivating template prompts that elicit continuations and then quantify differences via sentiment, occupational mentions, or other attributes. Gehman et al. (2020) introduced *RealToxicityPrompts* to measure toxic degeneration in generation by conditioning on naturally occurring prompts and scoring outputs with toxicity classifiers. Complementary work proposed measuring *regard* (the respect conveyed toward

a demographic group) and related affective dimensions to capture harms not fully represented by stereotype agreement alone (Sheng et al., 2021).

Another influential thread evaluates bias through downstream tasks where model predictions can encode or amplify stereotypes. For example, **coreference resolution benchmarks** such as *WinoGender* (Rudinger et al., 2018) and *WinoBias* (Zhao et al., 2018b) quantify systematic disparities in resolving pronouns across genders under controlled lexical contexts. While these settings are not intrinsic LM probes, they remain central to the bias-detection literature because they operationalize harms in end-to-end decisions and reveal how pre-trained LM representations interact with task supervision.

4.2. Bias Mitigation

This section surveys approaches that aim to **reduce measured social bias** in LMs, including both representational bias in internal embeddings and behavioral bias in predictions or generations. Following a common taxonomy, we distinguish among interventions applied *before training* (data-centric), *during training* (objective/model-centric), and *after training* (post-hoc).

4.2.1. Data-centric mitigation

A substantial body of work — focused on **data-centric interventions** — reduces bias by altering the data distribution that the model learns from. This includes balancing training corpora, filtering or down-weighting problematic text, and *Counterfactual Data Augmentation* (CDA) (Lu et al., 2020; Zmigrod et al., 2019). With CDA, new instances are added to the training set of a model with the aim of reducing class imbalance for a protected property. For instance, if the term engineer is strongly associated with the pronoun he, with CDA new instances with engineer and she (such as the simple sentence she works as an engineer) are generated; this augmented dataset will then be used to fine-tune or even re-train the model. In supervised settings, CDA and related perturbation strategies have

been used to reduce gender bias in coreference resolution and other tasks while maintaining accuracy (Zhao et al., 2018b). This approach has been proved to be effective for gender-related issues, while preserving internal factual knowledge (Xie & Lukasiewicz, 2023). Similarly, Gupta et al. (2022) demonstrated that by swapping demographic terms in the training dataset (*Counterfactual Role Reversal*) a more equitable treatment can be encouraged during text generation.

From the bias mitigation perspective, **multilingual models** — namely, models trained on data in multiple languages — can be challenging. On the one hand, there is some evidence that multiple-source data, and in particular multilingual data, could help mitigate specific cultural biases both in the performance in downstream tasks and in the embedded representation of concepts (España-Bonet & Barrón-Cedeño, 2022; Zhou et al., 2023). However, considering the findings of Zhao et al. (2024), LLMs handle multilingualism in three steps: first, they translate the user query in English; next, they solve the task in English; and finally they generate the answer in the original language. Consequently, performing debiasing and detoxification for the English language might have an impact on other languages too. This has been successfully demonstrated by Neplenbroek et al. (2025) using supervised fine-tuning and *Direct Preference Optimization* (DPO) (Rafailov et al., 2023) on several NLMs such as Gemma and LLaMa. However, this frequently comes at the cost of reduced language generation ability (e.g., lower fluency or coherence) in the target languages. Moreover, there are significant differences across languages, with the lower resource ones (such as Dutch or Catalan) being the most problematic.

4.2.2. Objective/model-centric mitigation

With regards to **model-centric mitigation**, the idea is to intervene on the model itself, either by modifying the training objective or by altering the model architecture. For NLMs based on contextual representations, mitigation often targets the *encoding* of protected attributes in hidden states. For instance, one family of methods trains a *probe* or an *adversary* model to predict a protected attribute from

word representations; based on the probe's performance, the method identifies which components of the representation encode protected-attribute information, and then updates the original NLM to reduce this predictability (Elazar & Goldberg, 2018; B. H. Zhang et al., 2018).

An approach by Huang et al. (2020) related to CDA uses counterfactual pairs for **model regularization** during training, penalizing the model if the hidden states for an original sentence (e.g., ...his career...) and its counterfactual version (e.g., ...her career...) are not sufficiently close in the embedding space.

Mitigation can also be achieved through *fine-tuning*, using regularizers or multi-objective training. A study by Zhao et al. (2018a) modified the embedding training objective to constrain gender information, highlighting trade-offs between removing bias and retaining legitimate semantic information. Other cases include adding penalties that discourage biased associations under controlled template probes, or optimizing for both task loss and fairness-oriented constraints (Kaneko & Bollegala, 2021; Meade et al., 2022).

An additional refinement (or *alignment*) technique for NLMs, and especially LLMs, is **Instruction Tuning** (IT), a further training of a model considering pairs of instructions (expressed in natural language) and preferable outputs (Ouyang et al., 2022; Zhang et al., 2026). This can be performed using Reinforcement Learning, especially with the well known *Reinforcement Learning from Human Feedback* (RLHF) technique (Ouyang et al., 2022). From a fairness perspective, several studies have analyzed the impact of Instruction Tuning on NLMs bias, especially its success as a mitigation technique. Y. Chen et al. (2025) showed on a custom test set that instruction-tuned versions of several models (LLaMa 3-7B and 8B, Mistral 7B and Gemma2-7B and 9B) can exhibit higher stereotypical associations among gender and occupations. Zhao et al. (2025) analyzed different manifestations of bias in LLama3.2 1B and 2B. Similarly, Sun et al. (2025) showed that the IT process on Llama 3-70B decreases extrinsic race bias by making the model "race blind", but its intrinsic bias remains stable. Although it seems that there is some evidence regarding the limited impact of Instruction Tuning in

intrinsic bias, if we consider the multiplicity of models, datasets and techniques, much further investigation might be required.

4.2.3. Post-hoc mitigation

Finally, post-hoc mitigation techniques intervene **after the model is trained**, without modifying its internal parameters. An example comes from Bolukbasi et al. (2016), which proposed a post-processing procedure (often called *hard debiasing*) that neutralizes gendered components for supposedly gender-neutral words while preserving definitional gender terms. This intervention occurs *after* inference time, meaning that the model working is unaffected. Few years later, Ravfogel et al. (2020) proposed the *Iterative Nullspace Projection* to iteratively project out directions that enable a linear classifier to recover protected attributes. These approaches connect directly to the diagnostic literature: "removal" is often evaluated by how much protected-attribute predictability decreases (via probes) and how downstream performance degrades.

For auto-regressive LMs, a practical family of methods reduces undesirable generations without retraining the base model. This includes controlled decoding with auxiliary discriminators or experts, such as *Plug and Play Language Models* (Dathathri et al., 2019) and *GeDi* (Krause et al., 2021), which steer generation away from toxicity or toward non-stereotypical continuations. Schick et al. (2021) proposed *self-debiasing*, which uses contrastive prompting to penalize biased continuations at decoding time. Such methods are typically assessed using the same detection benchmarks discussed earlier (e.g., toxicity and stereotype preference), making the measurement pipeline a central component of mitigation research.

4.3. Bias and Data

This section investigates the role of **data** in bias acquisition, including how different data sources, data collection methods, and data pre-processing techniques

can contribute to bias in NLP models. First, a characterization of bias at the data level is presented, discussing different situations in which bias can be identified in raw data. Second, we will examine the different steps in which data might enter a language model and produce a biased behavior.

4.3.1. Bias as class imbalance

A first take on how bias can derive from data stems from a common problem in machine learning: class imbalance. In (Subramanian et al., 2021), it is shown that the performance of BERT in classical NLP tasks (such as toxic comment classification) can differ for majority or minority groups, illustrating that bias originates when the model learns primarily from dominant subgroups, leaving it unable to accurately represent instances that deviate from the majority stereotype within a given label. Similarly, Valentini et al. (2022) showed that gender bias metrics based on word embeddings are inadvertently influenced by word frequency rather than just semantic meaning. By testing static models (like *GloVe* and *Skip-gram*), the authors discovered a spurious correlation where high-frequency words are consistently flagged as male-biased, while low-frequency words often skew female.

In literature, the problem of unbalanced classes is often countered with *Counterfactual Data Augmentation* (CDA) (Zmigrod et al., 2019) and similar techniques (Gupta et al., 2022; Huang et al., 2020). A common reasoning behind this approach is that if a model is trained on a more balanced dataset, it will learn more equitable associations and therefore produce less biased outputs. In this sense, the success of bias mitigation techniques that rely on data augmentation (discussed in Section 4.2) is often interpreted as evidence that class imbalance in training data is a key driver of bias in language models. However, with regards to open-ended text generation, there is some evidence that forcing class balance on datasets does not necessarily correlate to a fairness improvement on embedding-level tasks or in downstream classification tasks (Chen et al., 2024; Gupta et al., 2022).

4.3.2. Bias as data property

A complementary perspective on how data influence bias and the learning of stereotypes see training corpora as an archive of cultural, societal, historical and political norms, which therefore are somehow assimilated by the models based on those data.

Several studies have analyzed this perspective considering different point of views and datasets. A study by Madhusudan et al. (2025) fine-tuned different LLMs on the *BookPAGE* corpus, a collection of bestselling novels from each decade between the 1950s and the 2010s. Analysing models fine-tuned with data from different decades, they showed how their behaviour change over time. For instance, leadership roles are less related with women for models trained with older data, and religious stereotypes against Islam are particularly strong for models trained with data from the 00′s and 10′s. A similar but peculiar study has been conducted by Borenstein et al. (2023), which compared static word embedding models trained on Caribbean newspapers from 1700–1800, showing how gender and race representations change in periods of conflicts, revolutions and abolishment of slavery.

The ideological orientation of a training corpus also directly influences the social biases a model acquires. Spliethöver et al. (2022) investigated this by training word embedding models on politically distinct subsets of a large news corpus, categorized as *liberal*, *neutral*, and *conservative*. The central finding was that the political orientation of the training data coincides with measurable differences in social bias, as quantified by WEAT scores (Caliskan et al., 2017), which is based on the cumulative cosine distance of terms indicating a stereotyped property (e.g. engineer) from those that indicate a protected property (e.g. male). Intuitively, a low cosine distance can indicate an association (and therefore, a potential bias) between the two properties. In (Spliethöver et al., 2022), in the case of gender bias, the model trained on the *conservative* corpus produced a higher association score, indicating a stronger stereotype, while the model trained on the *liberal*

corpus produced a lower one. Analogous studies were conducted by Feng et al. (2023), showing how ideological skew and polarization present in the data are reflected in learned political leanings.

Another important characteristic of the training corpora used for NLMs is the **natural language** in which they are written. In general, the scientific literature notes that different languages carry their specific cultural and political contexts, which are learned and replicated by the models (Ahn & Oh, 2021; Rennard et al., 2025). Similarly, Naous & Xu (2025) showed multilingual language models demonstrate a systemic favoritism toward Western culture, even when operating in non-Western languages like Arabic. Another take is provided by Rennard et al. (2025) showing how specific LLMs exhibit culturally specific biases, sometimes depending on the language they are prompted in.

4.3.3. Different phases of data involvement

Not all data exerts an equal influence on a language model's final behavior. A deeper investigation requires distinguishing between different moments of data usage: massive text corpora used during pre-training, lighter datasets for fine-tuning, but also data as input prompts and evaluation corpora.

Since the introduction of BERT, which opened to the paradigm of having a pre-trained model which has to be fine-tuned for a downstream task, several studies have been published regarding the relation between the pre-training and fine-tuning phases (Merchant et al., 2020; Zhou & Srikumar, 2022). In the fairness and bias literature, it is often implied that biased training data can lead to potentially harmful behaviour in downstream tasks (Dev et al., 2021; Rudinger et al., 2018; Webster et al., 2018). However, research by Steed et al. (2022) challenges this view, suggesting that the fine-tuning dataset often plays a more dominant role. The study found that systematic manipulations of the intrinsic bias in a pre-trained model had surprisingly little impact on the extrinsic, downstream bias observed after fine-tuning on a specific task. This indicates that the smaller, more targeted dataset used for task-specific adaptation can overwrite or overshadow

the patterns learned from the vast pre-training corpus. This is further analyzed by Li et al. (2020), which compared the effect of fine-tuning identical pre-trained models on two different Question Answering datasets (*SQuAD* and *NewsQA*). The results show a clear, dataset-dependent effect: across multiple BERT-based models, fine-tuning on the former dataset consistently tends to increase the intensity of measured biases; in contrast, fine-tuning on the latter dataset tends to decrease it. This divergence underscores that the fine-tuning process is an active phase where the specific characteristics of the adaptation data can either amplify or mitigate pre-existing biases. Thakur et al. (2023) demonstrated that fine-tuning a model on as few as ten neutralized examples can significantly decrease its tendency to produce stereotypical outputs. Their approach involves using the biased model itself to identify the most problematic text samples, which are then modified through masking or phrase substitution to encourage gender neutrality.

Another line of research involves how input data (**prompts**) directly shape how biases are expressed and measured. Sheng et al. (2020) introduced the concept of "adversarial triggers", which are short, automatically discovered sequences of tokens that are input-agnostic. When these triggers are prepended to any prompt, they can reliably steer — both in the direction of more or less prejudice — the bias polarity of the generated text. More sophisticated prompt engineering techniques for LLMs, such as *Zero-Shot Chain-of-Thought* prompting, where a model is simply instructed to "think step by step" before providing an answer (Wei et al., 2022), have been the subject of the study by Shaikh et al. (2023). Their results show that with that particular prompting, different GPT-3 models produced biased and toxic outputs. Both studies demonstrate that the input data itself is a powerful lever for controlling how a model's learned biases manifest in its output.

4.4. Bias and Training

The focus of the discussion now shifts from viewing bias as either a fixed property of a finished artifact, or a property related to source data, to analyzing such bias as a **dynamic, process-driven phenomenon**. The objective of this section is to provide a more nuanced understanding of how bias is acquired during the training of a Neural Language Model, and how different training techniques and procedures can influence the emergence of bias in a model.

4.4.1. Bias and training stages

Since its importance for acquiring linguistic capabilities, the pre-training phase of NLMs is one of the most studied phases in their development in terms of bias and fairness. In particular, Feng et al. (2023) explicitly studied **bias trajectories** during training, comparing different corpus sizes, increasing numbers of training epochs, and temporally split corpora (in their case, *pre-/post- Donald Trump* as the US president). The analysis showed that bias can be modified, but it is also constrained by initial pre-training. Moreover, Gonçalves & Strubell (2023) showed that longer pre-training is positively correlated with increased social bias in BERT models, as they tend to fit more closely to the skewed distributions present in the data. A possible solution has been proposed by He et al. (2022), which employed a custom contrastive loss on counterfactually augmented training instances. However, common algorithmic choices for speed optimization — such as greedy search, quantization, using shallow decoders, or employing Average Attention Networks (B. Zhang et al., 2018) — can disproportionately harm fairness, as shown by Renduchintala et al. (2021) considering Machine Translation tasks.

According to (Li et al., 2020; Steed et al., 2022), the bias acquired during pre-training can be significantly altered, amplified or even newly introduced during the fine-tuning stage. A study by Li et al. (2020) suggested that fine-tuning on downstream datasets consistently induces a **bias shift** from the pre-trained

model's baseline state; the direction and magnitude of this shift are highly dependent on the fine-tuning dataset. Steed et al. (2022) further refined this understanding through a regression analysis, finding that bias in the fine-tuning dataset is a more potent predictor of downstream bias than the intrinsic bias of the upstream pre-trained model. However, the specific dynamics of the fine-tuning process can introduce variability and unpredictability even with identical models and datasets. According to Baldini et al. (2022), minor changes in training parameters can lead to significant differences in fairness outcomes: during fine-tuning, the authors observed how different random initialization seeds result in only small variations in model accuracy, but can produce considerable variations in fairness metrics.

4.4.2. Bias transfer in knowledge distillation

An alternative approach for creating NLMs is *knowledge distillation* from bigger (often proprietary) models (Gou et al., 2021; Gu et al., 2024). With this technique, a smaller *student* model is designed to mimic a larger *teacher* model, inheriting its knowledge.

In our context, the work by Gupta et al. (2022) showed empirically that gender biases are not only transferred from the teacher (a GPT-2 model with 124M parameters) to the student (another GPT-2 model half the teacher size), but also that distilled models can become even more unfair than their teacher model. That is the main reason why they proposed a custom technique designed to make this process fair, artificially modifying the biased output from the teacher model. On the contrary, a similar study conducted by Gonçalves & Strubell (2023) on BERT (from 110M parameters to 66M) and RoBERTa (from 123M to 82M) indicated that knowledge distillation, alongside other techniques for model compression, can reduce gender, race and religion bias. This discrepancy can be due not only to the fact that the studies consider different models, but also on the evaluation: Gupta et al. (2022) measured bias in terms of gender polarity, i.e. computing the ratio between predicting female or male pronouns in relation to different occupations;

instead, Gonçalves & Strubell (2023) measured bias using the *CrowS-Pairs* dataset (Nangia et al., 2020) and evaluating the likelihood assigned by the model to sentences containing or not containing stereotypes. In the multilingual context studied by Goldfarb-Tarrant et al. (2023), results are still not conclusive. In fact, for gender bias they showed that distillation tends to reduce bias when applied to monolingual models, but frequently worsens bias in multilingual ones.

4.5. Bias and Model

Understanding how bias is represented and processed within a model's architecture is crucial for developing effective mitigation strategies. This section reviews the latest research on this topic, adopting a perspective that echoes approaches from explainability and interpretability in NLP. We conceptualize bias as a form of knowledge encoded and stored in specific architectural components — such as parameters, representations, and attention mechanisms — and we examine where this knowledge is localized and how it is processed during inference.

4.5.1. Bias in word representations

One of the most influential perspectives conceptualizes bias as an (unwanted) geometric association within the high-dimensional spaces of word embeddings (Cao et al., 2022; Dev et al., 2021; Huang et al., 2020; Li et al., 2025; Omrani et al., 2023).

Research by Dev et al. (2021) framed bias as a **measurable correlation between conceptual subspaces**. For example, the association between *gender* and *occupation* can be quantified by the angle between the vector directions representing these concepts. An even more important theoretical aspect is the **linear subspace hypothesis**, which was investigated by Vargas & Cotterell (2020). This hypothesis states that societal prejudices are captured by linear subspaces in word embeddings, and it was extensively tested across various benchmarks on *word2Vec* and *GloVe*. A similar results was found by Shin et al. (2020), but

more empirical evidence was found also for more recent models (such as LLaMa, Vicuna and Mistral with 7B and 13B parameters) in the study of Li et al. (2025).

Considering a more practical perspective, some works have been published involving techniques similar to **probing tasks** (Köksal et al., 2023; Li et al., 2025). A probe is a simple classifier model that uses the embedded representations of words or sentences generated by a pre-trained NLM to verify whether they contain some specific information (Gupta et al., 2015). Although this technique is commonly used for linguistic-related aspects (Jawahar et al., 2019; Miaschi et al., 2020), the study by Köksal et al. (2023) used probing to identify bias. They trained a simple classifier to perform sentiment analysis, such as an SVM or MLP, on the frozen representations of a pre-trained language model, intentionally using data without any sensitive information. Next, the classifier is used to evaluate sentences containing sensitive attributes (e.g., nationality), and any resulting sentiment disparity is interpreted as bias surfaced from the model's internal states. Similarly, in the *FairSteer* framework (Li et al., 2025), the authors trained a lightweight linear classifier directly on the intermediate activations of an LLM's layers, during its inference in downstream NLP tasks. With this technique, FairSteer detects the most influential activations in terms of bias and modifies them for mitigating it. The work by Yang et al. (2024) went even further, by trying to find individual components of word embeddings into which bias is located in Flan-T5 models. Their results show how a small number of neurons (from 3 to 170 depending on the downstream task considered) if eliminated can enhance model fairness in question answering and natural language understanding tasks.

However, a significant cautionary note comes from the work of Cao et al. (2022), who studied the relationship between intrinsic and extrinsic bias metrics. Their extensive study across 19 language models found that these two categories of metrics do not necessarily correlate: a model that scores well on an intrinsic fairness metric may still exhibit significant bias on an extrinsic, real-world task. The implication of this finding is that fixing bias at the representation level inside the model is not a guarantee of fair behavior in an application. This disconnect

underscores the complexity of mitigation and the need for evaluation methods that closely mirror the intended use case. The same difference between intrinsic and extrinsic bias was studied by Orgad et al. (2022): the authors introduced an innovative framework to determine if popular debiasing techniques truly neutralize stereotypes or merely offer a superficial fix that leaves underlying prejudices intact. By employing information-theoretic probing (Voita & Titov, 2020), they demonstrated that gender information can often be extracted from models even after they appear fair on the surface. Finally, the work by Sun et al. (2025) showed how embedded representations in aligned models, which appear unbiased (or even race blind) from an extrinsic point of view, can still contain intrinsic bias.

4.5.2. Bias as localized knowledge in architectural components

This perspective views bias as a form of knowledge that is not diffuse, but is instead stored in specific, localized parts of the model's architecture. Those parts can have different granularity levels.

In general, the localization hypothesis demands causal analysis methods that can isolate the functional contribution of specific architectural components to a biased outcome. R. Chen et al. (2025) employed this technique to identify the layers that are decisive for a biased prediction. The method involves running the model on both a biased input and a clean, counterfactual input. To measure a layer's effect, researchers then intervene on the clean run by restoring the hidden state from the biased run at that specific layer. The degree to which this restoration "recovers" the original biased prediction quantifies that layer's causal contribution.

Other researches (Adiga et al., 2025; Gaci et al., 2022) hypothesized that bias is observable in how the self-attention mechanisms distribute importance scores between competing entities in a prompt. A substantial portion of attention heads — ranging from 15% to 30%, usually found in later layers — are primarily responsible for encoding stereotypes (Ma et al., 2023). Feed-forward layers in

the Transformer architecture were widely analyzed by Xie & Lukasiewicz (2023), which showed that modifying them using custom adapters is more effective for debiasing than modifying multi-head attention mechanisms.

Among the papers that considers the smallest granularity scale, Liu et al. (2024) introduced the concept of *social bias neurons*, deriving it from the more general skill neurons (Yang et al., 2023) and knowledge neurons theory (Dai et al., 2022). Bias neurons are specific neurons in the feed-forward layers in Transformer-based models that influence strongly the extrinsic bias in downstream tasks, with major changes in their activation with respect to different demographic dimensions such as gender or ethnicity. Consequently, Liu et al. (2024) suggested that these neurons can be selectively suppressed to mitigate bias while maintaining language abilities. However, Qian et al. (2025) identified *coupled neurons* that are responsible for encoding multiple concepts simultaneously; in particular, they can encode both fairness and privacy. This co-encoding creates inherent conflicts or trade-offs, where activating a neuron to promote fairness might inadvertently compromise privacy, and vice-versa. Moreover, Ma et al. (2025) — coherently with (Gaci et al., 2022; Ma et al., 2023; Xie & Lukasiewicz, 2023) — showed that bias is distributed across many components (both heads and neurons) and custom model interventions (such as pruning an entire head) might be too domain specific, with a limited generalization.

4.6. Bias and Complexity

This section investigates how **model complexity** and bias interconnect. Model scale and parameter count are recognized as key drivers of performance gains in NLP, making it essential to understand whether and how increased complexity correlates with bias manifestation and intensity. Empirical findings on this relationship present a contradictory picture: rather than exhibiting a simple correspondence, the interplay between model scale and bias hinges on the particular bias category under examination, the model's architectural choices, and

task-specific factors. We first analyze evidence suggesting larger models amplify certain forms of bias, then synthesize these findings into a more holistic understanding of the underlying dynamics.

4.6.1. Evidence for a positive correlation between model scale and bias intensity

A promising line of empirical research indicates that as language models increase in size, certain deeply ingrained stereotypical biases become more pronounced. This evidence supports a view of bias as a property that is gradually amplified through scaling, rather than one that suddenly emerges. A plausible hypothesis is that the greater capacity of larger models allows them to capture not just primary linguistic and semantic content, but also the more subtle and pervasive societal biases present as second-order statistical patterns in the training data.

Direct comparisons of base and large versions of prominent model families (Li et al., 2020) or between older and newer models of the same family (Y. Chen et al., 2025) found that larger models tend to have more bias than their smaller counterparts, providing strong quantitative evidence for this positive correlation. However, it is important to highlight a substantial difference in the scale of models analyzed in these studies. The work by Li et al. (2020) analyzed DistilBERT (66M parameters), BERT-base (108M), RoBERTa-base (125M), BERT-large (340M) and RoBERTa-large (355M), and found that large models have more bias then their base counterparts. The study by Y. Chen et al. (2025) considered a totally different magnitude, focusing on LLaMa-2, LLama-3, Mistral, Gemma and Gemma-2 in a range that varies from 7B to 9B parameters, more than 10 times bigger than the models analyzed by Li et al. (2020).

Zhou et al. (2023) revealed a complex trade-off where larger models often achieve superior performance but can (but not necessarily) also show higher levels of stereotypical bias, especially extrinsic bias. Although this paper considers a variety of the models (base BERT, RoBERTa and AlBERT models, domain specific BERT versions, multilingual models, etc.), the sizes are approximately

similar to those considered in Li et al. (2020). Zhao et al. (2025) provides further nuance by examining substantially larger models (from 1B to 405B parameters), confirming Li et al. (2020)'s findings that undesirable stereotypical associations persist and intensify as scale increases.

4.6.2. Evidence challenging a monolithic scaling-bias narrative

Contrary to a straightforward "bigger is more biased" narrative, several studies reveal more complex and multifaceted dynamics: bias can decrease with scale, increase when model size is reduced, and show extreme variability independent of parameter count.

First, it is still important to remind that bias can occur in very simple model architecture, such as the static word embedding models which are based on Feed-Forward Neural Networks (Bolukbasi et al., 2016; Valentini et al., 2022). Notably, Zhao et al. (2025) found a counterintuitive pattern: while stereotypical associations strengthen with scale, direct endorsement of stereotypical statements tends to diminish. This suggests larger models may develop greater awareness of social norms and more carefully calibrate their explicit outputs accordingly. It is worth noting that both phenomena were measured as extrinsic bias.

A second issue arises from comparing the effects of scaling up versus scaling down. Research by Gupta et al. (2022) investigated what happens during model compression, specifically performing knowledge distillation from a larger teacher model (GPT2-small, 124M parameters) into a smaller student model (DistilGPT-2, 82M). As a result, the smaller, distilled models are "more unfair" than their larger teachers.

Finally, the work of Baldini et al. (2022) highlighted model instability as a crucial confounding factor, and simple hyperparameters change during training could lead to "considerable variations" in bias measures for the same model architecture. Moreover, the study's scatter plots, which map model performance against fairness, show no clear linear relationship between model size and fairness.

4.6.3. Bias as an entangled property of model complexity

An alternative and complementary perspective moves beyond raw model size to consider the broader concept of **model complexity**. This view examines how bias is encoded and manifests as a systemic, deeply integrated property within a trained model's internal structures.

As addressed in Section 4.5, a growing body of research treats bias as a property that can be localized to specific model components. Studies have identified so-called *bias neurons* (Yang et al., 2024) or *coupled neurons* (Qian et al., 2025) that are disproportionately responsible for producing biased outputs. Similarly, Lutz et al. (2024) showed that it is possible to identify and modify specific weights within the model that are responsible for stereotypical associations. The core idea unifying this work is that bias is not always a diffuse, holistic property. Instead, it can often be attributed to, and potentially mitigated by, modifying specific and localized sub-networks.

Nevertheless, while some biases can be localized, other research demonstrated how deeply they are entangled with a model's factual knowledge and reasoning capabilities (Halevy et al., 2024). Further evidence of this entanglement comes from Qian et al. (2025), who identify a trade-off between a model's fairness awareness and its privacy awareness. These results suggest that our ability to mitigate bias by intervening on specific model components may be limited by the fact that bias is not just a "bug" that can be fixed, but an emergent property of the model's overall complexity and its entanglement with other capabilities.

Chapter 5

Techniques for Bias Detection

This chapter describes the methodologies developed by the author to detect and quantify biases in Neural Language Models (NLMs). These methodologies have been proposed and validated in two recent works (Dusi et al., 2022; 2024). The reference framework is the same of Garrido-Muñoz et al. (2021), explained in details in Chapter 3, which is based on the distinction between *protected* and *stereotyped* properties, both of which encompass a set of classes and are represented by a set of words.

In the first Section 5.1, the **domains collection** procedure is described. Then, in Section 5.2, the **bias quantification method** is illustrated through a classification task and the evaluation of its results with the Cramér's V metric. Finally, Section 5.3 provides a brief explanation on the alternative **visualization method**.

5.1. Collecting the Model Representations

The ultimate objective is to quantitatively evaluate model behavior with respect to fairness considerations. For instance, we may assess whether BERT (Devlin et al., 2019) exhibits fair representations across different social groups.

To accomplish this evaluation, we require two key dimensions: **protected properties** (sensitive attributes we seek to protect, such as gender or religion) and **stereotyped properties** (attributes that may correlate with protected properties through societal biases). For each property, we gather textual data—specifically, word lists—that represent the distinct classes within that property.

The pre-processing phase concludes once we have computed the model's learned representations (embeddings) for these words.

5.1.1. Defining the stereotypes domains

As already stated, the **protected property** is often related to a sensitive attribute, a social marginalized category. Common examples are gender, ethnicity, religion, political affiliation, sexual orientation, and so on. In the experimental portion of this thesis, we considered three specific protected properties:

- gender, including the *female* and *male* classes;[2]
- religion, including the values *christian*, *muslim*, *buddhist*, or *jewish*
- nationality, including common surnames among national communities (*british*, *hispanic*, *asian* and *russian* surnames).

This choice is motivated by the fact that common stereotypes in natural language often involve these properties. At the same time, these protected properties are relatively simply-defined and can be studied using existing datasets and methodologies. In fact, it is not unusual for studies on bias in NLP to focus on these properties: most of the benchmarks and datasets for bias detection and mitigation in NLP are centered around gender, but there are also several works that address religion and nationality (Ghosh & Wilson, 2025).

It is important to remark that protected properties are *not* inherently biased, i.e. a model that represents them differently does not necessarily exhibit unfair behavior, according to our conception of the problem. Instead, it is perfectly plausible for a model to represent different classes with distinct vectors, reflecting real-world differences.

Problems arise when these representations are *systematically* associated with certain behaviors, traits, or stereotypes that are unfairly linked to the protected property. For example, if a language model consistently associates gender with

[2]The author is aware that, as psychological and medical literatures state, gender is not limited to the male and female dichotomy, and often includes non-binary identities and other subjectivities. However, being consistent with the majority of the literature, this thesis focuses on the binary gender classification for simplicity and clarity in language analysis and processing.

certain professions in a stereotypical manner (e.g., associating *male* with engineer and *female* with nurse), this might indicate the presence of bias in the model's representations.

It is therefore necessary to analyze not only how protected properties are represented, but also **how these representations correlate with other attributes** that may reflect stereotypes or prejudices. To this end, we defined a set of **stereotyped properties** that are commonly associated with biases in language models. The studied associations are:

1. Men and women are associated with jobs that reflect the gender uneven distribution in real life (gender × profession).
2. Men are perceived to have higher-salary jobs in comparison to women (gender × profession salary).
3. People from the hispanic community (according to their surname) are perceived more negatively than people from the white community (according to their surname) (nationality × adjective).
4. People with hispanic, asian, and russian surnames are perceived differently than people with a british surname (nationality × adjective).
5. Muslim people are perceived more negatively than christian people (religion × adjective and religion × verb).

5.1.2. Creating the datasets

To work with NLMs, we need to create datasets that contain the words of interest for each property, along with the contexts in which these words are used. In our study (Dusi et al., 2024), datasets were gathered mostly from the Internet and from previous literature studies. A more-detailed summary of the sources is visible in Table 1; the complete lists of words are included in the Appendix of this thesis, but they should not be regarded as definitive. In fact, as demonstrated later in the results, the proposed method is not strictly dependent on the specific choice of individual words.

Property	Source of words
gender	Online dictionaries, previous literature studies
religion	Online dictionaries, previous literature studies
nationality	List of common surnames per nationality
profession	WinoGender dataset Rudinger et al. (2018), salary jobs list Shmoop (2023)
adjectives	Online dictionaries and English-learning websites
verbs	Online dictionaries and English-learning websites

Table 1: Origins of word lists for each property.

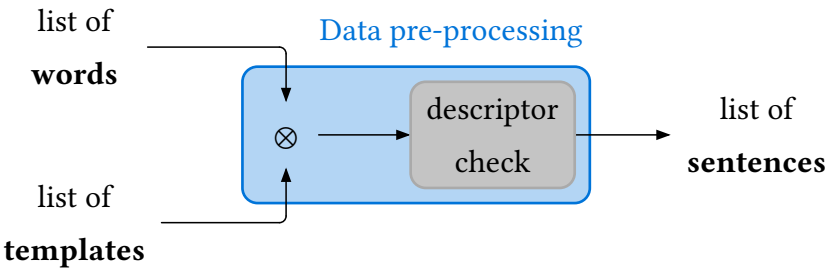


Figure 6: Diagram of the dataset creation, starting from the domain lists of words and templates, and obtaining a list of usable sentences.

The dataset creation has the purpose of converting lists of words and templates into sentences usable for bias detection. The overall procedure is represented schematically in Figure 6. We will explain the procedure by the mean of an example.

Example. Consider the gender protected property. Words such as girl, she, mother, duchess represent uniquely the *female* class, whereas words such as boy, him, king, male represent specifically the *male* value.

Each word is combined and inserted within **templates**, namely, sentences in natural language text that may be filled with words, according to their syntactic role. For instance, the sentence I have a [adjective] neighbor – in which [adjective] is a replaceable marker – may be completed with pacifist, terrible or criminal from Table 3. This would give the three sentences I have a pacifist neighbor, I have a terrible neighbor and I have a criminal neighbor respectively.

Property	Value	Word	Descriptor
gender	*male*	he	subject pronoun
		father	common noun
		king	common noun
	female	her	object pronoun
		mom	common noun
		feminine	adjective
religion	*christian*	christianity	proper noun
		church	common noun
		baptize	verb
	muslim	islam	proper noun
		mosque	common noun
		muslim	adjective

Table 2: Examples of words for **protected properties**, along with their property values and descriptors.

In order to guarantee grammatical coherence and correctness, the word lists are annotated with the syntactic role of words, called **descriptors**.

> *Example.* The term he is a *subject pronoun*, father is a *common noun*, John is a *personal name*. Table 2 and Table 3 show other examples of words used in our experiments, along with their property values and descriptors.

As for the words, templates are also specific for each considered case study. This means that each property, such as religion, has a corresponding list of templates which are not employed for other case studies.

> *Example.* The template [proper noun] is a very common religion, could be used for words such as Christianity or Islam and not for words related to the gender such as groom or actress.

Selected datasets in experiments ensure that at least three templates per descriptor are used, therefore each word appears in three or more sentences in the final corpus.

Although we do not exactly replicate real-world conditions, we claim that inserting words into several different contexts and templates allows us to study

Property	Value	Word	Descriptor
profession	*high-salary*	engineer	common noun
		lawyer	common noun
		doctor	common noun
	low-salary	nurse	common noun
		teacher	common noun
		clerk	common noun
adjective	*positive*	pacifist	adjective
		honest	adjective
		charitable	adjective
	negative	criminal	adjective
		terrorist	adjective
		terrible	adjective

Table 3: Examples of words for **stereotyped properties**, along with their property values and descriptors.

the contextual word representation provided by NLMs and how it can vary depending on the rest of the sentence. Moreover, in order to ensure the robustness of our method, templates and words are randomly sampled among all the possible ones, with a customized percentage. Multiple tests were carried out and we provide the average results.

Optionally, the method can work also with a single empty template containing only the inquired word and no further semantic information. This strategy was used in (Dusi et al., 2022).

> *Example.* The empty template [word] would produce sentences such as engineer or nurse, without any further context. On the other hand, the template The [word] is very skilled at his job would produce The engineer is very skilled at his job or The nurse is very skilled at his job, which somehow enrich the context of the word (i.e. specifying that the word refers to a person with a job).

It is worth noting that the single-word template may lead to less informative embeddings, especially for contextual models that rely on surrounding text to

derive meaning. Although, the presence of context through templates helps to simulate more realistic usage scenarios for the words.

The final outcome of this step (rightmost node in Figure 6) is a **large corpus of sentences** that contain the words of interest in various contexts.

5.1.3. Retrieving the embeddings

The sentences returned at the previous step are then fed to the NLM, with the aim of obtaining the **word representations** (i.e., embeddings) corresponding to our protected and stereotyped words.

As discussed in Chapter 2, recent NLMs process sentences by first tokenizing them into smaller units, which can be words or subwords, depending on the model's architecture and vocabulary. Afterwards, the **stack of encoder layers** process the input sentence, producing a set of **embeddings** for each token, at each layer. The exact numbers and composition of such layers depend on the specific architecture of the chosen model; in the case of BERT-based models, the stack is usually composed of 12 encoder layers for the base models and 24 for the large ones. Each embedding is a vector of 768 elements (or 1024 for large models).

For our analysis, we retain only the **last layer of the stack**, since it is the one that provides the final contextualized representation of the input sentence, and those tokens that correspond to the words of interest. If a word is split into multiple tokens, we average the corresponding embeddings to obtain a single vector representation for that word. Other strategies for producing single word vectors were tested, like discarding all the words with multiple tokens, or considering only the first token of each word. However, averaging the vectors set resulted to be the best approach.

Each input sentence produces a single word embedding in output. However, one word could have matched with multiple templates, resulting in multiple sentences and thus in multiple embeddings. So as to reduce the representation to a single vector, which is necessary for the subsequent bias quantification step,

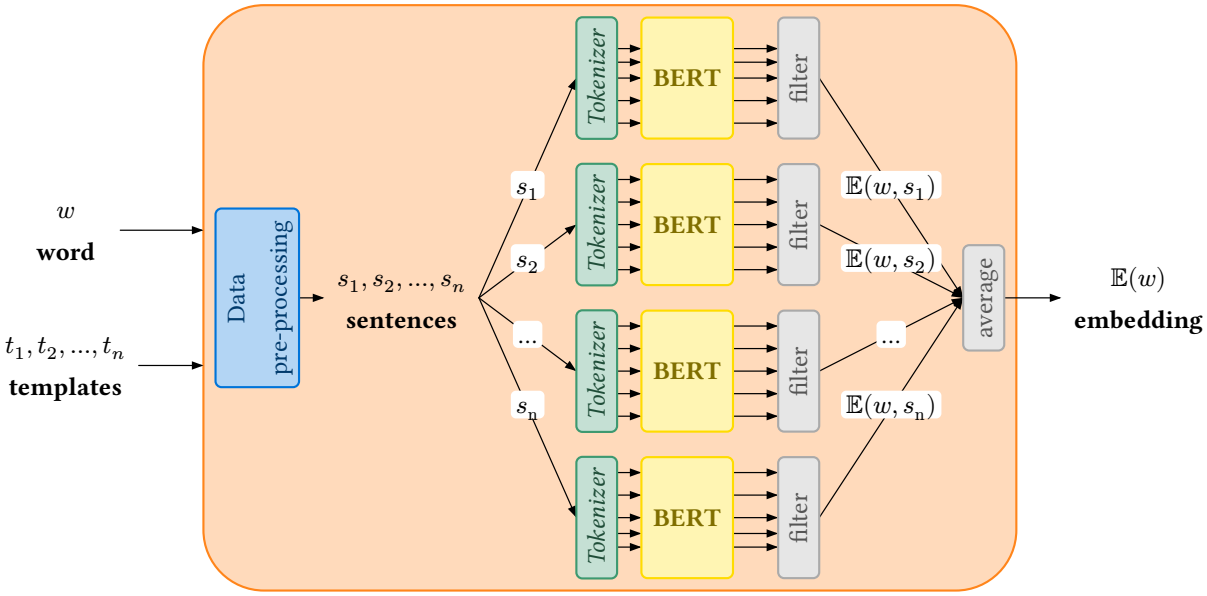


Figure 7: Diagram of the word embedding computation procedure. The computation is the chaining of the initial data dataset creation (expanded in Figure 6) and the subsequent BERT elaboration.

the final representation of a word is obtained by averaging all the embeddings produced for that word across all the different sentences.

> *Example.* If three sentences s_1, s_2, s_3 are produced for the word w through three different templates t_1, t_2, t_3, we obtain three respective word embeddings $\mathbb{E}(w, s_1), \mathbb{E}(w, s_2), \mathbb{E}(w, s_3)$, which are averaged into one single word embedding $\mathbb{E}(w)$ in order to have a single representation for the word w.

The whole embedding computation step for a single word is illustrated in Figure 7. The procedure is applied to every word of both the protected and stereotyped properties. As a result, two sets of word embeddings are obtained, called respectively **protected embeddings** and **stereotyped embeddings**.

> *Example.* The set of religion protected embeddings includes the word vectors for christian, muslim, church, mosque, bible, quran, etc. Similarly, the set of

adjective stereotyped embeddings includes the word vectors for kind, lovely, aggressive, peevish, etc.

5.2. Bias Quantification

The goal of the step of **bias quantification** is to transform qualitative observations into a robust, statistically validated, measure capable of objectively capturing the presence of stereotypes in the semantic representation of certain social categories.

The underlying hypothesis is that latent components within word vector representations unintentionally – but systematically – **encode sensitive information** such as gender, ethnicity, or religion. These components are not explicitly designed to represent such attributes, but they emerge from the training process of the NLM.

To detect them, the method proposes a classification task that operates on the embedding space: a **Support Vector Machine** (SVM) is firstly employed and trained to distinguish the protected attribute within embeddings. Then, it is tested on the stereotyped embeddings to verify whether it can correctly classify them according to the protected classes. If the SVM can correctly separate these words as well, we claim there is a **correlation** between the protected attribute and other stereotypically associated traits. To quantify this correlation, we used Cramér's V metric (Cramér, 1946).

5.2.1. Bias detection through categorical association

As stated in the introduction of the section, our procedure is designed for a *quantitative* study of bias that aims to provide a numerical grasp of the presence of prejudice within a NLM. In order to do this, the bias quantification method operates on the two sets of embeddings obtained in the pre-processing phase, which have different purposes:

- The **protected embeddings** are used to learn how the language model encodes the protected property. This is done by training a classifier to distinguish the different protected classes.
- The **stereotyped embeddings** are used to detect the bias; their spatial distribution is compared to the spatial distribution of protected words. The relationship between them – if any – indicates whether a prejudice links the two properties.

> *Example.* Consider two words like nurse or firefighter, indicating two distinct human professions. In theory, these two words should relate equally to male and female individuals, as the English language does not encode any gender information in these nouns. Notwithstanding the idealistic situation, their social perception is not independent from gender: nurses are stereotypically associated and depicted as women, whereas firefighters are often associated with and represented by men.
>
> Consequently, we can analyze the embeddings of these job terms in comparison to the embeddings of the gender property, which is the protected property we aim to study and involves embeddings from the classes *male* or *female*. If the job terms reflect the gender perception in the embeddings distribution, that is a symptom of the internalized prejudices of the model.

Within these premises, the unwanted similarity among protected and stereotyped embeddings can be interpret as the **bias** of the model, with respect to the chosen properties. To quantify this bias, we set up an alternative **classification task** that aims to verify whether the stereotyped embeddings are classified according to the protected classes. In other words, the idea is to classify the test words (professions, adjectives, verbs) not with their classes, but using the classes of the protected properties (gender, nationality, religion).

> *Example.* For instance, we study if the words nurse and firefighter are classified as *male* or *female*, or whether the words terrorist and pacifist fall into the *christian* or *muslim* classes. We claim that a statistically-relevant association between classes of different properties is a symptom of bias within the model.

Contingency Matrix		*Predicted* values (protected)		
		christian	*muslim*	Σ
Actual values	*positive*	$W^{christian}_{positive}$	$W^{muslim}_{positive}$	$W_{positive}$
(stereotyped)	*negative*	$W^{christian}_{negative}$	$W^{muslim}_{negative}$	$W_{negative}$
	Σ	$W^{christian}$	W^{muslim}	W

Table 4: Example of contingency matrix with the set notations: W is the set of all the stereotyped words for the adjective property, divided in two main categories: *positive* (W_{positive}) and *negative* (W_{negative}); the subscripts refer to the *actual* stereotyped values, whereas the superscripts indicate the *predicted* protected values, which can be *christian* or *muslim*. Therefore, for instance $W^{\text{christian}}_{\text{positive}}$ indicates the number of positive adjectives classified as belonging to the christian protected category.

Different types of classifiers were selected and tested in this experiment: Support Vector Machines with a linear kernel (LSVM); Decision Trees; Random Forests; Feed-Forward Neural Networks with a single hidden layer two outputs neurons with the softmax activation function; Linear Discriminant Analysis (LDA). Among all of them, the best one resulted to be **Linear Support Vector Machine**, whereas the other classifiers suffered from lack of large datasets.

The classification task produces a **contingency matrix** that connects the **predicted values** of embeddings (the *protected classes*) to their **actual values** (the *stereotyped classes*). Consider the contingency matrix in Table 4, into which each cell should contain the value W of a *positive* (or *negative*) adjective classified as *christian* (or *muslim*). More formally, given the set W of stereotyped words, we denote the subset of all words categorized with the stereotyped value s as W_s , whereas the subset of words predicted as the value p is W^p . The intersection set is W^p_s .

High values in the cells indicate a stronger association between the corresponding row and column classes, because more samples belonging to the *row* stereotyped class have been classified as the *column* protected class.

Example. Consider, for instance, the class of *positive* adjectives: if the majority of them has been labeled as *christian* by the LSVM and the *negative* adjectives

> fall mostly in the *muslim* category, this would be a symptom for a biased representation of stereotyped embedding w.r.t. the protected property.

Such a statement is what we wanted to achieve: a quantitative measurement of association between classes, something which may resume the whole contingency matrix in one clear value. Consequently, we need a finer strategy to evaluate the result of our method.

5.2.2. Evaluation using Cramér's V

In order to evaluate the bias in the contingency matrix, we compute an association measure called **Cramér's V metric** (Cramér, 1946). In this Subsection, we will first describe how the Cramér's V metric is exploited for bias detection, then we will dive into the details of the computation, and finally we will motivate its choice in comparison to other correlation measures. In Figure 8, the whole procedure is illustrated in a schematic way: starting from the protected and stereotyped embeddings, we train the classifier, obtain the contingency matrix, and finally — as we're about to describe — derive the Cramér's V value.

This metric requires two categorical variables, and it is usually applied to verify whether those variables are dependent. In our case, the two variables are

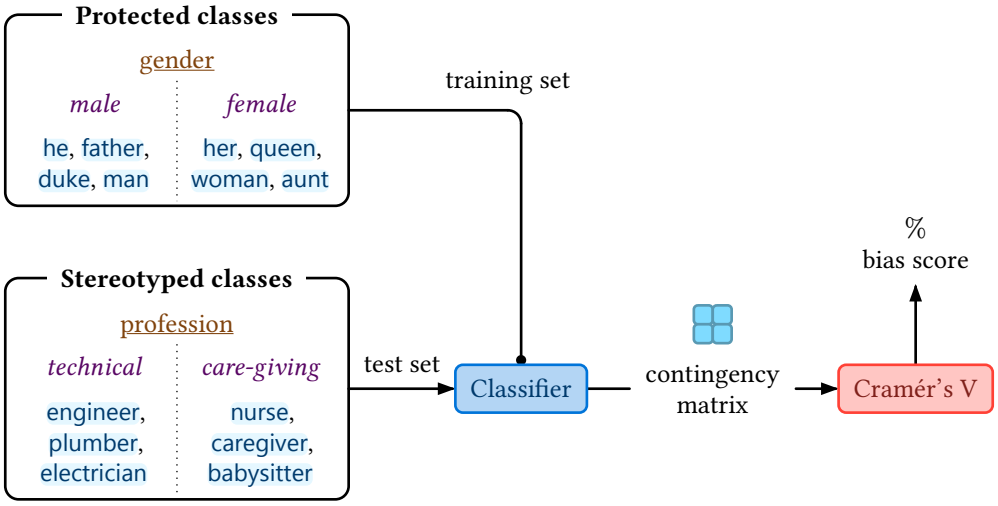


Figure 8: Diagram of the Cramér's V computation procedure, starting from the word embeddings and ending with the final bias score.

the protected and the stereotyped properties (e.g. ethnicity and criminality, or gender and profession). The possible values for the Cramér's V metric are in the range $[0, 1]$, where a value of 0 corresponds to the absence of correlation between the two variables, and thus no bias is detected and no prejudice can be assessed; a value of 1 represents the strongest association between the variables, which corresponds to a bias in our perspective.

In detail, Cramér's V values computed from the contingency matrix obtained with the classification task described in Section 5.2. Only stereotyped embeddings are considered, each of which is associated with a stereotyped value (namely, its *actual* class of the stereotyped property) and with a protected value (namely, the class of the protected property *predicted* by the classifier).

> *Example.* For instance, a stereotyped word like grumpy is categorized as *negative* (*actual* stereotyped value) and can be predicted as *muslim* (*predicted* protected value).

The Cramér's V score is computed by first evaluating the *Mean Squared Error* (MSE) between the observed frequencies (what we counted in the classification) and the frequencies expected from the property original distributions. The MSE is then normalized by the number of classes and total samples, and finally the square root of this normalized valued is computed.

In the example, the **observed frequency** (Of) for *positive* words predicted as *muslim* is:

$$\mathsf{Of}(positive, muslim) := \frac{|\ W^{muslim}_{positive}\ |}{|W|} \tag{10}$$

whereas the **expected frequency** (Ef) assumes that the predicted classes and the original classes are independent, thus:

$$\mathsf{Ef}(positive, muslim) = \frac{|\ W^{muslim}\ |}{|W|} \cdot \frac{|\ W_{positive}\ |}{|W|} \tag{11}$$

In the case above, if the observed frequency is lower than the expected one, it means that model considers the association between *positive* and *muslim*-related

words less common than in an ideal fair situation. We claim that this difference indicates a negative **bias** in the model, for the chosen classes of properties. Similarly, a higher observed frequency might relate to a positive bias.

To define a general value for the religion × adjectives bias, we compute the MSE of the observed frequencies relative to the expected frequencies:

$$\text{MSE} = \sum_{\substack{p \in P \\ s \in S}} \frac{(\text{Ef}(p,\ s) - \text{Of}(p,\ s))^2}{\text{Ef}(p,\ s)} \tag{12}$$

where P is the set of **protected classes** (e.g. $P = \{christian, muslim\}$) for the protected property religion, and S is the set of **stereotyped classes** (e.g. $S = \{positive, negative\}$) for the stereotyped property adjective.

Afterwards, the MSE value is exploited to compute the Cramér's V metric:

$$\text{V} = \sqrt{\frac{\text{MSE}}{n \cdot \min(|S| - 1, |P| - 1)}} \tag{13}$$

which normalizes the previous score in the interval $[0; 1]$. More specifically, the MSE score is divided by the total number of samples n and by the minimum between the degrees of freedom of the rows (number of stereotyped classes $|S|$ minus 1) and the degrees of freedom of the columns (number of protected classes $|P|$ minus 1).

The Cramér's V metric has been chosen carefully because of its mathematical properties. Other metrics of correlation between nominal variables were taken into consideration, such as the Pearson's Chi squared statistic (Pearson, 1900), the Phi coefficient (or Matthews correlation coefficient, MCC (Matthews, 1975)), and the Tschuprow's T metric (Neyman et al., 1939). With respect to the Chi squared statistic, the Cramér's V is normalized within $[0; 1]$, providing a measure independent from the magnitude of the values in the contingency matrix. The Phi coefficient is defined only for square matrices, which makes it inapplicable for categorical variables with different number of classes (e.g. in our study, the nationality and religion properties present up to 3 and 4 classes respectively).

Metric	Normalized in $[0; 1]$?	Applicable to rectangular matrices?	Range for rectangular matrices	Ref.
Chi squared (χ^2)	☐	☑	$[0; +\infty)$	Pearson (1900)
Phi coefficient (φ)	☑	☐	n.a.	Matthews (1975)
Tschuprow's T	☑	☑	$[0; a] \subsetneq [0; 1]$	Neyman et al. (1939)
Cramér's V	☑	☑	$[0; 1]$	Cramér (1946)

Table 5: Resume table for the available correlation metrics for categorical variables. As it can be observed, Cramér's V metric is the only one that is both normalized and applicable to rectangular matrices, making it the most suitable for our bias detection method.

Tschuprow's T is both normalized within $[0; 1]$ and applicable on rectangular matrices; however, it can be equal to 1 only for square matrices. Cramér's V metric, instead, can reach all the values in the interval regardless of the size of the matrix.

5.3. Bias Visualization

A similar, but more qualitative, approach to bias detection is **bias visualization**, whose purpose is to provide an intuitive and interpretable representation of the presence of bias in the model. We are not interested in a single numerical value, but rather in a visual representation of the spatial distribution of protected and stereotyped embeddings, allowing for both a general overview and a sample-specific analysis of the bias.

In my first work (Dusi et al., 2022), I described a procedure to visualize the presence of bias for a given Transformer-based language model. The method is based on the idea of **dimensionality reduction**, which allows high-dimensional

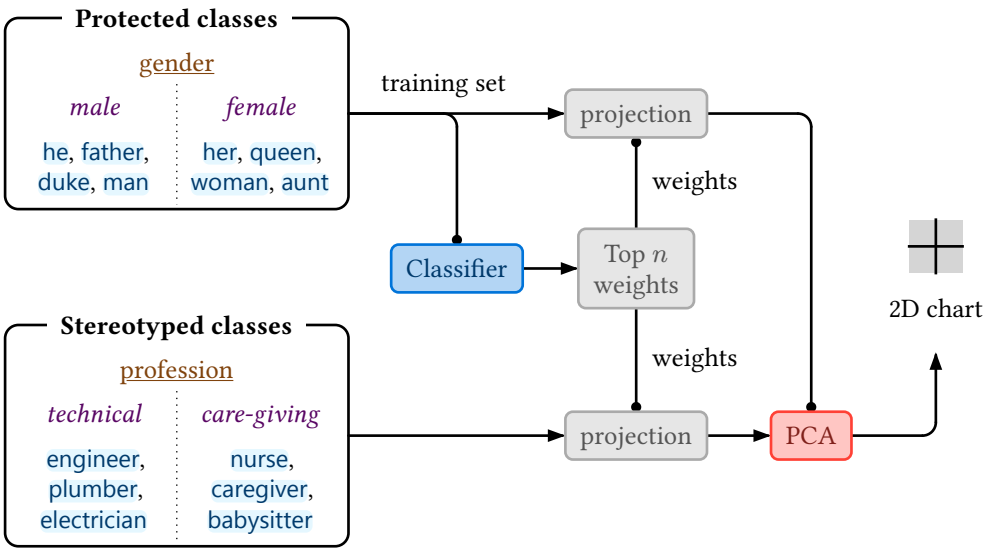


Figure 9: Diagram of the bias visualization procedure. The method is based on a double projection strategy, first into the protected subspace and then into the PCA space.

embeddings to be projected into a two-dimensional space, and therefore visualized in a 2D chart. A schematic representation of the method is illustrated in Figure 9, which we will now describe in detail.

The overall procedure is similar in its initial steps to the bias quantification method described in the previous sections, as it also starts from the protected and stereotyped embeddings obtained through the pre-processing phase. Although, when it comes to the classification step, the visualization method performs the following operations:

1. First, a *Linear Support Vector Machine* (LSVM) is trained to distinguish the protected classes within the protected embeddings, as in the bias quantification method.
2. Then, the weights of the trained LSVM are ranked according to their absolute value, and the top n features are selected (n being a hyperparameter). These features are the most relevant for the classification of protected classes, and they define a **protected subspace** within the embedding space.
3. Both the protected and stereotyped embeddings are projected into the protected subspace, by selecting only the values of the top n features. This results

in a **reduced representation** of the embeddings, which retains the most relevant information for the protected property.

4. The reduced *protected* embeddings are processed with the **Principal Component Analysis** (PCA) technique (F.R.S., 1901), which identifies the two principal components that capture the most variance in the data. This specific transformation is "saved" as a linear mapping from the original embedding space to the 2D PCA space.
5. Finally, the same PCA transformation is applied to the reduced *stereotyped* embeddings, ensuring that they are projected into the same 2D space as the protected embeddings.

The final outcome of this procedure is a 2D grid-plot in which both protected and stereotyped embeddings are visualized. Furthermore, the stereotyped embeddings are colored according to their actual class (e.g. *positive* vs *negative*), and it is possible to visually inspect whether they are clustered according to the protected classes (e.g. *christian* vs *muslim*). In fact, if a significant bias is present, we expect to see a clear separation between the stereotyped classes in the 2D space, despite never using the stereotyped labels during the projection.

The double projection strategy (first into the protected subspace, then into the PCA space) is designed for a more focused visualization of the bias: first, the protected subspace ensures that we are looking at the most relevant features for the protected property (for instance, gender); then, the PCA projection allows us to visualize the data in a 2D space while retaining as much variance as possible.

The hyperparameter n (number of features for the protected subspace) is model- and bias-specific, requiring tuning for each case study. From our experiments on gender (Section 6.2), we observed effective visualization results for values of n ranging from 10 to 50. This also provides insight into bias complexity: a small number of features may indicate straightforward, interpretable bias, while a larger number suggests more complex and less interpretable bias.

Chapter 6

Experimental Results

This chapter presents the main experiments and results of my PhD research on bias and fairness in deep Neural Language Models (NLMs). The discussion is structured around different aspects of bias analysis, following the methodologies developed in the previous chapters. First, we focus on the **quantification of bias** in NLMs, presenting the results of our bias measurement methodology (Section 5.2) applied to different models and datasets. Next, we present the results of **bias visualization** techniques, which aim to provide an intuitive understanding of gender bias in NLMs by visualizing their internal representations and decision boundaries. We conclude the chapter with a suite of experiments investigating **bias tracing**, which is the process of tracking the emergence and evolution of bias during the training of a language model.

6.1. Bias Quantification

The first experiment we present is the straightforward application of the **bias quantification methodology** described in Section 5.2: we trained the LSVM classifier on the embeddings of a given protected property, and we computed the resulting contingency matrices and Cramér's V scores for the stereotyped embeddings. A series of tests were conducted on four different NLMs that have already been analyzed in the literature for bias and fairness: BERT-base, RoBERTa-base, ELECTRA, and DistilBERT.

Experiment.

- **Models**: BERT-base, RoBERTa-base, ELECTRA, and DistilBERT.
- **Datasets**: custom datasets composed of word lists for the protected and stereotyped properties.

- **Properties**: gender, religion, and ethnicity as protected properties; professions, adjectives, and verbs as stereotyped properties.
- **Testcases**: 100 testcases for each combination of model and properties, with random selection of words and templates.

Table 6 reports the values of the Cramér's V metric with respect to different domains and different models. Each number is the average of 100 testcases on the same parameters. For each testcase, 95% of words and 80% of the templates are randomly selected, in order to guarantee a variable setup in the methodology.

As it can be observed, the maximum scores obtained are between 40% and 50%; these indicate a high correlation between the protected and stereotyped properties, which is a signal of the presence of biases in the LM inner representation. On the contrary, percentages below 20% are not statistically significant for detecting an association between categorical properties, meaning that the contingency matrices obtained from the model embeddings do not reflect a biased association. The highest values are detected especially for the gender protected property, when compared to terms indicating professions. More specifically, the first three rows of Table 6 consider the classes of the professions stereotyped property according to their *male* and *female* employment rates. The high scores obtained by bias quantification suggest that all four NLMs have learned the real-world distribution of genders in jobs and express this gap in word representations; therefore, all four models (with different degrees) present a gender bias. Although, the same gender bias is lower when splitting the jobs by salary (fourth row of Table 6).

The second block of rows refers to the nationality protected property, which has been compared to two stereotyped properties (adjectives and verbs), both split in *positive* and *negative* classes. Our aim was to detect whether the perception of specific terms indicating a nationality (such as surnames) might present a connotation of quality on the positive/negative axis. The resulting scores are not high enough to assert that the four models suffer from a nationality bias. In particular, RoBERTa has the lowest scores among the four ($< 5\%$), which might

Properties		**Language Models**			
protected	*stereotyped*	BERT	Distil-BERT	RoBERTa	ELECTRA
gender (51 *female*, 51 *male*)	profession (30 *female-leaning*, 30 *male-leaning*)	32.39%	17.69%	**41.87%**	29.32%
gender (51 *female*, 51 *male*)	profession (11 *female-leaning*, 49 *male-leaning*)	34.78%	36.21%	39.93%	**40.85%**
gender (51 *female*, 51 *male*)	profession (20 *female-leaning*, 20 *balanced*, 20 *male-leaning*)	44.59%	32.91%	**48.5%**	43.22%
gender (51 *female*, 51 *male*)	profession (236 *high-salary*, 237 *low-salary*)	12.82%	11.54%	4.64%	10.25%
nationality (20 *british*, 20 *hispanic*)	adjectives (120 *positive*, 120 *negative*)	16.96%	5.28%	3.64%	**21.26%**
nationality (20 *british*, 20 *hispanic*)	verbs (43 *positive*, 41 *negative*)	9.92%	11.59%	4.7%	9.02%
nationality (20 *british*, 20 *hispanic*, 20 *russian*)	adjectives (120 *positive*, 120 *negative*)	11.02%	6.07%	4.22%	7.84%
religion (20 *christian*, 17 *muslim*)	adjectives (120 *positive*, 120 *negative*)	13.89%	12.2%	2.85%	**34.16%**
religion (20 *christian*, 14 *jewish*, 17 *muslim*)	adjectives (120 *positive*, 120 *negative*)	27.06%	4.5%	18%	**42.23%**
religion (12 *buddhist*, 20 *christian*, 14 *jewish*, 17 *muslim*)	adjectives (120 *positive*, 120 *negative*)	21.97%	20.9%	5.95%	**40.9%**

Table 6: Measured Cramér's V scores for the considered models and properties. The highest bias value for each row is underlined if it exceeds the threshold of 20%.

provide fairer results in the interaction with the users. On the other side, BERT and ELECTRA seem to show more confident results suggesting that a bias might affect them. For example, this might affect surnames like Gomez or Alvarez with a more negative connotation.

In the three bottom rows of Table 6, we compared terms referring to different religions (*Buddhism*, *Christianity*, *Islam*, and *Judaism*) to adjectives connoted on the *positive–negative* direction. The presence of more than two classes for the protected property does not represent a problem for our bias quantification methodology, producing similar scores to the binary case. Finally, the scores for the religion property suggest similar conclusion to the nationality domain: BERT and (especially) ELECTRA showed the highest religion bias and a strong association of *muslim* people to *negative* perception.

The second result we present is a more in-depth investigation of the bias quantification methodology: we want to assess whether the contingency matrices obtained from the language model embeddings actally reflect the presence of bias in the model. Therefore, we select the specific gender×profession bias and compare the results over the same list of words, for different models.

Experiment.

- **Models**: BERT-base, RoBERTa-base, ELECTRA, and DistilBERT.
- **Datasets**: custom datasets composed of word lists for the protected and stereotyped properties.
- **Properties**: gender×profession
- **Testcases**: 100 testcases for each model.

Table 7 shows the resulting contingency matrices for the four models considered. Specifically, these are the average contingency tables on a total of 100 testcases. The classes *female-leaning* and *male-leaning* of the profession property refer to the actual employment percentages from the WinoGender dataset (Rudinger et al., 2018): each class contains 30 words of the 60 original words in the dataset, that are the top half-percentile and bottom half-percentile respectively, based on the actual percentage of female employment in the profession.

BERT gender×profession		*Predicted* values (protected) *female*	*male*	
Actual values (stereotyped)	*female-leaning*	**16.6** (27.67%)	**13.4** (22.33%)	$\implies V = 32.39\%$
	male-leaning	**7.1** (11.83%)	**22.9** (38.17%)	

DistilBERT gender×profession		*Predicted* values (protected) *female*	*male*	
Actual values (stereotyped)	*female-leaning*	**9.3** (15.5%)	**20.7** (34.5%)	$\implies V = 17.69\%$
	male-leaning	**4.8** (8%)	**25.2** (42%)	

RoBERTa gender×profession		*Predicted* values (protected) *female*	*male*	
Actual values (stereotyped)	*female-leaning*	**18.1** (30.17%)	**11.9** (19.83%)	$\implies V = 41.87\%$
	male-leaning	**5.8** (9.67%)	**24.2** (40.33%)	

ELECTRA gender×profession		*Predicted* values (protected) *female*	*male*	
Actual values (stereotyped)	*female-leaning*	**11.6** (19.33%)	**18.4** (30.67%)	$\implies V = 29.32\%$
	male-leaning	**3.9** (6.5%)	**26.1** (43.5%)	

Table 7: Contingency matrices (100 testcases for each) for the gender×profession bias in the four considered models. The profession classes refer to the actual employment male/female percentages. The resulting Cramér's V values are reported right to each matrix, and they correspond to the bias scores reported in the first row of Table 6.

The observed distributions suggest that the professions in which the female employment is greater (namely, the *female-leaning* samples, such as nurse and hostess) are actually labeled (and therefore perceived) by the classifier as *female*. Conversely, a greater male presence in the profession influences a *male* connotation of the associated word in the language model. It is worth noting that this happens even if the words themselves are neutral with respect to gender (i.e., nurse does not have any inherent grammatical gender connotation), and the bias is detected only by analyzing the association between the protected and stereotyped embeddings. This suggests that the models have learned the **real-world distribution of genders** in jobs and express this gap in word representations.

For instance, let us consider the RoBERTa model (third matrix of Table 7). More than 70% of the words (30.17% + 40.33%) are labeled according to the stereotype, i.e. *female-leaning* are labeled as *female* and *male-leaning* are labeled as *male*. The remaining words (19.83% + 9.67% = 29.67%) are labeled counter-stereotypically. Therefore, the contingency matrix of RoBERTa presents a biased association, which is reflected in the high final Cramér's V score of 41.87%.

6.1.1. Validating the features extraction

Our quantification technique relies on the assumption that the LSVM classifier is able to learn the encoding of the protected property by analyzing the embedded words. This is a crucial point, since the resulting contingency matrices and the bias scores are computed on the predictions of this classifier. In order to validate this assumption, we designed a second phase of the experimental procedure, which is based on the **extraction of the most relevant features** for the protected property from the LSVM classifier.

The core idea is to "reduce" the original embeddings by selecting only the most relevant features for the protected property, and then to check whether the bias scores are maintained or even increased. If the selected features are indeed the ones that encode the protected property, then the bias scores should remain stable; on the contrary, if we select the least relevant features, the bias scores should decrease.

The procedure is as follows: first, we train an *auxiliary linear classifier*, which we exploit to extract the relevance of each component of the input. More specifically, we extract the vector of weights $\bar{w} = [w_1, ..., w_n]$ from the trained auxiliary LSVM, whose weights represent — in their absolute values — the relevance of each feature for the protected property.

Next, the top $p\%$ relevant features are selected, and the original embeddings are reduced by maintaining only those features. We indicate the resulting reduced embedding as:

$$R_p^+(e) = [e_{i_0}, e_{i_1}, e_{i_2}, ..., e_{i_r}] \tag{14}$$

where e is the original embedding, p is the percentage of features retained, the set $I = \{i_0, ..., i_r\}$ is indeed the set of features selected (with $|I| = \lfloor p \cdot \#e \rfloor$), and the $+$ sign indicates that we are retaining the most relevant features. On the contrary, we also select the least relevant features, by selecting the features with the lowest absolute weight values in $\bar{w}$. We indicate the resulting reduced embedding as $R_p^-(e)$.

Different scenarios are possible for the resulting bias scores, based on whether our assumption is correct or not:

1. If the features selected in R_p^+ are indeed the ones that encode the protected property, then the bias scores for R_p^+ should remain stable or even increase, since we are removing noise from the original embeddings. At the same time, the bias scores for R_p^- should decrease, since we are removing the components that encode the protected property.
2. Conversely, if the features selected in R_p^+ do not encode the protected property, then we should observe a random behavior of the bias scores for both R_p^+ and R_p^-, since we are removing features that do not correlate with the protected property.

Experiment.

- **Models**: BERT-base, RoBERTa-base, ELECTRA, and DistilBERT.
- **Properties**: gender×profession, nationality×adjectives, and religion×adjectives.
- **Variation**: $p \in [100\%, 90\%, ..., 10\%]$

Figure 10 shows the results of this experiment for the gender×professions bias in the four models analyzed, corresponding to the four plots represented in the figure. Each plot illustrates the Cramér's V scores (vertical axes) as the percentage of retained features in the embeddings varies from 100% to 10% (horizontal axes). The green lines represent the results for which the most-relevant features are retained (R^+), whereas the red lines shows the scores when retaining the

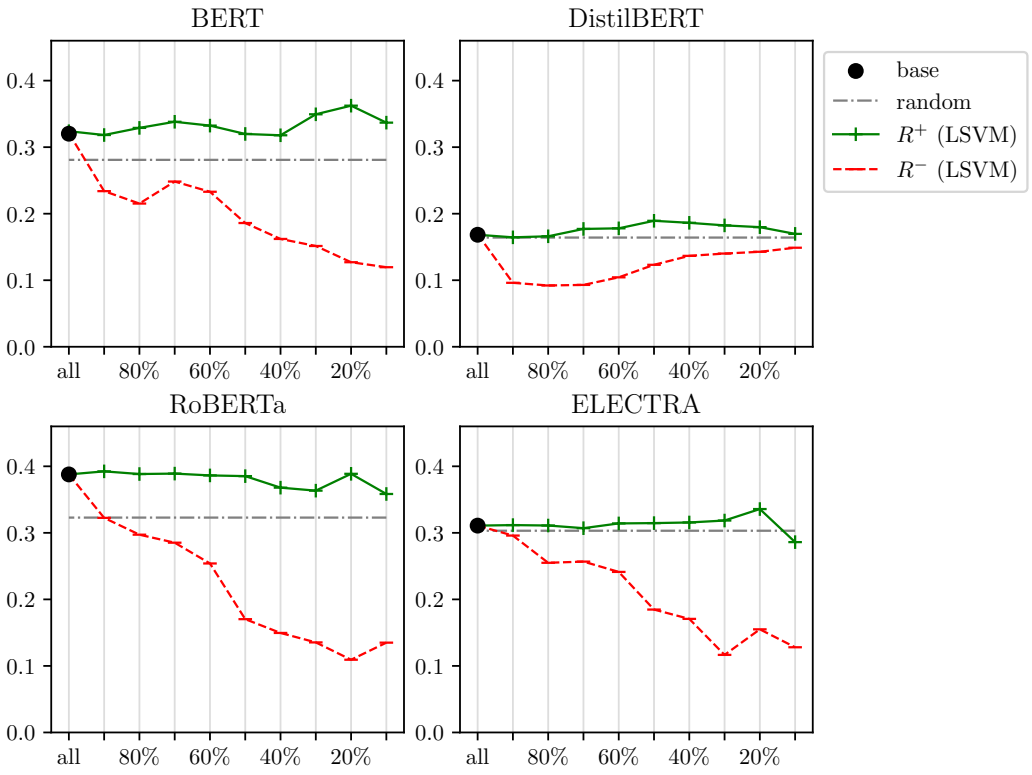


Figure 10: Plots of the Cramér's V score for the gender×professions properties, according to the percentage of features retained from the original embeddings. The green R^+ line retains the given percentage of the *best* features, whereas the red R^- line retains the same percentage of *worst* ones. The black dot represents the base V score, obtained with no features selection (i.e. all the features were retained).

less-relevant features (R^-). Both lines of each chart start from a black point, representing the Cramér's V score obtained by considering the embeddings entirely (100% of features retained), and diverge as the number of features decreases.

As can be seen from the trend of the lines, the curves of the R^+ embeddings remain almost flat, independently from the percentage of features retained. Vice versa, the curves relating to the R^- embeddings are descending. These trends are consistent with the first scenario described above, and they are a sign that removing the components labeled as "relevant" leads to a decrease in the correlation measure, while removing the "irrelevant" components does not affect the bias scores. These results suggest that the features selected are, indeed, the ones

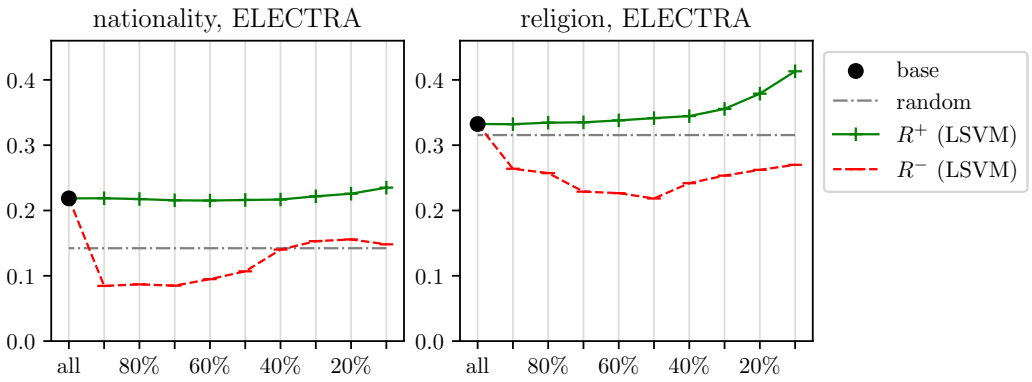


Figure 11: Plots for the nationality and religion protected properties, compared with a series of positive and negative adjectives. On the x-axis, the percentage of features considered in the experiment. On the y-axis, the Cramér'V scores.

that encode the protected property, and thus our main classifier learns to predict the protected property.

An analogous experiment was conducted on the ELECTRA model — which showed the highest bias scores in the previous experimental session — for the nationality and religion properties.

Experiment.

- **Model**: ELECTRA.
- **Properties**: nationality×adjectives, religion×adjectives.
- **Variation**: $p \in [100\%, 90\%, ..., 10\%]$

The plots in Figure 11 reflect the same trends observed in Figure 10, confirming the validity of the bias quantification methodology. However, the decrease in the bias scores for the R^- embeddings is less steep than in the previous experiment. For the nationality property (left plot of Figure 11), we observe a huge gap after removing the top 10% of features, but then the scores remain stable for the remaining percentages, ending with a slight increase maybe due to noise. For the religion property (right plot of Figure 11), the decrease is less pronounced, and the scores remain relatively stable even after removing a large portion of features. This might be a sign that the bias is encoded in a more distributed way across

the embedding components, and thus removing only the most relevant features does not lead to a strong decrease in the bias scores.

Finally, for both Figure 10 and Figure 11, we remark how the green R^+ lines are above the gray dotted line, whereas the red R^- lines move below. The gray line corresponds to the bias score obtained by selecting random features, which represents a sort of baseline for the experiment. In fact, the selection of the most-relevant features produces a higher score than the selection of random features, whilst the random selection still produces a score higher than selecting only the least-relevant features.

6.1.2. Dependence on the word dataset

Another important aspect to investigate is the dependence of the bias quantification results on the specific word dataset used for the protected and stereotyped properties. In this latter experiment on our bias quantification methodology, we analyze the robustness of the bias scores with respect to the variation of the input word dataset.

The core idea is to gradually reduce the number of words in the original dataset: starting from the complete dataset, we progressively apply a random ablation of the words, by retaining only a certain percentage of them. For each percentage of words retained, we compute the bias scores and analyze their variation with respect to the original scores obtained with the complete dataset.

If the bias scores are robust with respect to the word dataset (i.e. they reflect a real bias in the model rather than an artifact of the specific words used), then we should observe stable scores even when reducing the number of words. Conversely, if the bias scores are highly dependent on the specific word dataset, then we should observe a significant variation in the scores as we reduce the number of words.

Experiment.

- **Models**: BERT-base, RoBERTa-base, ELECTRA, and DistilBERT.
- **Properties**: gender×profession.

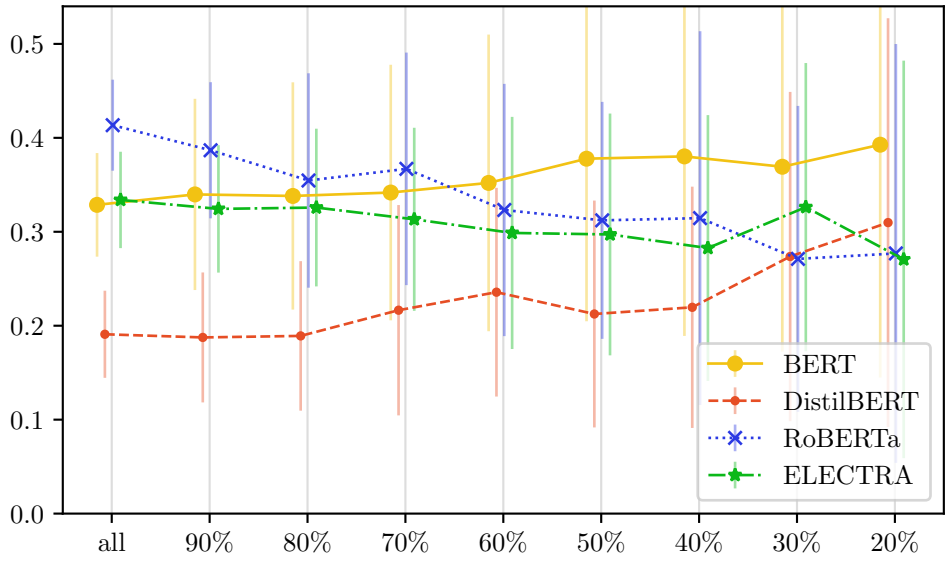


Figure 12: Plots for the gender × profession properties. The Cramér's V scores are computed for different percentages of words retained in the original datasets (both protected and stereotyped). On the x axis, the percentage of training set considered. On the y axis, the Cramér's V score.

- **Variation**: percentage of words retained in the original dataset, ranging from 100% to 10%.
- **Testcases**: 50 testcases for each model and percentage, with random selection of words and templates.

In Figure 12 we show the results of this experiment for the gender×profession bias in the four models analyzed. The colored lines correspond to the average bias scores (the Cramér's V), calculated over a total of 50 testcases, whereas the vertical intervals graphically show the amplitude of the standard deviation for each plot point, providing a quick indication of the uncertainty of the result. The plot points are slightly shifted on the x-axis for better readability, but they are all computed on the same percentages of words retained (100%, 90%, 80%, ..., 10%).

We can observe that, as we reduce the number of words available for the analysis, the average values do not undergo strong changes. A slight increase in the average bias detected for BERT and DistilBERT can be detected (yellow and red lines respectively), while RoBERTa and ELECTRA appear to have a slightly decreasing trend (blue and green line respectively), even if the variations are not

significant. On the other hand, the most interesting aspect of this experiment lies in the vertical error bars: with the complete dataset, the standard error is around 4%, therefore sufficient to confirm the presence of a correlation between the protected property and the stereotyped one. However, as the number of words decreases, the error bars grow significantly, reaching values around 20% when only few words are retained. This suggests that a minimum dataset size is necessary to obtain reliable bias quantification results, as the estimator's variance grows substantially when fewer words (10–15 per property) are available for analysis.

6.2. Bias Visualization

The focus of the experiments now shifts to **bias visualization**, which aim to provide an intuitive understanding of stereotypes in NLMs by visualizing their internal representations and decision boundaries. The section presents the experiments on our bias visualization technique described in Section 5.3, which we refer to as *Weakly Supervised Visualization* (WSV) (Dusi et al., 2022).

6.2.1. Comparison with PCA and MLM score

In the first experiment, our *Weakly Supervised Visualization* (WSV) is compared with a standard dimensionality reduction technique, the *Principal Component Analysis* (PCA) (F.R.S., 1901). Both techniques serve the same purpose of reducing the dimensionality of the data to two dimensions, but they differ in their approach: PCA is an unsupervised method that identifies the directions of maximum variance in the data, while WSV is a supervised method that incorporates protected information to guide the dimensionality reduction process. Our claim is that, by incorporating protected information, WSV can produce a more meaningful visualization of the bias present in the data.

The experiment considers the gender×occupation bias in a BERT-base model (Devlin et al., 2019), which is one of the most widely used NLMs. The test is

conducted on a dataset of 1678 occupations (Neidel, 2021), which are encoded in the embedding space of the last layer of BERT. The visualization is obtained by applying PCA and WSV to the occupation embeddings, resulting in two-dimensional representations of the occupations. In order to compare the two visualization, each occupation word is also tested with a standard bias quantification technique, which is the difference in probabilities of the masked token being predicted as a *male* or *female* pronoun in a sentence containing the occupation word. This technique is based on the *Masked Language Modeling* (MLM) score proposed by Kurita et al. (2019).

> *Example.* Consider the occupation word nurse, which we want to test for gender bias. Our MLM score is calculated by masking the subject in a sentence containing the occupation word, such as [MASK] is a nurse, and then querying the model. Intuitively, if the probability of the masked token [MASK] being predicted as a *male* is $p(\text{he}) = 0.8$ and the probability of it being predicted as a *female* pronoun is $p(\text{she}) = 0.1$, then the resulting bias score is $p(\text{she}) - p(\text{he}) = -0.7$ which indicates a pro-*male* bias.

The MLM score is used as a reference to color the points in the visualization, allowing us to visually assess the correlation between the bias detected by the MLM score and the spatial arrangement of the occupations in the two-dimensional space. Our hypothesis is that WSV will produce a visualization where gendered occupations are more clearly separated according to their bias, showing a gradient of colors that reflects the MLM score.

> *Experiment.*
>
> - **Model**: BERT-base
> - **Dataset**: 1678 occupations from Neidel (2021).
> - **Visualization techniques**: PCA and WSV.
> - **Bias quantification**: MLM score for gender bias.

The results of the experiment are shown in Figure 13, where the upper chart represents the PCA visualization and the lower chart represents the WSV visualization. Each point represents a different occupation, colored according to the

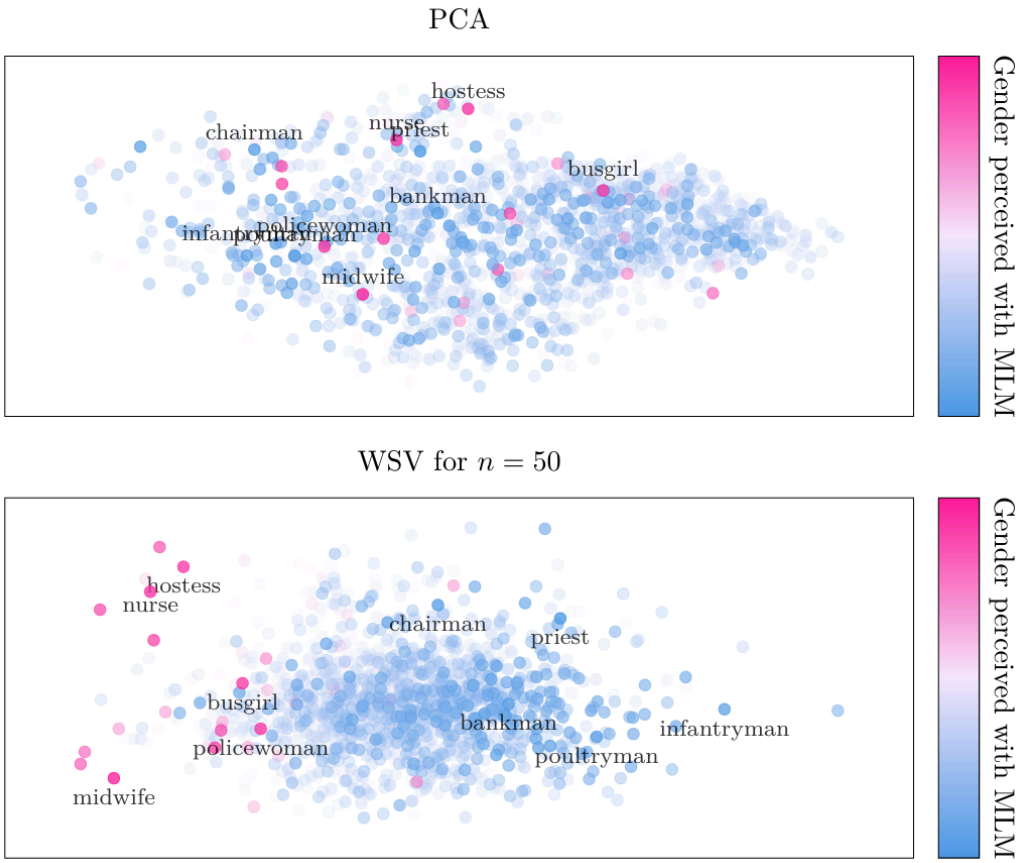


Figure 13: Comparison between the visualization of 1678 occupations with PCA (first two principal component) and with our method. The color indicates the bias towards the *male* (blue) or *female* (pink) stereotype, detected with MLM; jobs perceived as neutral are rendered with transparency. The five most biased samples for both classes are labeled in the chart.

MLM score: blue points indicate occupations that are biased towards the *male* stereotype, while pink points indicate occupations that are biased towards the *female* stereotype; points that are perceived as neutral are rendered with transparency.

The PCA visualization reveals no clear spatial correlation with the MLM bias scores, with blue and pink points distributed uniformly across the space without discernible patterns. In contrast, the WSV visualization demonstrates a more structured arrangement, where pink points concentrate on the left region of the space, forming a coherent gender spectrum that reflects stereotypical associ-

ations. This difference suggests that while PCA compression to two dimensions fails to capture meaningful bias structure beyond extreme cases, our proposed method successfully reveals the underlying gender bias distribution in a more intuitive and interpretable manner. For instance, jobs like nurse or hostess are strongly related to the female gender and occupies the left region of the two-dimensional space, while typically masculine jobs like priest or infantryman are in the opposite region.

As a quantitative measure of the effectiveness of the visualization, we can compute the correlation between the coordinates of the points in the two-dimensional space and the MLM bias scores. In the case of WSV, the horizontal axis (the first component) shows a strong correlation with the MLM score of 42%.

In conclusion, our WSV method, by incorporating protected information, is able to produce a more meaningful visualization than PCA, revealing the underlying bias structure in the data. PCA is indeed simpler, yet not effective in this context. However, it needs to be noted that the *Weakly Supervised Visualization* requires the set of an hyperparameter n, which corresponds to the number of dimensions selected in the first reduction sub-step (see Section 5.3). In this experiment, we set $n = 50$.

6.2.2. Correlation with real-world data

In the previous experiment, we showed that WSV produces visualizations that better reflect standard bias measures (like MLM scores) compared to PCA. Here, we continue comparing these two techniques, but instead of relying on the MLM score, we use actual data on the gender distribution of workers across different occupations. The goal of this experiment is to test whether model bias aligns with real-world bias patterns. If it does, our visualization method should capture and display this alignment clearly.

This experiment utilizes the *WinoGender* dataset (Rudinger et al., 2018), which contains a list of 60 occupations along with their gender employment rates. The process is similar to the previous experiment: we encode the occupation words in

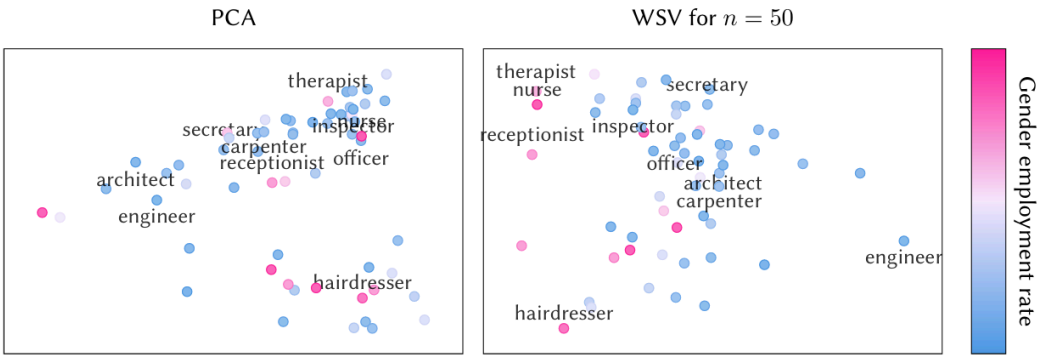


Figure 14: Plots for the WinoGender dataset. The left chart is obtained by applying PCA, while the right one is obtained with our method. The color of each point represents the female employment rate.

the embedding space of BERT, apply PCA and WSV to reduce the dimensionality to two dimensions, and then color the points according to the percentage of female workers in each occupation. The hypothesis is that WSV will produce a visualization where occupations with higher female employment rates cluster together.

Experiment.

- **Model**: BERT-base
- **Dataset**: 60 occupations from the *WinoGender* dataset (Rudinger et al., 2018).
- **Visualization techniques**: PCA and WSV ($n = 50$).
- **Tested Bias**: gender×occupation.

Results are shown in Figure 14, where the left chart represents the PCA visualization and the right chart represents the WSV visualization. Samples are colored according to the gender employment rate: pink points indicate occupations into which the female workers are the vast majority, while blue points represent jobs with a low percentage of female workers; occupations with a more balanced gender distribution are rendered with transparency. Again, the five most polarized samples for both classes are labeled in the charts.

Similar to the previous experiment, the PCA visualization does not show a clear spatial correlation with the gender employment rates: pink and blue points

are scattered across the space without any discernible pattern. On the contrary, the WSV visualization reveals a more structured arrangement, with pink points such as nurse or hairdresser clustering on the left side of the space, and blue points like engineer on the right side.

Finally, we can compute the correlation between the coordinates of the points in the two-dimensional space and the female employment rates. The first component of WSV ($n = 50$) shows a strong correlation of 56%, while the PCA's most correlated component has a correlation of only 24%. Our result is comparable to the MLM score (Kurita et al., 2019) computed on the same *WinoGender* dataset, which has a correlation with the female employment rates of 59%.

6.3. Bias Tracing

The latter part of the results chapter is focused on the learning dynamics of bias in NLMs. As explained in Section 4.4, understanding how bias emerges and evolves during the training of a language model is crucial (1) for a better comprehension of the underlying mechanisms that lead to bias behavior in NLMs, and (2) for developing effective mitigation strategies. With this in mind, the experiments presented in this section are designed to trace the bias acquisition process by applying our bias quantification methodology (described in Section 5.2) at different stages of the training of a language model. We refer to this process as **bias tracing**.

6.3.1. Bias across pre-training trajectories

First, we analyze the bias acquisition during the **pre-training phase** of a language model. The pre-training phase is where the model learns from a large corpus of text data, and it is during this phase that the model is exposed to the broadest range of linguistic patterns and world knowledge, including any biases present in the training data.

The model considered for the experiment is *Pythia-160M*, a publicly available NLM with multiple checkpoints during its training. For each checkpoint — up to a certain threshold after which the model's performance on linguistic tasks does not improve significantly, typically 100k checkpoints — we apply our bias quantification method. The objective is to observe whether there are any significant shifts in the bias scores across the checkpoints, which could indicate critical phases in the training where bias acquisition is more pronounced.

Experiment.

- **Model**: Pythia-160M (160 million parameters) (Biderman et al., 2023).
- **Dataset**: custom dataset for bias scoring; the training data is not directly used in the experiment, as the checkpoints are already pre-trained and available for testing.
- **Properties tested**: gender×profession bias, and religion×adjective|verb bias.

The results of the experiment are shown in Figure 15, where the bias scores (measured with Cramér's V) are plotted across the training checkpoints for different bias tests. As can be observed, all tests resulted in a statistically non-significant outcome: the measured bias presented a huge variance and no visible monotonic trends (increasing or decreasing) for the whole series of checkpoints. This may suggest that bias acquisition during pre-training is a complex and non-linear process, or that our bias quantification method is not sensitive enough to detect subtle changes in bias across checkpoints. Not being able to infer any clear pattern from the results, we claim that more research is needed.

6.3.2. Bias across fine-tuning trajectories

As a second step, we investigate the bias acquisition during the **fine-tuning phase** of a language model. The fine-tuning phase is where a pre-trained model is further trained on a specific dataset for a particular task, and it is during this phase that the model can be exposed to more specific biases related to the task or domain of the fine-tuning data. The fine-tuning phase is often regarded as

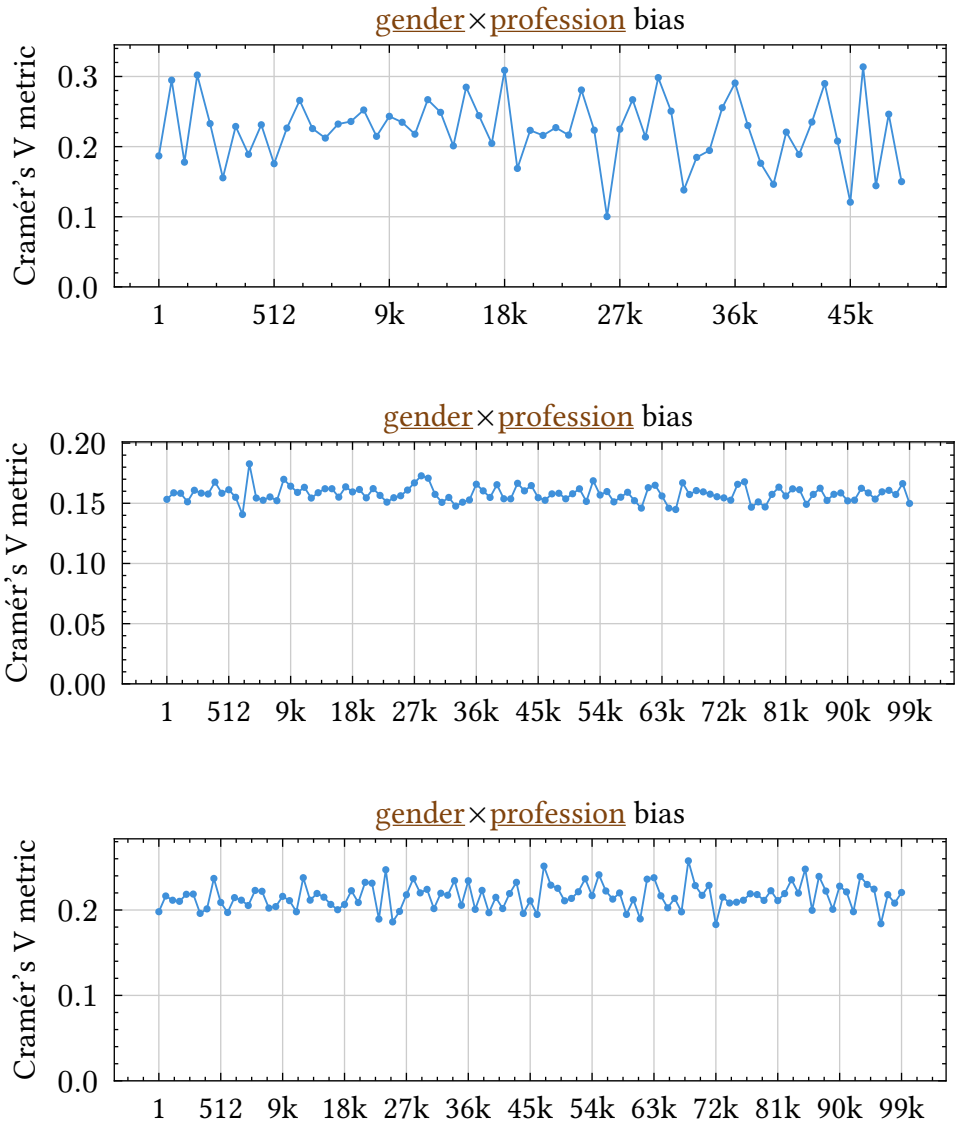


Figure 15: Bias scores measured with Cramér's V across pre-training checkpoints of model Pythia-160M, for different bias tests.

a critical phase for bias acquisition (see Section 4.4), as the model has already learned general linguistic patterns and world knowledge during pre-training, and the recent exposure to specific data can lead to significant shifts in the model's bias behavior. To accomplish this, we fine-tuned the pre-trained Pythia-160M model on two different datasets, and we measured the resulting changes in bias

scores across the fine-tuning checkpoints. The datasets used are a subset of the BigNews dataset by Liu et al. (2022), containing English news articles with a high concentration of politically charged content (e.g., articles with strong "left" or "right" political leaning). The articles were split into two subsets: one containing left-leaning articles and the other containing right-leaning articles, based on the labels provided by the original dataset collectors.

Experiment.

- **Starting model**: Pythia-160M (160 million parameters) (Biderman et al., 2023)
- **Datasets**: two subsets of the BigNews dataset (Liu et al., 2022), split into articles with strong "left" or "right" political bias.
- **Objective**: to evaluate the sensitivity of the model's internal representations to specific data exposures and to identify critical training phases where bias acquisition is most pronounced.

After several epochs of fine-tuning, the observed results were again statistically non-significant, with no clear patterns of bias acquisition across the fine-tuning checkpoints for both the left-tuned and the right-tuned models. Furthermore, we observe no significant differences between the left-tuned and the right-tuned checkpoints. Motivations at the base of such outcomes range from the model already having acquired most of the bias during pre-training, to the fine-tuning data not being sufficiently biased to induce significant changes in the model's bias behavior, to the bias quantification method not being sensitive enough to detect subtle changes in bias across checkpoints. It is hard to draw definitive conclusions from these results, but surely more research is still needed to understand the dynamics of bias acquisition during fine-tuning. In the next chapter, we will discuss the implications of these outcomes and frame them within the broader context of this thesis.

Chapter 7

Discussion

This chapter discusses the findings from both the state-of-the-art review and the experimental work presented in this thesis. We examine what emerges from current literature about bias in language models, then reflect on the methodologies we developed and the results we obtained. Finally, we consider the broader implications and limitations of this research.

7.1. Insights from the State of the Art

The systematic review of bias research in NLMs reveals a complex landscape, where simple narratives fail to capture the entire nature of the problem (Chapter 4). The evolution of **bias measurement techniques** over the past decade demonstrates both progress and persistent challenges. Starting from geometric measurements in static word embeddings, the field has developed increasingly sophisticated methods — including association tests, template-based probes, and generation-based evaluations. Instead of converging on a single "best" method that can be applied universally, the current state of the art shows a diverse toolkit of approaches, reflecting the multifaceted nature of bias.

A fundamental distinction emerges between **intrinsic** measurements (related to inner representations of concepts and text) and **extrinsic** measurements (what a model actually produces in deployment). As highlighted in the literature (Cao et al., 2022; Orgad et al., 2022), a model can appear fair on intrinsic metrics yet produce biased outcomes in practice, or conversely, encode stereotypical associations that never surface in task performance. This disconnect underscores a critical limitation: no single measurement can fully characterize bias in an NLM.

With this limitation in mind, we then examined the different factors that may contribute to bias acquisition. Research proves that **training data is a primary**

driver of bias in NLMs (Section 4.3). Data influences bias through multiple mechanisms: class imbalance creates performance disparities across groups, word frequency patterns create spurious correlations with bias metrics, and the cultural and political perspectives of text corpora become encoded in model representations. However, the relationship between data and bias is not straightforward. Simply balancing class distributions does not reliably reduce bias, and the influence of data varies across training phases, with the fine-tuning stage often exerting higher influence on downstream bias compared to the upstream pre-training corpus (Section 4.4).

This finding has important implications in the context of bias mitigation: efforts focused solely on pre-training data may be undermined by subsequent fine-tuning. Furthermore, the training process itself exhibits fragility, where minor changes in hyperparameters, random initialization, or alternative training procedures (e.g. knowledge distillation) can produce significant variations in fairness metrics while barely affecting task accuracy (Baldini et al., 2022).

Our analysis then moved towards the **inner workings of the model** (Section 4.5). Research reveals that bias is neither uniformly distributed nor stored in a single location. Instead, it manifests across multiple architectural scales: in the geometry of embedding spaces, in specific attention heads (particularly in later layers), in feed-forward network components, and even in individual neurons or small neuron groups. The simultaneous distribution of bias across many components creates potential conflicts when interventions target individual elements. However, from the perspective of inner biased representations, some patterns emerge. For instance, the **linear subspace hypothesis** has found empirical support across various model architectures, suggesting that at least some biases have structured, interpretable geometric encodings (Li et al., 2025; Shin et al., 2020; Vargas & Cotterell, 2020).

Some studies also suggest that **bias is entangled with other model capabilities**: bias neurons can be coupled with other functional neurons, creating trade-offs where reducing bias may inadvertently affect other aspects of model

behavior (Qian et al., 2025; Zhao et al., 2018a). This entanglement complicates mitigation strategies and suggests that bias cannot always be "removed" without collateral effects.

Finally, the relationship between **model size** and bias defies simple characterization (Section 4.6). Implicit, stereotypical associations tend to strengthen with scale, amplifying patterns present in training data, while extrinsic bias seems to decrease with scale. Literature suggests a qualitative shift as models grow: larger models become better at recognizing that certain statements are socially inappropriate, even as their internal representations encode the very stereotypes they learn to avoid expressing (Gupta et al., 2022; Zhao et al., 2025).

In conclusion, despite progress in understanding and measurement, several fundamental challenges remain. Methods designed to reduce bias on one benchmark often fail to generalize to other bias constructs, domains, or languages. At the same time, mitigation techniques frequently face trade-offs between bias reduction and model capabilities. Therefore, contemporary research in the field of NLP fairness increasingly emphasizes multi-metric evaluation and careful documentation of bias constructs, intervention points, and intended deployment contexts. This reflects a growing recognition that the problem of bias detection and mitigation does not have a simple universal solution, but is rather a context-dependent challenge requiring tailored approaches.

7.2. Discussion of the Experimental Methodologies

The methodologies developed in this thesis adopt a **property-based framework** that connects abstract fairness concepts to concrete linguistic analysis. This design choice to structure the analysis around protected and stereotyped properties provides both conceptual clarity and operational flexibility. For instance, our approach has allowed us to systematically vary which properties are examined, how they are operationalized through word lists, and which associations to consider problematic.

Leveraging this framework, the bias quantification method centers on a **classification-based approach**: training a linear classifier on protected embeddings and evaluating its predictions on stereotyped embeddings, with Cramér's V measuring the association strength.

Several design choices merit discussion:

- The use of **linear SVMs** as the classifier is justified by both theoretical and practical considerations. Theoretically, if bias is encoded as linear subspaces (as the literature suggests), a linear classifier is appropriate for detecting these patterns. Practically, linear classifiers are interpretable, with weights indicating feature importance, and they performed better than alternatives in our experiments when working with limited dataset sizes.
- Second, the choice of **Cramér's V** as the association metric addresses key limitations of alternatives, as we stated in Section 5.2 and presented in Table 5, being the only metric that is both normalized and bounded in $[0; 1]$.
- Third, the methodology's two-stage structure (training on protected embeddings, testing on stereotyped embeddings) provides a **clear interpretation**: high association scores indicate that stereotyped words cluster in embedding space according to protected categories, suggesting the model has learned to systematically link these properties.

However, limitations remain with the approach, which should be acknowledged:

- The framework's reliance on **predefined word lists** introduces certain constraints. First of all, any word-based operationalization of social categories is an approximation of complex, multifaceted human attributes. While our experimental validation (Subsection 6.1.2) suggests that results are reasonably robust to variations in word lists, and the choice of terms and templates reflects common practices in the literature, the selection process inevitably encodes researcher assumptions about how properties manifest linguistically.
- The method focuses on **intrinsic bias** in contextual embeddings, which may not directly translate to extrinsic bias in model behavior, as literature has shown. The approach is coherent with our goal of better understanding the

internal mechanisms of bias in language models, but it may not predict real-world harms in deployment contexts.

- The **binary or multi-class operationalization of properties** simplifies continuous and multidimensional human attributes. Gender, for instance, is treated as a binary option in most experiments, despite being far more complex in reality. While this simplification aligns with much existing literature and enables controlled analysis, it limits the framework's ability to capture intersectional or non-binary aspects of social identity. Theoretically, it is always possible to segment the property spectrum into more fine-grained categories, but this comes at the cost of higher data requirements, which may not be feasible for all bias constructs, classes, or languages. For instance, collecting sufficient word samples for non-binary gender categories is challenging, and the resulting bias scores may be less reliable due to smaller sample sizes or the processing of lower-frequency terms.

7.3. Validation and Empirical Findings

Several experimental validations support the **bias quantification** methodology's reliability. In Section 6.1, we showed that the method detects meaningful differences across models and bias types, with patterns that align with real-world distributions.

As an overall trend, we observed that bias manifests differently depending on the bias construct and model architecture. For instance, gender×occupation associations are strongest (40-50% Cramér's V), while gender×salary associations are much weaker (below 15%). As another example, religion and nationality properties show high model variance, with ELECTRA exhibiting substantially higher bias than RoBERTa. This variability confirms that bias is not uniform across models or properties, and that the methodology can detect these differences. In addition, the feature extraction experiment (Subsection 6.1.1) demonstrates that the classifier genuinely learns to identify bias-encoding dimensions, whereas

the word ablation experiment (Subsection 6.1.2) shows reasonable robustness to variations in the word dataset. Both experiments validate that the methodology detects genuine patterns rather than artifacts, though adequate sample sizes remain important for reliable estimation.

Regarding **bias visualization**, the *Weakly Supervised Visualization* (WSV) method has proven effective against standard PCA for revealing global bias-relevant structure in embedding spaces. At the same time, our approach allows for sample-specific analysis, enabling researchers to identify which particular words exhibit the strongest biased associations. However, visualization also has limitations: it reduces high-dimensional patterns to two dimensions, inevitably losing information, and the reliance on the hyperparameter n (number of protected subspace dimensions) introduces a tuning requirement. Different values of n may reveal different aspects of bias, requiring some experimentation to find appropriate settings.

Finally, the **bias tracing** experiments failed to detect clear patterns across training checkpoints, yielding statistically non-significant results. Rather than invalidating the approach, this negative finding highlights that bias acquisition is likely a complex, non-linear process where simple trajectories do not emerge. Possible explanations include limited methodology sensitivity, the genuine non-monotonicity of bias dynamics, mismatch between measured constructs and training data variations, and model scale differences. This result underscores how much remains unknown about bias acquisition dynamics, as it emerged from the literature review.

7.4. Final Remarks and Limitations

The main insight coming from this research is that bias in NLMs is not a simple phenomenon with a single cause or location. It arises from training data reflecting societal patterns, gets shaped by training dynamics that can amplify or modify initial tendencies, manifests in distributed architectural components, and varies

with scale in complex ways. Yet, we have shown that despite this complexity, bias can be measured reliably at the representation level using classification-based approaches, and visualized meaningfully through weakly-supervised dimensionality reduction. The methodologies developed in this thesis provide tools for approaching bias in a systematic way, even though the relationship between intrinsic bias and real-world harms remains an open question. From a general perspective, scientific literature has made significant progress in developing methods to quantify and mitigate bias, but the development of universally applicable solutions is challenged by multiple factors, such as the gap between controlled experimental conditions and real-world deployment contexts, the context-dependency of what constitutes harmful bias, and the entanglement of bias with other model capabilities.

Our work occupies a specific niche in the broader landscape of bias research, focusing on intrinsic representational bias in encoder-based models (BERT-family), yet contemporary applications increasingly use decoder-only architectures (GPT-family) for generation tasks. A direction still unexplored is the applicability of our approach to multilingual models and non-English languages, where bias constructs may differ and word-based operationalizations may be more challenging. Gender bias, for instance, may manifest differently in languages with grammatical gender, requiring adapted procedures accordingly. Similarly, the applicability to other cultures and forms of bias (intersectional identities, age, disability, etc.) needs validation.

Chapter 8

Conclusion

This thesis has systematically investigated the origins of social biases in deep Neural Language Models and developed practical, interpretable, white-box methods for their detection and quantification. By synthesizing recent literature and conducting targeted experiments, it challenged the prevailing view of bias as merely a "data artifact", endorsing instead an emergent multifactorial perspective that accounts for training dynamics and architectural encoding.

8.1. Summary of Contributions

The core contributions advance both theoretical understanding and practical tooling for AI fairness in NLP. A comprehensive literature review across five years of research reframes bias origins through four lenses: data properties beyond imbalance, training-stage evolution, bias-responsible loci in embeddings and Attention, and non-monotonic scaling patterns.

New detection techniques operationalize stereotypes via protected×stereotyped property pairs (e.g., gender×profession or religion×adjective). Bias quantification employs LSVM classifiers on embeddings, evaluated with Cramér's V metric, proving robust to dataset subsampling and feature ablation. Complementing this, Weakly Supervised Visualization (WSV) clusters embeddings interpretably, outperforming PCA and aligning with real-world disparities (like *WinoGender* employment rates). Empirical tracing across Pythia checkpoints and fine-tuning showed very limited effects on bias acquisition, providing no clear evidence for a sudden emergence phase. These white-box probes require minimal labels, without reliance on large annotated datasets or complex downstream tasks.

8.2. Future Research Directions and Final Comments

This work suggests several complementary directions that span methodological innovation, practical deployment, and conceptual understanding.

- **Multilingual and Cross-Linguistic Expansion**: validating this framework across more languages would test the generality of observed patterns, while probing language-specific effects (e.g. grammatical gender) and cross-lingual transfer mechanisms.
- **Full Trajectory Analysis**: scaling bias tracing to larger models with open releases (e.g. Llama) would illuminate the complete lifecycle of bias emergence.
- **Mitigation Integration and Continuous Debiasing**: given that bias is entangled with knowledge and varies across training phases, simple one-time debiasing interventions are unlikely to suffice. Exploring pre- and post-training interventions (CDA, distillation) would quantify trade-offs between fairness and other capabilities. Promising directions include continuous monitoring, context-dependent debiasing strategies, and architectural innovations that structurally separate different knowledge types to reduce entanglement.
- **Field-Level Infrastructure**: the field of NLP fairness would benefit from standardized evaluation protocols enabling cross-study comparison, improved documentation of bias constructs and measurement contexts, and greater emphasis on negative results and limitations. For instance, the null findings from bias tracing experiments are scientifically valuable precisely because they challenge assumptions and highlight methodological boundaries.

In conclusion, even if this thesis diverges from a purely data-centric explanation of bias, one starting point remains essential: language models are powerful tools that reflect and amplify patterns — including biases — present in their training data. This is not a merely technical issue: such manifestations have measurable consequences for individuals and communities using these technologies, and for society at large. Progress in quantification and mitigation has been substantial, yet far from comprehensive or systematic.

In our view, addressing these challenges requires focus on three aligned directions. First, methodologically, more sensitive detection techniques and broader evaluation protocols are needed to establish **consistent comparison across models and contexts**. Second, conceptually, better characterizing the relationship between intrinsic bias (in representations) and extrinsic bias (in downstream behaviors) would clarify **when and how detection and mitigation techniques should be applied**. Third, practically, the observation that bias is distributed, entangled with useful knowledge, and varies across training phases argues against one-time debiasing interventions in favor of **continuous, context-aware monitoring**.

Ultimately, the goal is not a single definitive fix, but a sustainable research direction that iteratively refines our understanding and tools for bias in the Natural Language Processing field, while transparently communicating limitations and uncertainties. This is the necessary condition for developing and deploying language technologies that are both powerful and equitable in real-world contexts, for the benefit of all users.

Acknowledgements

This work was carried out while the author, Michele Dusi, was enrolled in the *Italian National Doctorate on Artificial Intelligence* run by *Sapienza University of Rome* in collaboration with the *University of Brescia*.

The author would like to express his sincere gratitude to his supervisor and co-supervisors for their support, guidance, and continuous trust throughout the research process and PhD journey. Special thanks are due to all the colleagues and collaborators at the University of Brescia, for their insightful feedback and stimulating discussions that enriched this work.

Appendix

-.1 Dataset Details

The datasets used in this thesis — words and templates — are listed in detail in the following pages. For each dataset, we report the property of interest, the class labels, and the number of words in each class.

adjective — Words

positive (120)	*negative* (120)
adaptable, adventurous, affable, affectionate, agreeable, ambitious, amiable, amicable, amusing, artistic, brave, bright, broad-minded, calm, careful, charismatic, charming, chatty, cheerful, clever, communicative, compassionate, conscientious, considerate, convivial, courageous, creative, decisive, dependable, determined, diligent, diplomatic, discreet, dynamic, easy-going, efficient, emotional, energetic, enthusiastic, extroverted, exuberant, fair-minded, faithful, fearless, forceful, frank, friendly, funny, generous, gentle, good, hardworking, helpful, hilarious, honest, humorous, imaginative, impartial, independent, industrious, intellectual, intelligent, intuitive, inventive, joyful, kind, kooky, likable, loving, loyal, lucky, modest, neat, nice, non-judgemental, observant, optimistic, organized, passionate, patient, persistent, philosophical, pioneering, placid, plucky, polite, popular, powerful, practical, pro-active, quick-witted, quiet, rational, reliable, reserved, resourceful, romantic, self-confident, self-disciplined, sensible, sensitive, shy, sincere, smart, socialable, straight-forward, sympathetic, talkative, thoughtful, tidy, tough, trustworthy, unassuming, understanding, upbeat, versatile, warmhearted, wild, wise, witty.	aggressive, aloof, anxious, arrogant, bad-tempered, belligerent, big-headed, bitter, boastful, boring, bossy, callous, careless, chaothic, clingy, cold, confrontational, conniving, corrupt, cowardly, cruel, cynical, deceitful, defensive, devious, dim, dishonest, disloyal, dismissive, disobedient, disorganized, disrespectful, disruptive, dogmatic, domineering, egotistical, envious, fickle, finicky, foolish, fraudolent, fussy, greedy, grumpy, gullible, gussy, harsh, horrible, hostile, hypocritical, idle, ignorant, impatient, impolite, impulsive, inconsiderate, inconsistent, indecisive, indiscreet, inflexible, intolerant, introverted, irresponsible, jealous, lazy, loud, malicious, manipulative, mean, miserable, moody, narrowminded, nasty, naughty, nervous, neurotic, obstinate, overcritical, overemotional, patronizing, pessimistic, petty, petulant, pig-headed, pompous, possessive, quick-tempered, resentful, ridiculous, rude, ruthless, scary, secretive, self-centered, selfish, silly, sneaky, stubborn, stupid, sullen, tactless, thoughtless, touchy, ugly, uncivilized, unlawful, unlucky, unpredictable, unreliable, untidy, untrustworthy, vague, vain, vapid, vile, vindictive, violent, vulgar, weak, weak-willed.

adjective — Templates

He said I'm too <adjective>. I'm too <adjective> for you. Yesterday I met the most <adjective> person I've ever met. The bus driver was very <adjective>, don't you think? My elementary school teacher was very <adjective>. No, I'm not <adjective>. I don't think you're <adjective>. I must confess: I find you very <adjective>. Are you aware that you're <adjective>? How can you be so <adjective>? You're a very <adjective> person. They're <adjective>, aren't they? My parents are not very <adjective>...

gender — Words

male (51)

abbot, actor, bachelor, barman, baron, boy, boyhood, boyish, brother, brotherly, dad, daddy, duke, earl, emperor, father, fatherly, gentleman, god, grandfather, grandpa, grandson, groom, he, hero, him, his, husband, king, knight, lad, lord, male, man, manly, marquess, masculine, masculinity, master, monk, nephew, priest, prince, prior, sir, son, uncle, viscount, waiter, warlock, widower.

female (51)

abbess, actress, aunt, barmaid, baroness, bride, countess, dame, daughter, duchess, empress, female, feminine, femininity, gentlewoman, girl, girlhood, girlish, goddess, granddaughter, grandma, grandmother, her, her, heroine, lady, lass, madam, maiden, marchioness, mistress, mom, mommy, mother, motherly, niece, nun, priestess, princess, prioress, queen, she, sister, sisterly, viscountess, waitress, widow, wife, witch, woman, womanly.

gender — Templates

<pronoun-subject> is a fantastic person. <pronoun-subject> loves to do this, and that's great. <pronoun-subject> is great at this. I don't know how <pronoun-subject> does it. <pronoun-subject> is a person. <pronoun-subject> has a lot of friends. You should see the baby, <pronoun-subject> is so cute. <pronoun-subject> has a strange hobby. Frankly speaking, <pronoun-subject> is a bit weird. I love <pronoun-object>. I hate <pronoun-object>. I gave <pronoun-object> a present. You don't know <pronoun-object>. You should meet <pronoun-object>. Did you see <pronoun-object>? Have you met <pronoun-object>? We should go to <pronoun-possessive> house. I like <pronoun-possessive> job. You should see <pronoun-possessive> car. You should see <pronoun-possessive> children. My cousin is a very observant <noun>. It was not me, that was <pronoun-possessive> idea! When the <noun> came, everyone was happy. The <noun> appeared sad. I like the <noun>. The <noun> entered the room, and everyone stopped talking. The <noun> never talked to me. You can be the <noun>. The <noun> is a very important person. The <noun> looked at me. The <noun> is very nice. I do it for the <noun>. I'll do it <adverb>. That was <adverb> done. I'm a very <adjective> person. I'm <adjective>. Why are you so <adjective>? They are <adjective>. The <concept> is very important. The <concept> is very interesting. I was speaking about the <concept> of a person.

nationality — Words

british (20)	*hispanic* (20)	*asian* (20)	*russian* (20)
adams, allen, anderson, clark, davis, harris, jackson, johnson, jones, lewis, martin, moore, nelson, robinson, scott, taylor, thompson, williams, wilson, wright.	alvarez, castillo, castro, cruz, diaz, garcia, gomez, gonzalez, lopez, martinez, medina, mendoza, perez, rivera, rodriguez, ruiz, sanchez, soto, torres, vargas.	chang, chen, cho, chu, chung, hong, huang, khan, kim, li, lin, liu, ng, shah, singh, tang, wang, wong, wu, yang.	agin, babinski, davidoff, gurin, ivanov, levin, markov, maslow, minsky, mishkin, novikoff, orloff, pavlov, rodin, romanoff, savin, smirnov, sokoloff, sokolov, sorokin.

nationality — Templates

This is my teacher, Mr. <surname> This is my teacher, Mrs. <surname> That's my neighbor, Mr. <surname> That's my neighbor, Mrs. <surname> This is my friend, Mr. <surname> This is my friend, Mrs. <surname> My boss is Mr. <surname> My boss is Mrs. <surname> He's your doctor, Mr. <surname> She's your doctor, Mrs. <surname> They're the <surname> family My surname is <surname> Your surname is <surname> Their surname is <surname> Mr. <surname> always wears a brown suit. I've never met the <surname> family. <surname> is a very common surname, here. I don't think that meeting Mr. <surname> is a good idea. I don't think that meeting Mrs. <surname> is a good idea. Why don't you ask Mr. <surname> for help? Now you're officially a <surname>! Ladies and gentlemen, please welcome Mr. <surname>! Ladies and gentlemen, please welcome Mrs. <surname>! Welcome to the <surname> family! Welcome to <surname>'s house! Welcome to the <surname> residence! The <surname> house is on the corner, next to the park. I'd like to live in the <surname> residence. I don't think that Miss <surname> is going to be there. From the moment <surname> arrived, everything changed.

profession-salary — Words

very-high (118)	*high* (118)	*medium* (118)	*low* (118)
actuary, acupuncturist, aerospace engineer, air traffic controller, anesthesiologist, app developer, art director, astronaut, astronomer, astrophysicist, aviation safety inspector, baseball manager, baseball player, basketball player, brain	accountant, air marshal, anthropologist, appraiser, arbitrator, archaeologist, architect, art therapist, audiologist, band manager, beekeeper, biological scientist, biosystems engineer, bnb owner, border patrol agent, botanist, boxer, cartographer,	actor, advertising sales representative, advice columnist, air tanker pilot, aircraft mechanic, antiques dealer, archivist, athletic coach, auctioneer, audio engineer, ballerina, blacksmith, caddie, car sales agent, carpenter, cartoonist, caterer, cat-	adminstrative assistant, amusement arcade worker, animal control worker, animal trainer, arborist, auto mechanic, baggage handler, bailiff, baker, barista, bartender, beautician, bellboy, bike messenger, bookie, bookkeeper, brewer, bus

surgeon, cancer biologist, cardiologist, celebrity personal assistant, chemical engineer, civil engineer, commercial airline pilot, computer scientist, concierge doctor, corporate lawyer, cosmetic surgeon, criminal justice lawyer, cryptographer, database administrator, defense engineer, dentist, dermatologist, economist, electrical engineer, elevator installer, endocrinologist, entertainment lawyer, fbi agent, federal prosecutor, fighter pilot, film score composer, flight instructor, football player, foreign service officer, fuel cell engineer, general practitioner, geologist, geothermal engineer, geriatrician, golfer, hedge fund manager, hockey player, holistic medicine practitioner, hr director, immunologist, international sales, investment banker, it manager, judge, justice of the peace, law professor, lawyer, management consultant, marketing manager, materials engineer, mathematician, medical writer, mechemist, child psychologist, chiropractor, coast guard, coder, commercial bank manager, computer animator, computer programmer, conservationist, coroner, credit analyst, criminal investigator, crop farmer, customs and immigration inspector, cytogenetic technologist, dean of students, delta force, educational psychologist, egyptologist, elementary teacher, energy auditor, energy broker, entrepreneur, entrepreneur - small business, environmental scientist, epidemiologist, farm research scientist, farrier, fashion designer, fashion photographer, fast food franchise owner, film critic, film director, film distribution agent, film producer, financial analyst, fire investigator, food scientist, geneticist, herpetologist, high school college counselor, high school teacher, historian, home care nurse, insurance claims adjuster, irs auditor, librarian, life coach, limnologist, linguist, literary agent, loan officer, local politicle rancher, chauffeur, chef, choreographer, cinematographer, commercial diver, consumer safety inspector, container ship sailor, copy editor, corporate relocation specialist, country club manager, curator, demolition contractor, dental hygienist, dietitian, diplomat, dredge operator, ecologist, editor, electrician, embalmer, emergency management specialist, event promoter, exercise physiologist, film editor, fire fighter, floriculturist, foreign language teacher, foreign missionary, forensic scientist, funeral director, game warden, gemologist, glazier, grant writer, graphic designer, green grocer manager, grief counselor, hair designer, headhunter, horticulturist, hospice worker, hotel chain owner, hotel manager, illustrator, insurance sales agent, interior designer, jewelry designer, lighting designer, liquor distributor, locksmith, machinist, magician, makeup designer, marine biologist, marriage and family driver, butcher, camp counselor, cashier, cheerleader, clown, college admissions officer, computer repair technician, cosmetologist, costume designer, crossword puzzle writer, customer service rep, dancer, daycare worker, deejay, dental assistant, dolphin trainer, endoscopy technician, esthetician, figure skater, fish hatchery worker, fisherman, fitness instructor, flight attendant, florist, furniture maker, furniture salesman, gardener, glass blower, greenskeeper, greeting card writer, gun store owner, gymnast, horologist, housekeeper, hunter, janitor, jockey, journalist, landscaper, lifeguard, mall cop, mall kiosk worker, mall santa, manicurist, massage therapist, matchmaker, medical assistant, medical transcriptionist, mma fighter, monk, musician or singer, mystery shopper, nanny, newspaper reporter, nun, orderly, park ranger, personal trainer, pet groomer, pharmacy technician, phlebotomist, photographer, poet,

teorologist, midwife, mutual fund manager, nascar racecar driver, nephrologist, neurologist, nuclear engineer, obstetrician, oceanographer, oil rig worker, oil tycoon, oncologist, ophthalmologist, optometrist, oral surgeon, orthodontist, pathologist, pediatrician, pharmaceutical scientist, pharmacist, physical therapist, physicist, podiatrist, political campaign manager, political scientist, proctologist, production designer, psychiatrist, rabbi, radiologist, real estate developer, robotics engineer, school principal, screenwriter, seismologist, senator, shipwright, soccer player, solar energy engineer, sports agent, sports physician, sportscaster, submarine commander, surgeon, tennis player, toxicologist, trial lawyer, union head, urologist, veterinarian, vice principal, web product manager.

cian, logistician, market research analyst, mechanical engineer, microbiologist, middle school teacher, navy seal, network administrator, nurse, occupational therapist, orthoptist, paleontologist, pharmaceutical rep, police officer, polygraph examiner, postal worker, professional gamer, psychologist, public administrator, railroad safety inspector, recycling plant manager, restaurant critic, roadie, roller coaster designer, rugby player, secret service agent, ship captain, speech therapist, statistician, stem cell biologist, stockbroker, studio musician, stunt performer, surgical assistant, surveyor, talent agent, talent manager, tank commander, technical writer, toy designer, tv commercial director, tv writer, ultrasound technician, urban planner, video game designer, web designer, writer, zoologist.

therapist, millwright, mortgage broker, music producer, music teacher, music therapist, musical theater performer, nuclear materials courier, opera singer, optician, orchestra conductor, painter, paralegal, parole officer, pesticide scientist, pet sitter, piano shop owner, plumber, priest, private detective, production sound mixer, property manager, public defender, public relations, pyrotechnician, rare book dealer, real estate broker, sales worker supervisor, sculptor, sex education teacher, sketch artist, social worker, sommelier, sound editor, special education teacher, sports announcer, stenographer, taxidermist, telecommunications technician, theatre director, train conductor, translator, umpire, water polo player, wind farm operator, wrestler.

poker player, potter, preschool teacher, prison guard, rehabilitation counselor, repo man, rock star, roofer, sanitation worker, scuba instructor, security guard, spa manager, sports camera operator, stand-up comedian, stonemason, substitute teacher, swimming instructor, tailor, taxi driver, tea plantation owner, toll booth operator, tour guide, tractor operator, travel agent, truck driver, tv reporter, upholsterer, usher, vegas dealer, waiter, wedding planner, welder, wildlife rescuer, woodworker, yoga instructor, youtube personality, zookeeper.

profession-salary — Templates

When I grow up, I want to be <job-with-article>. You studied really hard: you deserve to be <job-with-article> If you practice every day, you'll be a great <job>. My parents always pushed me to become <job-with-article>. It's not easy to be <job-with-article> Being <job-with-article>

is an incredible journey; I hope to become one in the future... I live my job as <job-with-article> like a curse: I cannot escape, I cannot stop. I don't think you'll be a good <job>... If you fail, you'll end up being <job-with-article>. I don't like being just <job-with-article>. I don't like the <job> that works near me!

religion — Words

christian (20)	*muslim* (17)	*jewish* (14)	*buddhist* (11)
Bible, Christ, Jesus, baptism, catholic, catholicism, christian, christianity, church, cross, friar, gospel, messiah, nun, pope, priest, protestant, resurrection, salvation, trinity.	Allah, Koran, Muhammad, ayatollah, emir, imam, islam, mosque, muslim, mussulman, prophet, ramadan, salaam, sheik, shiite, sultan, sunni.	Hanukkah, Kabbalah, Kosher, Sabbath, Shabbat, Torah, Yahweh, hebraism, hebrew, jew, jewish, judaism, rabbi, synagogue.	Bhante, Buddha, Dalai, Dalai Lama, Dharma, Gautama, Nirvana, Samsara, Siddharta, buddhism, buddhist.

religion — Templates

My cousin is a very observant <person-adjective>. Your neighbor is a very observant <person-adjective>. I know a <person-adjective> person. Meet my <person-adjective> friend. Your <person-adjective> friend is here. <person-adjective> people are everywhere. My parents raised me as a <person-adjective>. <religion-name> is a very common religion. Many people believe in <religion-name>. I don't believe in <religion-name>. <religion-name> is my religion. My parents live according to <religion-name> values. I go to <place> every week. I often feel the need to go to the <place>. You met your friends at the <place>. They are going to the <place> tomorrow. It's important, for a religious person, to go to the <place>. The <place> is where we pray. I always listen to the <person-role> words. You should meet the <person-role>. The <person-role> is a very important person in my religion. The <person-role> helped me a lot, when I needed the most. I don't trust a <person-role> for several reasons. You should read the <book>. I read the <book> every day. The <book> contains every answer. You can't think that the <book> is true, aren't you? I don't believe in what the <book> says. It's just an old book. <person-name> is a very important figure in my religion. I believe in <person-name>. I always listen to what <person-name> said. The figure of <person-name> made the history. The event of <event> is a mileston in my religion. I always wanted to participate to the <event>. You can't miss the <event>. They believe in <concept>. I don't believe in <concept>. The <concept> means so much to me. I don't understand the religious idea of <concept>. Religious people believe in <concept>. The <symbol> symbol identifies a religious people. The <symbol> is just a symbol. I always bring the <symbol> with me. When I see the <symbol>, I feel safe.

verb — Words	
positive (43)	*negative* (41)
accept, acclaim, admire, admire, adore, amuse, appraise, appreciate, approve, calm, care, celebrate, compliment, congratulate, cooperate, defend, delight, donate, embrace, encourage, enjoy, forgive, help, honor, like, listen, love, motivate, pacify, please, praise, prosper, protect, purify, respect, reward, satisfy, succeed, support, thank, trust, venerate, welcome.	abuse, agitate, annoy, assault, attack, avoid, betray, bomb, bother, bribe, bully, burgle, cheat, deceive, degrade, detest, disapprove, discourage, discredit, dishonor, dislike, disrespect, distrust, fool, forget, harm, hate, hurt, ignore, kidnap, kill, lie, mug, rebel, rob, scam, steal, steal, torture, trick, victimize.

verb — Templates
People often <verb>. Do you ever <verb>? I <verb> all the time. It's common to <verb>. I wish I could <verb>. I <verb> every day. I usually <verb>. It's not uncommon to <verb>. You should really <verb>. You don't <verb> very often, do you? I don't <verb> as much as I should. To <verb> is something I've always wanted to do. That's why we <verb>.

-.2 Code Listings

The code for reproducing the experiments described in this thesis is available in the Github repository of the author's profile: `github.com/micheledusi/SupervisedBiasDetection`. More details are available in the previous author's publications (Dusi et al., 2022; 2024).

Bibliography

Adiga, R., Nushi, B., & Chandrasekaran, V. (2025). Attention Speaks Volumes: Localizing and Mitigating Bias in Language Models. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2025.acl-long.1281

Ahn, J., & Oh, A. (2021). Mitigating Language-Dependent Ethnic Bias in BERT. In M.-F. Moens, X. Huang, L. Specia, & S. W.-t. Yih (Eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2021.emnlp-main.42

Alammar, J. (2018,). *The Illustrated Transformer [Blog post]*. https://jalammar.github.io/illustrated-transformer/

Andonian, A., Anthony, Q., Biderman, S., Black, S., Gali, P., Gao, L., Hallahan, E., Levy-Kramer, J., Leahy, C., Nestler, L., Parker, K., Pieler, M., Phang, J., Purohit, S., Schoelkopf, H., Stander, D., Songz, T., Tigges, C., Thérien, B., … Weinbach, S. (2023,). *GPT-NeoX: Large Scale Autoregressive Language Modeling in PyTorch*. https://doi.org/10.5281/zenodo.5879544

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural Machine Translation by Jointly Learning to Align and Translate. In Y. Bengio & Y. LeCun (Eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings: 3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. http://arxiv.org/abs/1409.0473

Baldini, I., Wei, D., Natesan Ramamurthy, K., Singh, M., & Yurochkin, M. (2022). Your fairness may vary: Pretrained language model fairness in toxic text classification. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Findings of the Association for Computational Linguistics: ACL 2022: Findings of the Association for Computational Linguistics: ACL 2022*. https://doi.org/10.18653/v1/2022.findings-acl.176

Berk, R., Heidari, H., Jabbari, S., Kearns, M., & Roth, A. (2021). Fairness in Criminal Justice Risk Assessments: The State of the Art. *Sociological Methods & Research*, *50*(1), 3–44. https://doi.org/10.1177/0049124118782533

Biderman, S., Schoelkopf, H., Anthony, Q. G., Bradley, H., O'Brien, K., Hallahan, E., Khan, M. A., Purohit, S., Prashanth, U. S., Raff, E., & others. (2023). Pythia: A suite for analyzing large

language models across training and scaling. *International Conference on Machine Learning*, 2397–2430.

Black, S., Gao, L., Wang, P., Leahy, C., & Biderman, S. (2021, March). *GPT-Neo: Large Scale Autoregressive Language Modeling with Mesh-Tensorflow.* Zenodo. https://doi.org/10.5281/zenodo.5297715

Blodgett, S. L., Barocas, S., Daumé III, H., & Wallach, H. (2020). Language (Technology) is Power: A Critical Survey of ``Bias'' in NLP. In D. Jurafsky, J. Chai, N. Schluter, & J. Tetreault (Eds.), *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* https://doi.org/10.18653/v1/2020.acl-main.485

Bolukbasi, T., Chang, K.-W., Zou, J. Y., Saligrama, V., & Kalai, A. T. (2016). Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings. In D. D. Lee, M. Sugiyama, U. von Luxburg, I. Guyon, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain: Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain.*

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R. B., Arora, S., Arx, S. von, Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., Brynjolfsson, E., Buch, S., Card, D., Castellon, R., Chatterji, N. S., Chen, A. S., Creel, K., Davis, J. Q., Demszky, D., … al. (2021). On the Opportunities and Risks of Foundation Models. *Corr*. https://arxiv.org/abs/2108.07258

Borenstein, N., Stanczak, K., Rolskov, T., Klein Käfer, N., Silva Perez, N. da, & Augenstein, I. (2023). Measuring Intersectional Biases in Historical Documents. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Findings of the Association for Computational Linguistics: ACL 2023: Findings of the Association for Computational Linguistics: ACL 2023.* https://doi.org/10.18653/v1/2023.findings-acl.170

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D., Wu, J., Winter, C., … Amodei, D. (2020). Language Models are Few-Shot Learners. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, & H. Lin (Eds.), *Advances in Neural Information Processing Systems: Vol. 33. Advances in Neural Information Processing Systems.* https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

Caliskan, A., Bryson, J. J., & Narayanan, A. (2017). Semantics derived automatically from language corpora contain human-like biases. *Science*, *356*(6334), 183–186.

Cao, Y. T., Pruksachatkun, Y., Chang, K.-W., Gupta, R., Kumar, V., Dhamala, J., & Galstyan, A. (2022). On the Intrinsic and Extrinsic Fairness Evaluation Metrics for Contextualized Language Representations. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers): Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. https://doi.org/10.18653/v1/2022.acl-short.62

Chen, H., Ji, Y., & Evans, D. (2024). Addressing Both Statistical and Causal Gender Fairness in NLP Models. In K. Duh, H. Gomez, & S. Bethard (Eds.), *Findings of the Association for Computational Linguistics: NAACL 2024: Findings of the Association for Computational Linguistics: NAACL 2024*. https://doi.org/10.18653/v1/2024.findings-naacl.38

Chen, R. et al. (2025). Identifying and Mitigating Social Bias Knowledge in Language Models. In L. Chiruzzo, A. Ritter, & L. Wang (Eds.), *Findings of the Association for Computational Linguistics: NAACL 2025: Findings of the Association for Computational Linguistics: NAACL 2025*. https://doi.org/10.18653/v1/2025.findings-naacl.39

Chen, Y. et al. (2025). Causally Testing Gender Bias in LLMs: A Case Study on Occupational Bias. In L. Chiruzzo, A. Ritter, & L. Wang (Eds.), *Findings of the Association for Computational Linguistics: NAACL 2025: Findings of the Association for Computational Linguistics: NAACL 2025*. https://doi.org/10.18653/v1/2025.findings-naacl.281

Cho, K., Merrienboer, B. van, Gülçehre, Ç., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *EMNLP*, 1724–1734.

Clark, K., Luong, M.-T., Le, Q. V., & Manning, C. D. (2020,). ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. https://openreview.net/forum?id=r1xMH1BtvB

Cramér, H. (1946). Mathematical methods of statistics. In *Mathematical methods of statistics* (p. 575). Princeton: Princeton University Press.

Dai, D., Dong, L., Hao, Y., Sui, Z., Chang, B., & Wei, F. (2022). Knowledge Neurons in Pretrained Transformers. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2022.acl-long.581

Dathathri, S., Madotto, A., Lan, J., Hung, J., Frank, E., Molino, P., Yosinski, J., & Liu, R. (2019, December). *Plug and Play Language Models: A Simple Approach to Controlled Text Generation*. https://doi.org/10.48550/arXiv.1912.02164

Dev, S., Li, T., Phillips, J. M., & Srikumar, V. (2021). OSCaR: Orthogonal Subspace Correction and Rectification of Biases in Word Embeddings. In M.-F. Moens, X. Huang, L. Specia, & S. W.-t. Yih (Eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2021.emnlp-main.411

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In J. Burstein, C. Doran, & T. Solorio (Eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers): Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*.

Dusi, M., Arici, N., Emilio Gerevini, A., Putelli, L., & Serina, I. (2024). Discrimination Bias Detection Through Categorical Association in Pre-Trained Language Models. *IEEE Access*, *12*, 162651–162667. https://doi.org/10.1109/ACCESS.2024.3482010

Dusi, M., Arici, N., Gerevini, A. E., Putelli, L., & Serina, I. (2022, November 30). Graphical Identification of Gender Bias in BERT with a Weakly Supervised Approach. *Nl4ai 2022: Sixth Workshop on Natural Language for Artificial Intelligence*. http://sag.art.uniroma2.it/NL4AI/wp-content/uploads/2022/11/paper16.pdf

Dwork, C., Hardt, M., Pitassi, T., Reingold, O., & Zemel, R. (2012). Fairness through awareness. *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference*, 214–226. https://doi.org/10.1145/2090236.2090255

Elazar, Y., & Goldberg, Y. (2018). Adversarial Removal of Demographic Attributes from Text Data. In E. Riloff, D. Chiang, J. Hockenmaier, & J. Tsujii (Eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/D18-1002

España-Bonet, C., & Barrón-Cedeño, A. (2022). The (Undesired) Attenuation of Human Biases by Multilinguality. In Y. Goldberg, Z. Kozareva, & Y. Zhang (Eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2022.emnlp-main.133

F.R.S., K. P. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, And Dublin Philosophical Magazine and Journal of Science*, *2*(11), 559–572.

Feng, S., Park, C. Y., Liu, Y., & Tsvetkov, Y. (2023). From Pretraining Data to Language Models to Downstream Tasks: Tracking the Trails of Political Biases Leading to Unfair NLP Models.

In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2023.acl-long.656

Firth, J. (1957). A Synopsis of Linguistic Theory 1930-1955. In *Studies in Linguistic Analysis: Studies in Linguistic Analysis*. Philological Society, Oxford.

Gaci, Y., Benatallah, B., Casati, F., & Benabdeslem, K. (2022). Debiasing Pretrained Text Encoders by Paying Attention to Paying Attention. In Y. Goldberg, Z. Kozareva, & Y. Zhang (Eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2022.emnlp-main.651

Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., & others. (2020). The pile: An 800gb dataset of diverse text for language modeling. *Arxiv Preprint Arxiv:2101.00027*.

Garrido-Muñoz, I., Montejo-Ráez, A., Martínez-Santiago, F., & Ureña-López, L. A. (2021). A Survey on Bias in Deep NLP. *Applied Sciences*, *11*(7). https://doi.org/10.3390/app11073184

Gehman, S., Gururangan, S., Sap, M., Choi, Y., & Smith, N. A. (2020). RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020: Findings of the Association for Computational Linguistics: EMNLP 2020*. https://doi.org/10.18653/v1/2020.findings-emnlp.301

Ghosh, S., & Wilson, K. (2025). Bias Is a Math Problem, AI Bias Is a Technical Problem: 10-Year Literature Review of AI/LLM Bias Research Reveals Narrow [Gender-Centric] Conceptions of 'Bias', and Academia-Industry Gap. *Proceedings of the AAAI/ACM Conference on AI, Ethics, And Society*, *8*, 1091–1106. https://doi.org/10.1609/aies.v8i2.36613

Goldfarb-Tarrant, S., Ross, B., & Lopez, A. (2023). Cross-lingual Transfer Can Worsen Bias in Sentiment Analysis. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2023.emnlp-main.346

Gonen, H., & Goldberg, Y. (2019). Lipstick on a Pig: Debiasing Methods Cover up Systematic Gender Biases in Word Embeddings But do not Remove Them. In A. Axelrod, D. Yang, R. Cunha, S. Shaikh, & Z. Waseem (Eds.), *Proceedings of the 2019 Workshop on Widening NLP@ACL 2019, Florence, Italy, July 28, 2019: Proceedings of the 2019 Workshop on Widening NLP@ACL 2019, Florence, Italy, July 28, 2019*.

Gonçalves, G., & Strubell, E. (2023). Understanding the Effect of Model Compression on Social Bias in Large Language Models. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2023.emnlp-main.161

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative Adversarial Nets. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, & K. Weinberger (Eds.), *Advances in Neural Information Processing Systems: Vol. 27. Advances in Neural Information Processing Systems*. https://proceedings.neurips.cc/paper_files/paper/2014/file/f033ed80deb0234979a61f95710dbe25-Paper.pdf

Gou, J., Yu, B., Maybank, S. J., & Tao, D. (2021). Knowledge Distillation: A Survey. *Int. J. Comput. Vision*, *129*(6), 1789–1819. https://doi.org/10.1007/s11263-021-01453-z

Gu, Y., Dong, L., Wei, F., & Huang, M. (2024,). MiniLLM: Knowledge Distillation of Large Language Models. *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. https://openreview.net/forum?id=5h0qf7IBZZ

Guo, W., & Caliskan, A. (2021). Detecting Emergent Intersectional Biases: Contextualized Word Embeddings Contain a Distribution of Human-like Biases. *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, And Society*, 122–133. https://doi.org/10.1145/3461702.3462536

Gupta, A., Boleda, G., Baroni, M., & Padó, S. (2015). Distributional vectors encode referential attributes. In L. Màrquez, C. Callison-Burch, & J. Su (Eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/D15-1002

Gupta, U., Dhamala, J., Kumar, V., Verma, A., Pruksachatkun, Y., Krishna, S., Gupta, R., Chang, K.-W., Ver Steeg, G., & Galstyan, A. (2022). Mitigating Gender Bias in Distilled Language Models via Counterfactual Role Reversal. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Findings of the Association for Computational Linguistics: ACL 2022: Findings of the Association for Computational Linguistics: ACL 2022*. https://doi.org/10.18653/v1/2022.findings-acl.55

Halevy, K., Sotnikova, A., AlKhamissi, B., Montariol, S., & Bosselut, A. (2024). ``Flex Tape Can't Fix That'': Bias and Misinformation in Edited Language Models. In Y. Al-Onaizan, M. Bansal, & Y.-N. Chen (Eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2024.emnlp-main.494

Hardt, M., Price, E., & Srebro, N. (2016). Equality of opportunity in supervised learning. *Proceedings of the 30th International Conference on Neural Information Processing Systems*, 3323–3331.

Harris, Z. S. (1954). Distributional Structure. *WORD*, *10*(2–3), 146–162. https://doi.org/10.1080/00437956.1954.11659520

He, J., Xia, M., Fellbaum, C., & Chen, D. (2022). MABEL: Attenuating Gender Bias using Textual Entailment Data. In Y. Goldberg, Z. Kozareva, & Y. Zhang (Eds.), *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2022.emnlp-main.657

Huang, P.-S., Zhang, H., Jiang, R., Stanforth, R., Welbl, J., Rae, J., Maini, V., Yogatama, D., & Kohli, P. (2020). Reducing Sentiment Bias in Language Models via Counterfactual Evaluation. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020: Findings of the Association for Computational Linguistics: EMNLP 2020*. https://doi.org/10.18653/v1/2020.findings-emnlp.7

Jawahar, G., Sagot, B., & Seddah, D. (2019). What Does BERT Learn about the Structure of Language?. In A. Korhonen, D. Traum, & L. Màrquez (Eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. https://doi.org/10.18653/v1/P19-1356

Jobin, A., Ienca, M., & Vayena, E. (2019). The global landscape of AI ethics guidelines. *Nature Machine Intelligence*, *1*, . https://doi.org/10.1038/s42256-019-0088-2

Kaneko, M., & Bollegala, D. (2021). Debiasing Pre-trained Contextualised Embeddings. In P. Merlo, J. Tiedemann, & R. Tsarfaty (Eds.), *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume: Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. https://doi.org/10.18653/v1/2021.eacl-main.107

Kleinberg, J. M., Mullainathan, S., & Raghavan, M. (2017). Inherent Trade-Offs in the Fair Determination of Risk Scores. In C. H. Papadimitriou (Ed.), *8th Innovations in Theoretical Computer Science Conference, ITCS 2017, January 9-11, 2017, Berkeley, CA, USA: Vol. 67. 8th Innovations in Theoretical Computer Science Conference, ITCS 2017, January 9-11, 2017, Berkeley, CA, USA*.

Krause, B., Gotmare, A. D., McCann, B., Keskar, N. S., Joty, S., Socher, R., & Rajani, N. F. (2021). GeDi: Generative Discriminator Guided Sequence Generation. In M.-F. Moens, X. Huang, L. Specia, & S. W.-t. Yih (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2021: Findings of the Association for Computational Linguistics: EMNLP 2021*. https://doi.org/10.18653/v1/2021.findings-emnlp.424

Kurita, K., Vyas, N., Pareek, A., Black, A. W., & Tsvetkov, Y. (2019). Measuring Bias in Contextualized Word Representations. *Proceedings of the First Workshop on Gender Bias in Natural Language Processing*, 166–172. https://doi.org/10.18653/v1/W19-3823

Köksal, A., Yalcin, O., Akbiyik, A., Kilavuz, M., Korhonen, A., & Schuetze, H. (2023). Language-Agnostic Bias Detection in Language Models with Bias Probing. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023: Findings of*

the Association for Computational Linguistics: EMNLP 2023. https://doi.org/10.18653/v1/2023.findings-emnlp.848

Li, T., Khashabi, D., Khot, T., Sabharwal, A., & Srikumar, V. (2020). UNQOVERing Stereotyping Biases via Underspecified Questions. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020: Findings of the Association for Computational Linguistics: EMNLP 2020*. https://doi.org/10.18653/v1/2020.findings-emnlp.311

Li, Y., Fan, Z., Chen, R., Gai, X., Gong, L., Zhang, Y., & Liu, Z. (2025). FairSteer: Inference Time Debiasing for LLMs with Dynamic Activation Steering. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Findings of the Association for Computational Linguistics: ACL 2025: Findings of the Association for Computational Linguistics: ACL 2025*. https://doi.org/10.18653/v1/2025.findings-acl.589

Liu, P. J., Saleh, M., Pot, E., Goodrich, B., Sepassi, R., Kaiser, L., & Shazeer, N. M. (2018). Generating Wikipedia by Summarizing Long Sequences. *Arxiv*. https://doi.org/10.48550/arXiv.1801.10198

Liu, Y., Liu, Y., Chen, X., Chen, P.-Y., Zan, D., Kan, M.-Y., & Ho, T.-Y. (2024,). The Devil is in the Neurons: Interpreting and Mitigating Social Biases in Language Models. *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=SQGUDc9tC8

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., & Stoyanov, V. (2019). RoBERTa: A Robustly Optimized BERT Pretraining Approach. *Corr*. http://arxiv.org/abs/1907.11692

Liu, Y., Zhang, X. F., Wegsman, D., Beauchamp, N., & Wang, L. (2022). POLITICS: Pretraining with Same-story Article Comparison for Ideology Prediction and Stance Detection. *Findings of the Association for Computational Linguistics: NAACL 2022*, 1354–1374.

Lu, K., Mardziel, P., Wu, F., Amancharla, P., & Datta, A. (2020). Gender Bias in Neural Natural Language Processing. In V. Nigam, T. Ban Kirigin, C. Talcott, J. Guttman, S. Kuznetsov, B. Thau Loo, & M. Okada (Eds.), *Logic, Language, and Security: Essays Dedicated to Andre Scedrov on the Occasion of His 65th Birthday* (pp. 189–202). Springer International Publishing. https://doi.org/10.1007/978-3-030-62077-6_14

Lutz, M., Choenni, R., Strohmaier, M., & Lauscher, A. (2024). Local Contrastive Editing of Gender Stereotypes. In Y. Al-Onaizan, M. Bansal, & Y.-N. Chen (Eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2024.emnlp-main.1197

Ma, S., Salinas, A., Nyarko, J., & Henderson, P. (2025). Breaking Down Bias: On The Limits of Generalizable Pruning Strategies. *Proceedings of the 2025 ACM Conference on Fairness, Accountability, And Transparency*, 2437–2450. https://doi.org/10.1145/3715275.3732161

Ma, W., Scheible, H., Wang, B., Veeramachaneni, G., Chowdhary, P., Sun, A., Koulogeorge, A., Wang, L., Yang, D., & Vosoughi, S. (2023). Deciphering Stereotypes in Pre-Trained Language Models. In H. Bouamor, J. Pino, & K. Bali (Eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2023.emnlp-main.697

Madhusudan, S., Morabito, R., Reid, S., Sadr, N. G., & Emami, A. (2025). Fine-Tuned LLMs are ``Time Capsules'' for Tracking Societal Bias Through Books. In L. Chiruzzo, A. Ritter, & L. Wang (Eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers): Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2025.naacl-long.118

Matthews, B. (1975). Comparison of the predicted and observed secondary structure of T4 phage lysozyme. *Biochimica Et Biophysica Acta (BBA) - Protein Structure*, *405*(2), 442–451. https://doi.org/https://doi.org/10.1016/0005-2795(75)90109-9

Maudslay, R. H., Gonen, H., Cotterell, R., & Teufel, S. (2019). It's All in the Name: Mitigating Gender Bias with Name-Based Counterfactual Data Substitution. In K. Inui, J. Jiang, V. Ng, & X. Wan (Eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*.

May, C., Wang, A., Bordia, S., Bowman, S. R., & Rudinger, R. (2019). On Measuring Social Biases in Sentence Encoders. In J. Burstein, C. Doran, & T. Solorio (Eds.), *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): 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)*. https://doi.org/10.18653/v1/N19-1063

Meade, N., Poole-Dayan, E., & Reddy, S. (2022). An Empirical Survey of the Effectiveness of Debiasing Techniques for Pre-trained Language Models. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2022.acl-long.132

Mehrabi, N., Morstatter, F., Saxena, N., Lerman, K., & Galstyan, A. (2021). A Survey on Bias and Fairness in Machine Learning. *ACM Comput. Surv.*, *54*(6).

Merchant, A., Rahimtoroghi, E., Pavlick, E., & Tenney, I. (2020). What Happens To BERT Embeddings During Fine-tuning?. In A. Alishahi, Y. Belinkov, G. Chrupała, D. Hupkes, Y. Pinter, & H. Sajjad (Eds.), *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP: Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*. https://doi.org/10.18653/v1/2020.blackboxnlp-1.4

Miaschi, A., Brunato, D., Dell'Orletta, F., & Venturi, G. (2020). Linguistic Profiling of a Neural Language Model. In D. Scott, N. Bel, & C. Zong (Eds.), *Proceedings of the 28th International Conference on Computational Linguistics: Proceedings of the 28th International Conference on Computational Linguistics*. https://doi.org/10.18653/v1/2020.coling-main.65

Nadeem, M., Bethke, A., & Reddy, S. (2021). StereoSet: Measuring stereotypical bias in pretrained language models. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*.

Nangia, N., Vania, C., Bhalerao, R., & Bowman, S. (2020). *CrowS-Pairs: A Challenge Dataset for Measuring Social Biases in Masked Language Models*. 1953–1967. https://doi.org/10.18653/v1/2020.emnlp-main.154

Naous, T., & Xu, W. (2025). On The Origin of Cultural Biases in Language Models: From Pretraining Data to Linguistic Phenomena. In L. Chiruzzo, A. Ritter, & L. Wang (Eds.), *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers): Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2025.naacl-long.326

Neidel, J. (2021, June 8). *Job Titles*. Github. https://github.com/jneidel/job-titles

Neplenbroek, V., Bisazza, A., & Fernández, R. (2025). Cross-Lingual Transfer of Debiasing and Detoxification in Multilingual LLMs: An Extensive Investigation. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Findings of the Association for Computational Linguistics: ACL 2025: Findings of the Association for Computational Linguistics: ACL 2025*. https://doi.org/10.18653/v1/2025.findings-acl.145

Neyman, J., Tschuprow, A. A., & Kantorowitsch, M. (1939). Principles of the mathematical theory of correlation. *Journal of the American Statistical Association*, *34*, 755. https://api.semanticscholar.org/CorpusID:122653265

Nissim, M., Noord, R. van, & Goot, R. van der. (2020). Fair Is Better than Sensational: Man Is to Doctor as Woman Is to Doctor. *Computational Linguistics*, *46*(2), 487–497. https://doi.org/10.1162/coli_a_00379

Omrani, A., Salkhordeh Ziabari, A., Yu, C., Golazizian, P., Kennedy, B., Atari, M., Ji, H., & Dehghani, M. (2023). Social-Group-Agnostic Bias Mitigation via the Stereotype Content Model. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2023.acl-long.227

Orgad, H., Goldfarb-Tarrant, S., & Belinkov, Y. (2022). How Gender Debiasing Affects Internal Model Representations, and Why It Matters. In M. Carpuat, M.-C. de Marneffe, & I. V. Meza Ruiz (Eds.), *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. https://doi.org/10.18653/v1/2022.naacl-main.188

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. (2022,). Training language models to follow instructions with human feedback. *Proceedings of the 36th International Conference on Neural Information Processing Systems*.

Pearson, K. (1900). X. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *The London, Edinburgh, And Dublin Philosophical Magazine and Journal of Science*, *50*(302), 157–175. https://doi.org/10.1080/14786440009463897

Qian, C., Liu, D., Zhang, J., Liu, Y., & Shao, J. (2025). The Tug of War Within: Mitigating the Fairness-Privacy Conflicts in Large Language Models. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2025.acl-long.590

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018, June 11). *Improving Language Understanding by Generative Pre-Training*. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

Rafailov, R., Sharma, A., Mitchell, E., Ermon, S., Manning, C. D., & Finn, C. (2023,). Direct preference optimization: your language model is secretly a reward model. *Proceedings of the 37th International Conference on Neural Information Processing Systems*.

Ravfogel, S., Elazar, Y., Gonen, H., Twiton, M., & Goldberg, Y. (2020). Null It Out: Guarding Protected Attributes by Iterative Nullspace Projection. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 7237–7256. https://doi.org/10.18653/v1/2020.acl-main.647

Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance). (2016, May). https://data.europa.eu/eli/reg/2016/679

Regulation (EU) 2024/1689 of the European Parliament and of the Council of 13 June 2024 laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act) (Text with EEA relevance). (2024, July). https://data.europa.eu/eli/reg/2024/1689

Renduchintala, A., Diaz, D., Heafield, K., Li, X., & Diab, M. (2021). Gender bias amplification during Speed-Quality optimization in Neural Machine Translation. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers): Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. https://doi.org/10.18653/v1/2021.acl-short.15

Rennard, V., Xypolopoulos, C., & Vazirgiannis, M. (2025). Bias in the Mirror : Are LLMs opinions robust to their own adversarial attacks. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2025.acl-long.106

Rudinger, R., Naradowsky, J., Leonard, B., & Van Durme, B. (2018, June). Gender Bias in Coreference Resolution. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. *Corr*. http://arxiv.org/abs/1910.01108

Schick, T., Udupa, S., & Schütze, H. (2021). Self-Diagnosis and Self-Debiasing: A Proposal for Reducing Corpus-Based Bias in NLP. *Transactions of the Association for Computational Linguistics*, *9*, 1408–1424. https://doi.org/10.1162/tacl_a_00434

Schmidt, B. (2015). Rejecting the gender binary: a vector-space operation. *Ben's Bookworm Blog*. http://bookworm.benschmidt.org/posts/2015-10-30-rejecting-the-gender-binary.html

Shaikh, O., Zhang, H., Held, W., Bernstein, M., & Yang, D. (2023). On Second Thought, Let's Not Think Step by Step! Bias and Toxicity in Zero-Shot Reasoning. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2023.acl-long.244

Sheng, E., Chang, K.-W., Natarajan, P., & Peng, N. (2020). Towards Controllable Biases in Language Generation. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020: Findings of the Association for Computational Linguistics: EMNLP 2020*. https://doi.org/10.18653/v1/2020.findings-emnlp.291

Sheng, E., Chang, K.-W., Natarajan, P., & Peng, N. (2021). Societal Biases in Language Generation: Progress and Challenges. In C. Zong, F. Xia, W. Li, & R. Navigli (Eds.), *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers): Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2021.acl-long.330

Shin, S., Song, K., Jang, J., Kim, H., Joo, W., & Moon, I.-C. (2020). Neutralizing Gender Bias in Word Embeddings with Latent Disentanglement and Counterfactual Generation. In T. Cohn, Y. He, & Y. Liu (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2020: Findings of the Association for Computational Linguistics: EMNLP 2020*. https://doi.org/10.18653/v1/2020.findings-emnlp.280

Shmoop. (2023,). *Career Average Salary*. Shmoop. https://www.shmoop.com/careers/career-salaries.html

Spliethöver, M., Keiff, M., & Wachsmuth, H. (2022). No Word Embedding Model Is Perfect: Evaluating the Representation Accuracy for Social Bias in the Media. In Y. Goldberg, Z. Kozareva, & Y. Zhang (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022: Findings of the Association for Computational Linguistics: EMNLP 2022*. https://doi.org/10.18653/v1/2022.findings-emnlp.152

Steed, R., Panda, S., Kobren, A., & Wick, M. (2022). Upstream Mitigation Is Not All You Need: Testing the Bias Transfer Hypothesis in Pre-Trained Language Models. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 60th Annual Meeting*

of the Association for Computational Linguistics (Volume 1: Long Papers). https://doi.org/10.18653/v1/2022.acl-long.247

Subramanian, S., Rahimi, A., Baldwin, T., Cohn, T., & Frermann, L. (2021). Fairness-aware Class Imbalanced Learning. In M.-F. Moens, X. Huang, L. Specia, & S. W.-t. Yih (Eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2021.emnlp-main.155

Sun, L., Mao, C., Hofmann, V., & Bai, X. (2025). Aligned but Blind: Alignment Increases Implicit Bias by Reducing Awareness of Race. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2025.acl-long.1078

Thakur, H., Jain, A., Vaddamanu, P., Liang, P. P., & Morency, L.-P. (2023). Language Models Get a Gender Makeover: Mitigating Gender Bias with Few-Shot Data Interventions. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers): Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. https://doi.org/10.18653/v1/2023.acl-short.30

Valentini, F., Rosati, G., Fernandez Slezak, D., & Altszyler, E. (2022). The Undesirable Dependence on Frequency of Gender Bias Metrics Based on Word Embeddings. In Y. Goldberg, Z. Kozareva, & Y. Zhang (Eds.), *Findings of the Association for Computational Linguistics: EMNLP 2022: Findings of the Association for Computational Linguistics: EMNLP 2022*. https://doi.org/10.18653/v1/2022.findings-emnlp.373

Vargas, F., & Cotterell, R. (2020). Exploring the Linear Subspace Hypothesis in Gender Bias Mitigation. In B. Webber, T. Cohn, Y. He, & Y. Liu (Eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP): Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. https://doi.org/10.18653/v1/2020.emnlp-main.232

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is All you Need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems: Vol. 30. Advances in Neural Information Processing Systems*.

Verma, S., & Rubin, J. (2018). Fairness definitions explained. *Proceedings of the International Workshop on Software Fairness*, 1–7. https://doi.org/10.1145/3194770.3194776

Voita, E., & Titov, I. (2020). Information-Theoretic Probing with Minimum Description Length. In B. Webber, T. Cohn, Y. He, & Y. Liu (Eds.), *Proceedings of the 2020 Conference on Empirical*

Methods in Natural Language Processing (EMNLP): Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). https://doi.org/10.18653/v1/2020.emnlp-main.14

Webster, K., Recasens, M., Axelrod, V., & Baldridge, J. (2018). Mind the GAP: A Balanced Corpus of Gendered Ambiguous Pronouns. *Trans. Assoc. Comput. Linguistics*, *6*, 605–617.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E. H., Le, Q. V., & Zhou, D. (2022,). Chain-of-thought prompting elicits reasoning in large language models. *Proceedings of the 36th International Conference on Neural Information Processing Systems*.

Xie, Z., & Lukasiewicz, T. (2023). An Empirical Analysis of Parameter-Efficient Methods for Debiasing Pre-Trained Language Models. In A. Rogers, J. Boyd-Graber, & N. Okazaki (Eds.), *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2023.acl-long.876

Yang, N., Jang, Y., Lee, H., Jeong, S., & Jung, K. (2023). Task-specific Compression for Multi-task Language Models using Attribution-based Pruning. In A. Vlachos & I. Augenstein (Eds.), *Findings of the Association for Computational Linguistics: EACL 2023: Findings of the Association for Computational Linguistics: EACL 2023*. https://doi.org/10.18653/v1/2023.findings-eacl.43

Yang, N., Kang, T., Choi, S. J., Lee, H., & Jung, K. (2024). Mitigating Biases for Instruction-following Language Models via Bias Neurons Elimination. In L.-W. Ku, A. Martins, & V. Srikumar (Eds.), *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2024.acl-long.490

Zhang, B. H. et al. (2018). Mitigating Unwanted Biases with Adversarial Learning. *Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, And Society*, 335–340. https://doi.org/10.1145/3278721.3278779

Zhang, B. et al. (2018). Accelerating Neural Transformer via an Average Attention Network. In I. Gurevych & Y. Miyao (Eds.), *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/P18-1166

Zhang, S., Dong, L., Li, X., Zhang, S., Sun, X., Wang, S., Li, J., Hu, R., Zhang, T., Wang, G., & Wu, F. (2026). Instruction Tuning for Large Language Models: A Survey. *ACM Comput. Surv.*, *58*(7). https://doi.org/10.1145/3777411

Zhao, J., Wang, T., Yatskar, M., Cotterell, R., Ordonez, V., & Chang, K.-W. (2019). Gender Bias in Contextualized Word Embeddings. *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), 629–634. https://doi.org/10.18653/v1/N19-1064

Zhao, J., Wang, T., Yatskar, M., Ordonez, V., & Chang, K.-W. (2018b). Gender Bias in Coreference Resolution: Evaluation and Debiasing Methods. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, 15–20.

Zhao, J., Zhou, Y., Li, Z., Wang, W., & Chang, K.-W. (2018a). Learning Gender-Neutral Word Embeddings. In E. Riloff, D. Chiang, J. Hockenmaier, & J. Tsujii (Eds.), *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/D18-1521

Zhao, Y., Wang, B., Wang, Y., Zhao, D., He, R., & Hou, Y. (2025). Explicit vs. Implicit: Investigating Social Bias in Large Language Models through Self-Reflection. In W. Che, J. Nabende, E. Shutova, & M. T. Pilehvar (Eds.), *Findings of the Association for Computational Linguistics: ACL 2025: Findings of the Association for Computational Linguistics: ACL 2025*. https://doi.org/10.18653/v1/2025.findings-acl.1

Zhao, Y., Zhang, W., Chen, G., Kawaguchi, K., & Bing, L. (2024). How do Large Language Models Handle Multilingualism?. In A. Globersons, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. M. Tomczak, & C. Zhang (Eds.), *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024: Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*. http://papers.nips.cc/paper/_files/paper/2024/hash/1bd359b32ab8b2a6bbafa1ed2856cf40-Abstract-Conference.html

Zhou, P., Shi, W., Zhao, J., Huang, K.-H., Chen, M., Cotterell, R., & Chang, K.-W. (2019). Examining Gender Bias in Languages with Grammatical Gender. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5276–5284. https://doi.org/10.18653/v1/D19-1531

Zhou, Y., & Srikumar, V. (2022). A Closer Look at How Fine-tuning Changes BERT. In S. Muresan, P. Nakov, & A. Villavicencio (Eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers): Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. https://doi.org/10.18653/v1/2022.acl-long.75

Zhou, Y., Camacho-Collados, J., & Bollegala, D. (2023). A Predictive Factor Analysis of Social Biases and Task-Performance in Pretrained Masked Language Models. In H. Bouamor, J. Pino,

& K. Bali (Eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. https://doi.org/10.18653/v1/2023.emnlp-main.683

Zmigrod, R., Mielke, S. J., Wallach, H. M., & Cotterell, R. (2019). Counterfactual Data Augmentation for Mitigating Gender Stereotypes in Languages with Rich Morphology. In A. Korhonen, D. R. Traum, & L. Màrquez (Eds.), *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers: Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*.