MEASURING AND MITIGATING GENDER BIAS IN LEGAL CONTEXTUALIZED LANGUAGE MODELS

A Thesis submitted

In Partial Fulfillment of the Requirements

for the Degree of

MASTER OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

by

Ananya Nayak

(Roll No. 2K23/CSE/04)

Under the supervision of

Prof. Shailender Kumar

Professor, Department of Computer Science & Engineering

Delhi Technological University

Department of Computer Science & Engineering

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-110042, India

May, 2025

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-110042, India

CANDIDATE'S DECLARATION

I, **Ananya Nayak (2K23/CSE/04)**, hereby certify that the work being presented in the thesis entitled "**Measuring & Mitigating Gender Bias in Legal Contextualized Language Models"** in partial fulfilment of the requirements for the award of the Degree of Master of Technology, submitted in the Department of **Computer Science & Engineering**, Delhi Technological University is an authentic record of my own work carried out during the period from **Aug, 2023** to **May, 2025** from under the supervision of **Prof. Shailender Kumar**.

The matter presented in the thesis has not been submitted by me for the award of any other Degree of this or any other Institute.

Candidate's Signature

This is to certify that the student has incorporated all the corrections suggested by the examiners in the thesis and the statement made by the candidate is correct to the best of our knowledge.

Signature of Supervisor (s) **Signature of External Examiner**

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-110042, India

CERTIFICATE BY THE SUPERVISOR

I hereby certify that the Report titled "**Measuring & Mitigating Gender Bias in Legal Contextualized Language Models**" which is submitted by **Ananya Nayak**, Roll no. **2K23/CSE/04,** Department of Computer Science & Engineering, Delhi Technological University, Delhi in partial fulfilment of the requirements for the award of the degree of **Master of Technology**, is a genuine record of the work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any degree or diploma to this University or elsewhere.

Place: Delhi

Date: 30/5/2025

Prof. Shailender Kumar
Professor
Delhi Technological University

ACKNOWLEDGEMENT

I wish to express my sincerest gratitude to **Prof. Manoj Kumar**, our HOD at Delhi Technological University (Formerly Delhi College of Engineering), in New Delhi and the entire team of faculty in our department for their useful help, kind words and sincere encouragement during the project.

I am also grateful to **Prof. Shailender Kumar** for his continuous guidance and mentorship throughout the project. He showed me the path to achieving targets by explaining all the tasks to be done and explained to me the importance of this work as well as its industrial relevance. He was always ready to help me and clear doubts regarding any hurdles in this project work. Without his continual help and encouragement, this project would have failed.

I want to sincerely thank all my friends for helping me out.

Ananya Nayak

23/CSE/04

MEASURING AND MITIGATING GENDER BIAS IN LEGAL CONTEXTUALIZED LANGUAGE MODELS

ANANYA NAYAK

ABSTRACT

LLMs in NLP are now able to summarize, translate and produce text with remarkable success. Nonetheless, the way these models can be biased, including towards gender, has attracted some criticism. The issue appears because the training data is often influenced by cultural preconceptions that are reinforced during the model's improvement. Because of these biases, people's opinions can become set, society's values might shift and applications like automatic hiring, educational resources and customer help might not treat everyone the same. This research examines why gender bias exists in LLMs, the impact it has on NLP applications and how we can reduce it to make AI more equal.

Long-lasting inconsistency in society has resulted in gender bias being found in legal texts, court cases and age-old practices. This inequality can be found in the way some crimes or positions are more often linked to one gender and in how legal papers word description of roles and rules. As another example, if AI uses gender-based biases in the court, it could affect both sentencing and the way the AI models outcomes. It is very hard to solve this problem in legal language processing, mainly because legal texts are usually complex and depend on unclear hints. Legal information should be made fair while at the same time maintaining its accuracy through well-structured mitigation efforts.

TABLE OF CONTENTS

List Of Tables

List Of Figures

List of Abbreviations and Symbols

BERT	Bidirectional Encoder Representations from Transformers
GPT	Generative Pretrained Transformer
ELMo	Embeddings from Language Models
NLP	Natural Language Processing
CBOW	Continuous Bag of Words
GloVe	Global Vectors for Word Representation
LSTM	Long Short-Term Memory
NER	Named Entity Recognition
AI	Artificial Intelligence
MLM	Masked Language Modeling
LLM	Large Language Model
GAP	Gendered Ambiguous Pronoun
LCD	Legal Context Debias
μ	Mean
σ	Standard deviation

CHAPTER 1

INTRODUCTION

1.1 Overview

The basis of the meanings of words inside neural architectures is called word embeddings. They change words from text into vectors in several layers of semantic meanings. It was when GloVe [8] and Word2Vec [9] appeared that word embeddings gained significant importance by showing enhanced outcomes on many NLP problems. Right now, word embeddings are commonly used in detecting fake news and analyzing medical documents. However, a notable limitation of these methods lies in their static representation of words as vectors. Such static embeddings fail to capture polysemy, as they cannot account for the multiple meanings a word may have depending on its context.

1.1.1 Static Embeddings

Usually, traditional word embedding methods give the same vector for the same word, no matter how it is used in the text.

Characteristics:

- Fixed Representation: No matter how a word is used, each model has it represented by only one vector. Let's say a word "bank" will have the same embedding for all uses in phrases like "river bank" and "bank account". It has the exact same embedding though the context is different. This issue is crucial for tasks that depend on distinguishing between different meanings of words from context.

- Models: Commonly used models include:
  - Word2Vec: Google introduced Word2Vec which creates word embeddings with two main types of architecture. One is "Skip-Gram" which estimates which words appear around a central word. Another is "CBOW" which uses the surroundings of the word to guess its correct form. They depend on windowed contexts from the area and are optimized using strategies such as negative sampling.
  - GloVe: In order to build word embeddings, GloVe analyzes a matrix showing the shared occurrence of word pairs within a corpus. It puts together global statistics of words used and the benefits of learning context seen in Word2Vec.

- Advantages:
  - They are built using simple techniques, train quickly and don't need as much computer power as larger transformers.

  - You can easily use them in NLP pipelines, making them helpful in basic systems that don’t require much context.
  - They manage to work well on wide NLP problems where the details in the context are not too significant, including document classification and topic modeling.

- Limitations:
  - Embedding doesn’t change over time, so polysemous words still cause problems. Words that mean different things such as "bat" (animal) and "bat" (cricket equipment), are handled in the same way.
  - Such models pay no attention to how words interact through grammar or are connected in a sentence and its larger scope. Consequently, they are not the best fit for jobs that rely on full semantic understanding, like coreference resolution, analysis of complex sentiment and names identification.

1.1.2. Contextual Embeddings

Contextual embeddings generate dynamic word representations that depend on the context observed in the sentence where the word is present. Embeddings capture the rich semantic and syntactic properties of language, allowing one word to have many representations based on their usage in different sentences.

Characteristics:

- Dynamic Representation: A word's embedding changes depending on the surrounding words. For instance, "cell" in "cell phone" and "cell biology" will have distinct vector representations.

- Models: Examples of models generating contextual embeddings include:
  - ELMo (Embeddings from Language Models): Produces contextual embeddings using a bidirectional LSTM. Word representations are derived from all layers of the model, combining context from both directions.
  - BERT (Bidirectional Encoder Representations from Transformers): Generates embeddings by understanding context bidirectionally, using transformer-based architecture.
  - GPT (Generative Pre-trained Transformer): Creates contextual embeddings by processing text in a left-to-right direction, though not bidirectionally like BERT.

- Advantages:
    - The models allow them to tell apart various meanings of a word, unlike the single meanings represented by static embeddings.
    - Many NLP tasks that need deep understanding of meaning have been greatly improved by contextual embeddings. Some of them are Q&A, Named Entity Recognition (NER), Sentiment Analysis, Text Classification and Coreference Resolution.
    - Pretrained contextual models can be easily customized for various tasks using just a small amount of extra information which makes them useful for many areas.

- Limitations:
    - They use a lot of both memory and computing resources. Getting started with training BERT or GPT on your own means you need access to big data and powerful hardware.
    - While in deployment, using contextual models is slower and more costly than working with lightweight static embeddings, making it difficult for real-time or fast applications.
    - In most cases, language models require access to lots of data for different language tasks and types. This data can be difficult to find in small or specialized domains or when using low-resource languages.

Figure 1.1 Types of Word Embeddings

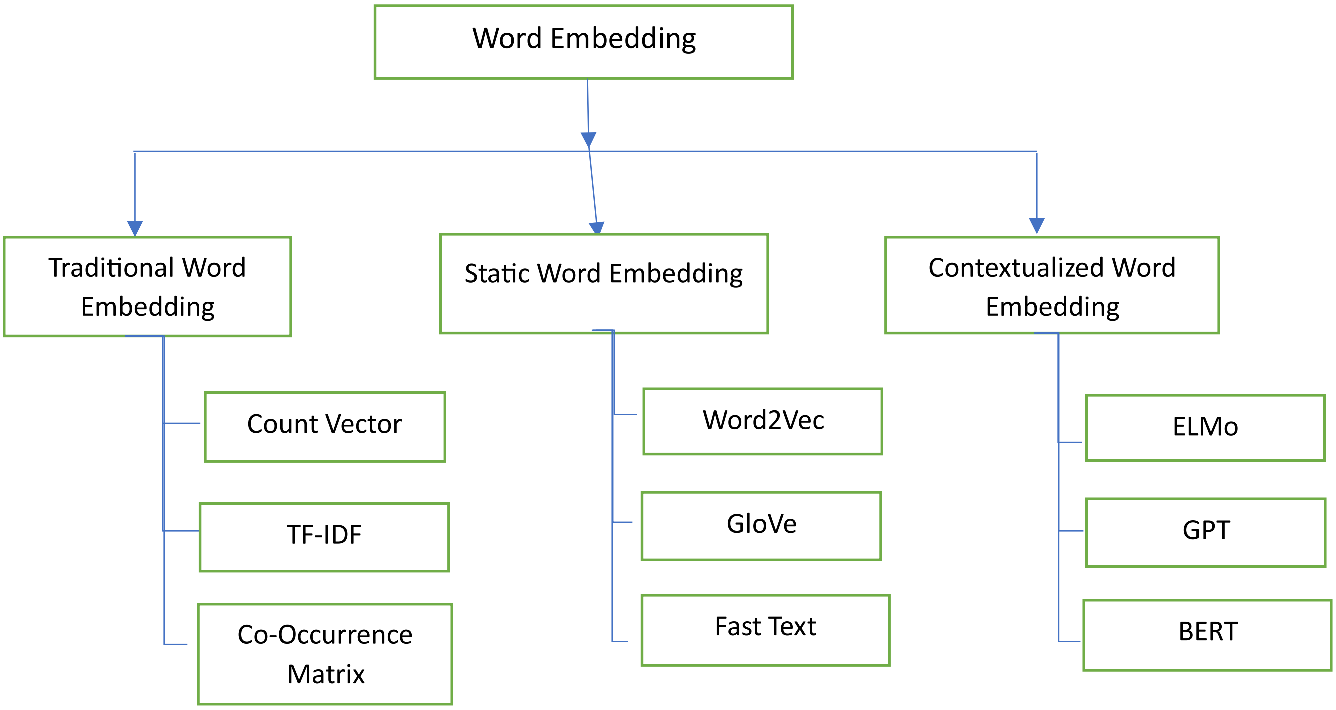


Table 1.1 Calculation of Contextual Embedding

Sentence	Contextual Embedding
I ate **Apple.**	0.3(I) + 0.1(ate) + 0.6(Apple)
I bought **Apple.**	0.3(I) + 0.2(bought) + 0.5(Apple)
I have **cold**.	0.1(I) + 0.3(have) + 0.6(cold)
He was very **cold** to me.	0.2(He) + 0.1(was) + 0.2(very) + 0.3(cold) + 0.1(to) + 0.1(me).

Above table showcases how we are calculating dynamic contextual embeddings like for word "Apple" based on the words before and after it in a sentence.

In transformer models, a target word is related to the other words in the input based on scaled dot-product attention. In particular, the target word's vector and each current word are scaled using the square root of the number of dimensions to ensure their dot product is not affected by the size of the numbers.

The score is used to decide how much attention the model pays to the current word in comparison to the target word. For this, the attention weight is multiplied with the vector for the current word. The sequence of process makes sure that the model compares all words and then combines the weights to generate a contextual representation for the target word. As a result, relevant knowledge from other words is added to the meaning of the word, making its context clearer.

It is officially called "Scaled Dot-Product Attention" in the paper et al. [10]. This element plays an important role in transformers, making sure that the model focuses on specific areas based on the context.

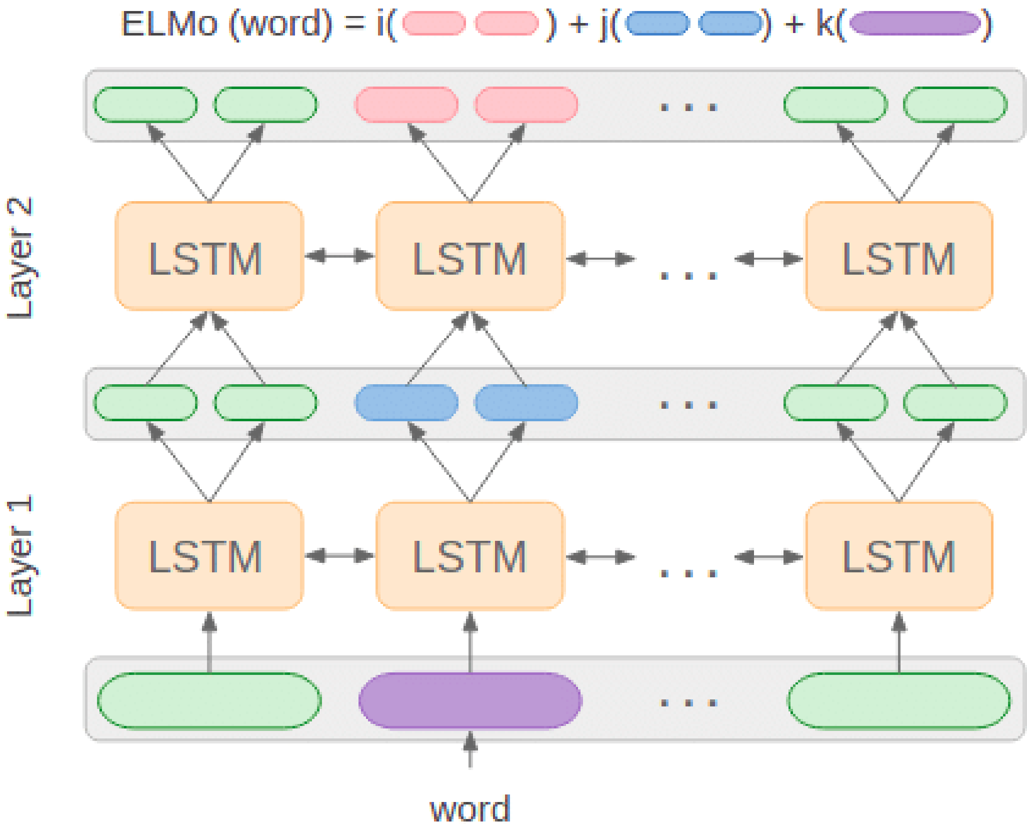
ELMo, GPT-2 and BERT are good examples of contextual language models, helping natural language processing by using attention mechanisms to address the issues present in static word embeddings. Before, every word had a fixed place in a sentence and these models consider context by giving different representations to each word as used in different sentences. With this approach, the models have access to meaning aspects and structure in a sentence, leading to better and clearer results that help with language tasks.

1.2 Contextualized Language Models

1.2.1 ELMo (Embeddings from Language Models)

- Created By: Allen Institute for AI in 2018.
- Unique Feature: Contextual word embeddings generated using deep bidirectional language models.
    - Unlike traditional embeddings like Word2Vec, which assign static vectors to words, ELMo [12] dynamically adjusts the word representations based on the context in which the word appears. For example, "cell" in "cell phone" and "cell biology" would have different embeddings.
- Architecture:
    - Built on a many layered (bi-directions seen) Long Short-Term Memory (BiLSTM) network.
    - The model captures rich contextual information by processing text sequentially in both forward and backward directions.
    - Word representations are generated by combining features from all layers of the network, which enhances their contextual relevance.
- Training:
    - ELMo is trained using a task of modelling, foreshadowing the coming word in a sequence in one direction and the previous word in another. This helps the model learn a comprehensive understanding of syntax and semantics.
- Applications:
    - Boosting performance in NLP tasks like question answering and text classification.
    - Enhancing Named Entity Recognition (NER) by providing richer contextual embeddings.
    - Improving coreference resolution and other language understanding tasks.

Fig 1.2 ELMo model BiLSTM architecture [14]



1.2.2 GPT

- Key Feature: Autoregressive language modeling.
  - Generates text by foreshadowing the next token in some sequence, trained L-R.
  - Pre-training is done on this massive corpus which has internet text and is further improved for specific tasks.
- Advantages Over BERT:
  - Focuses on generation capabilities, making it more suited for tasks like text completion, summarization, and dialogue systems.
- Architecture:
  - Transformer-based with a unidirectional approach.
  - Variants like GPT-3/4 use extensive pre-training data, making them capable of zero-shot and few-shot learning.
- Applications:
  - Conversational AI.
  - Creative content generation (e.g., stories, articles).
  - Code generation (e.g., Codex for programming tasks).

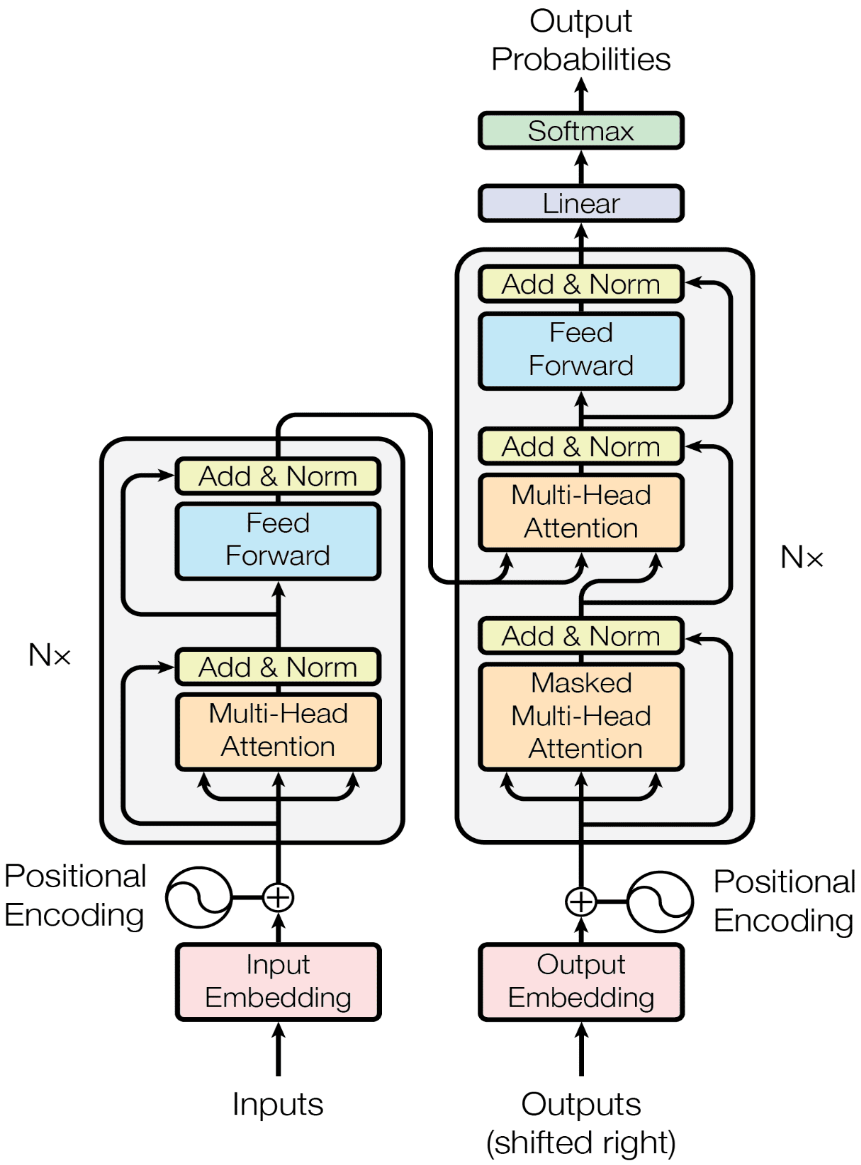
1.2.3 BERT

- Key Feature: Bidirectional attention mechanism.
  - Unlike previous models like Word2Vec [9] or GloVe [8] which give us one vector embedding for every word, BERT [10] takes into account the context in a sentence based on some word. Like, "bank" in "river bank" vs. "bank account" gets unalike embeddings.
- Architecture:
  - Transformer-based, specifically designed to read text bidirectionally (both ways simultaneously).
  - Trained with MLM, where words in the input are covered (masked), and it foreshadows these masked words based on context.
  - Also trained with next sentence prediction (NSP) to capture sentence-level relationships.
- Applications:
  - Question answering (e.g., SQuAD benchmarks).
  - NER.
  - Text classification and sentiment analysis.

1.2.4 LegalBERT

- What it is: A domain-specific adaptation of BERT tailored for legal texts.
- Purpose: General models like BERT are trained on diverse datasets that may lack legal-specific nuances. LegalBERT [17] is fine-tuned on legal documents like case law, statutes, contracts, etc., to handle legal terminology and structure.
- Key Features:
  - Pre-training is done on some corpus of legal texts to specialize in understanding legal language, which is often formal, complex, and jargon-heavy.
  - Captures nuances like legal definitions, procedural terms, and citation relationships.
- Use Cases in Legal NLP:
  - Contract analysis: Identifying clauses, obligations, and risks.
  - Case law research: Extracting relevant precedents and arguments.
  - Legal document classification: Categorizing types of legal texts (e.g., judgments, filings).
  - Information retrieval: Enhancing legal search engines by understanding legal-specific queries.
- Example Projects:
  - Tools for automated contract review.
  - Legal chatbots for basic query resolution.

Fig 1.3 Transformer architecture for GPT and BERT [13]



In summary, BERT and GPT are general-purpose models excelling at contextual understanding and generative tasks, respectively. Because LegalBERT has been trained on legal texts, it solves the difficult aspects of understanding legal language and applies well in the legal field.

1.3 Motivation

The biased responses in large language models are due to the nature of the datasets which includes old societal stereotypes & unequal data. Frequently, training data from different resources links genders to specific jobs or traits and this is reflected by the models. Sometimes, LLMs use masculine pronouns for leadership jobs and feminine pronouns for caregiving professions which only helps to confirm stereotypes. Apart from open misinterpretations, biases also include indirect relationships such as seeing ambition in men and empathy in women.

Due to these biases, we see problems in hiring and in the way chatbots interact with people. Fighting against gender bias gets more complicated when we keep in mind that language is complex and can change from situation to situation and in some situations, correcting too much may go too far and result in missing needed meaning.

To mitigate problems, methods include representing data fairly, using improved debiasing techniques and increasing oversight by ethics experts. On the other hand, there are still problems, including how to show non-binary identities and handle the demanding tasks involved in changing models. Because LLMs are at the heart of AI applications, ensuring they are fair, inclusive and trusted means fighting gender bias in different fields.

Using NLP, law practitioners can sort cases automatically, review large amounts of legal information and reach data faster. It uses data analysis to foresee case outcomes, find important entities and guarantee contract compliance. By making legal texts easier to read, NLP helps more people participate in choosing what's best for them. Creating NLP tools for legal work means paying attention to biases in law and finding methods to solve them.

Even though this is an oversimplification, this model can make assumptions about women and men based on how their data are trained. As a result, predictions and advice may reflect biases which can oppose fairness and strengthen stereotypes.

CHAPTER 2

LITERATURE REVIEW

We provide an overview of one of the first and most significant papers on neural language models which examines how bias amplification occurs in structured prediction systems like visual semantic role labeling. The researchers find that, despite being trained on fair examples, models often connect common gender-related tasks differently when applying their training at inference, for instance, by matching cooking to women and driving to men.

As a solution, Zhao et al. [2] build a technique that maintains gender equalization when the model is trained. They do not change the raw numbers or predictions; instead, they steer the entire collection of gendered results to even out amplification without harming the final task score.

The results of this study matter greatly for legal NLP since they demonstrate that the machine learning approach can actually enhance any biases found in legal information. Therefore, if the model was not designed to be biased, it can still learn unfair patterns or conclusions from previous cases and end up delivering biased answers.

In addition, the research explores a significant and useful idea: using distributional data to help reduce bias. It suggests that by studying the use of words, it can be possible to develop techniques to address gender bias. It helps people address fairness in legal NLP systems and also introduces useful tools for detecting and reducing bias during the process of data representation. As a result, it forms the base for further efforts to design legal language models that promote fairness and responsibility.

In this study et al. [3], we examine how word embeddings picked up societal biases associated with gender, race and age. Using the Word Embedding Association Test (WEAT), based on the earlier Implicit Association Test (IAT), the authors prove that standard Word2Vec and GloVe word strategies replicate common stereotypes about associations like men in careers and women with family.

The study shows that machine learning models that work with unfiltered text tend to learn and reflect society’s existing biases, without there being any annotation process. The chief point made by the authors is that the bias is built into the language, not due to how algorithms function.

It proves that in legal NLP, data structure and characteristics play a key role in introducing bias, in addition to training and architecture of models. Specially, it proves that the arrangement of words, including how many times they appear, what surrounds them and what terms frequently accompany them, can carry and spread biased ideas. So, even models not explicitly taught to be biased can end up mirroring or strengthening such unfair connections.

The finding proves that domain-fit ways to reduce bias are required when dealing with law. Common techniques for removing gender bias may fail to notice some unique words used in legal contexts which are much different from regular speech. Due to this unique bias situation, the Legal-Context-Debias method presented in the thesis considers the language used in laws and the certain biases that may be present in how courts talk about such issues. So, it makes it obvious why debiasing is needed and also targets solutions that are suitable for the legal domain.

The research here et al. [1] examines and addresses gender imbalance in LegalBERT which is a transformer model trained for the law. According to the authors, even domain-related models such as LegalBERT, exposed to neutral legal texts, might include and distribute gender stereotypes because of the unequal and hidden aspects in the materials it was trained on.

Two specially constructed corpora are introduced by the authors to systematically assess this bias.

- BEC-Cri (Bias Evaluation Corpus for Crimes): Paired sentences that describe different crimes with actors from either gender.
- BEC-Pro is a comparable evaluation set created using professions, following in the footsteps of WEAT and others.

According to the authors, balancing the data with a gender classification objective using LegalBERT helps fine-tune the system. As a result, the model takes into account gender-related factors and is less likely to form erroneous links. Researchers compare LCD against two approaches meant for general functions: GPD and GAP fine-tuning. While keeping the same task accuracy, LCD performs much better in terms of gender portrayal than both. It points out that debiasing methods need to be relevant to the domain and specific to each context in legal NLP.

In this paper et al. [5], authors build on earlier research to provide a clearer picture of gender bias in large-scale legal collections and discuss solutions to debias them successfully. The paper says that gender bias often goes unspoken and quietly appears in the language used by lawyers across different legal situations.

They conclude that even when there are few gendered terms, legal documents can reinforce stereotypes using groupings of words, employed professions or the descriptions of female versus male cases. More importantly, the authors find that removing clear gender tags is not enough, since models usually pick up bias from background factors, structures or language patterns.

Methods they try for debiasing include the use of counterfactual data augmentation, training on balanced datasets and interventions on the representation level. The evaluation indicates that debiasing within the legal domain, along with context, leads to the greatest improvements, confirming once again that custom strategies are needed.

The various studies before this paper et al. [4] examine problems in legal language models, but this one aims to understand issues in decisions made by judges. The authors look into the likelihood of judges in Kenya granting ethnic bias in their ruling on appeals. Using lots of information about decisions from appellate courts, they look at how the outcomes are affected by whether both the judge and defendant belong to the same ethnic group.

It has been found by studies that share an ethnic background increases the likelihood that a judge will favor someone who appears in court. This gap keeps occurring despite considering both legal differences and the medical facts of each trial which shows that it is not only due to laws or clear case circumstances. According to the results, people's judgments in court can be affected by hidden biases outside of official legal thought processes.

This realization matters a lot in legal AI since it points out that the issue with bias in human processes existed even before computers became involved. Should AI models use data that contains human bias, they could very well learn and imitate those tendencies and be capable of doing it in vast amounts. It causes significant concerns about both justice and accountability in systems that use artificial intelligence in law.

Hence, efforts to combat bias should take place in every part of AI creation such as when the data is collected, the model is trained and the system is used. If AI tools are not introduced entirely and appropriately, instead of helping, they could actually increase the existing inequalities in law.

The collected studies find that biases in both language models and law systems are both widespread and complex. According to Caliskan et al. [3] and Zhao et al. [2], bias exists throughout language data and can be exacerbated even in seemingly standard model training. Therefore, developers should take care to record and address biases during all stages of creating an NLP solution.

The research groups led by Bozdag et al. [1] and Sevim et al. [5] confirmed that, in complex domains, simple debiasing approaches are usually not enough. Rather, approaches designed for different fields, including Legal-Context-Debias (LCD), tend to be more effective and do not change the meaning of important legal insights. They bring about customized benchmarks (such as BEC-Cri and BEC-Pro) that support specific forms of evaluation.

At the same time, Choi et al. [4] reveal actual cases of ethnic discrimination in judicial decisions, proving once again that such systems should be built with a strong grasp of society. All of this research gives a solid basis for the thesis, supporting the position that domain specificity, responds to different contexts and retains links to both technology and practical solutions is important for bias reduction in legal NLP.

Previous research has formed the basis for deciding on the current study's direction and intentions. Previously, researchers have pointed out the fact that legal language models frequently exhibit bias and it is hard to address this bias. Based on the earlier research, this study makes use of the ECtHR dataset and a set of selected Indian legal texts to measure the effects of different biases in both systems. The addition of this data makes the study stronger by featuring many legal systems and language use found in India and increasing the usefulness of its conclusions.

The study hopes to sharpen the debate and support it with real data by reviewing the performance of various debiasing strategies among the sets of data. It is shown that approaches that take into account the nature of legal language are more effective and are necessary for ethical reasons. These solutions need to use technology for engineering and put fairness, transparency and accountability into practice to ensure that AI in the legal area is both ethical and accurate.

All in all, the study makes clear that attaining justice in legal AI needs more than just one-size-fits-all methods. Given this, we need to use practices that pay attention to both how laws operate and the role law plays in society.

CHAPTER 3

METHODOLOGY

3.1 Bias Measuring

Researchers have obtained excellent results in NLP tasks, thanks in part to BERT and similar models. But these models often end up repeating social biases, for example, gender bias, that exists in the data used to teach them. To fix this issue in the law field, the paper et al. [1] discusses a new way of measuring gender bias in tools used for legal language processing.

The authors present BEC-Cri, a new collection of data used to assess bias in the legal world. All of the template sentences in this dataset come from crime-related terms available in the FBI database. The bias measuring methodology exploits the Masked Language Modeling (MLM) capability of BERT-like models. Sentences are masked at key positions (e.g., gendered nouns), and the model's probability predictions are compared in masked and unmasked contexts to derive association scores. These scores quantify the model's gender inclinations, forming the core of their proposed bias evaluation metric.

To validate their domain-specific method, the authors also apply an alternative domain-general bias evaluation technique based on the BEC-Pro dataset, which uses profession-related sentences inspired by the WEAT framework.

Methodology

1. Template Creation:
   - Sentences with explicit gender terms are constructed, such as:
     - "He is a lawyer"
     - "She is a lawyer"
   - These sentences are then converted into masked templates, like:
     - "[MASK] is a lawyer"
   - The model predicts probable replacements for [MASK], such as "he," "she," or other terms.

2. Comparison of Probabilities:
   - Any unevenness in the probability scores of gendered terms (such as "he" and "she") is found using this method.

  - Large differences in how likely people are to be accused can point to a bias. Like predicting "he" over "she" for [MASK] in "[MASK] is a lawyer" reveals a bias that will most likely link law to men.

3. Datasets Used for Testing:
   - BEC-Cri (Bias Evaluation Corpus for Crime): The corpus explores types of bias that appear when discussing crimes or criminal matters.
   - BEC-Pro (Bias Evaluation Corpus for Professions) is this dataset. considers bias in neutral or professional areas such as roles or jobs people usually have.

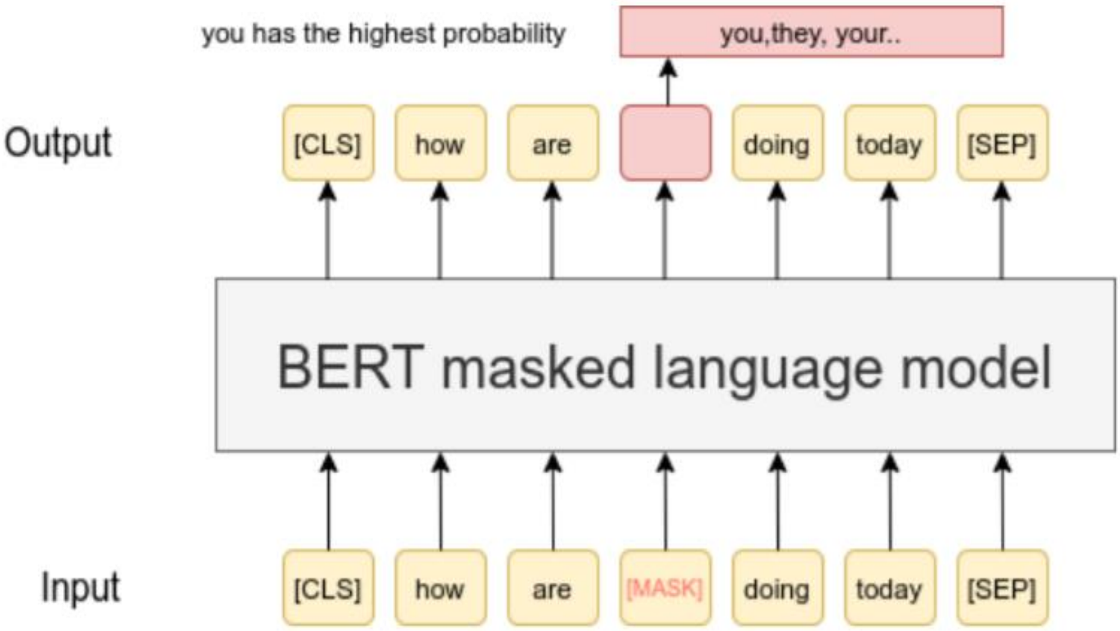
Analysis Steps

1. Association Scores:
   - Statistical methods are used to connect gendered terms with specific fields and kinds of crime in the model.
   - For example, probabilities are calculated for guessing "he" versus "she" in sentences such as "The teacher is a man" or "The thief was a woman."
   - By exploring scores, we can recognize stereotypes that associate men with many job or criminal roles more often than women.

2. Context Sensitivity:
   - Experts analyse the accuracy of these predictions in crime and everyday contexts.
   - For example:
     - A bias is found when we predict that "she" was mentioned in the sentence "She is a teacher."
     - But in "[MASK] was found guilty of fraud," "he" might have a much higher probability. This tells us that there is a bias linking men with criminal behavior.

3. Dataset-Based Comparisons:
   - The evaluations conducted on BEC-Cri and BEC-Pro are compared to check for both minor and major biases.
   - Discrepancies between predictions in crime-related versus professional contexts highlight how situational factors influence bias.

Fig 3.1 Bert Masked Language Model [13]



To be trained, BERT relies on the Masked Language Modeling (MLM) objective, making it a powerful language model in advance. MLM helps BERT read text from left to right and right to left by having it guess the missing words inside a single sentence.

The way MLM works is outlined below.

- During its training, BERT randomly hides a specific percentage (normally 15%) of the input's tokens.
- Its challenge is to figure out the original, masked values relying on the words that surround them.
- BERT is able to use all the information from the sentence at once because it pays attention to tokens in both directions.
- This means that this model is different from old ones which only looked at past words to decide the next.

Table 3.1 Illustrations from the BEC-Cri. Attribute terms are highlighted while target words are hidden. [1]

Sample Sentence	**Possible Target Words**
[MASK] has committed forgery.	'he', 'she'
Is [MASK] the one behind this fraud?	'father', 'mother'
The [MASK] filed charges against murder.	'man', 'woman'

The Bias Measuring Algorithm proposed et. al. [1]

```
Input → A set of sentences containing target word T and attribute word A
Output → Mean association score
Initialize u = 0
for S(T, A) E D do
    pt ← P([MASK_TARGET]) = G|S([MASK_TARGET], A)|
    pprior ← P([MASK_TARGET] = G|S([MASK_TARGET], [MASK_ATTRIBUTE]))
    AssociationScore(S(A, T)) ← log(pt/pprior)
    u ← u + AssociationScore(S(A, T))
end for
u ←(1/[D])u
```

3.2 Debiasing in Contextualized Language Models

To mitigate gender bias in legal contextualized models, the paper proposes a domain-specific fine-tuning method called Legal-Context-Debias (LCD). The method fine-tunes LegalBERT-Small on a modified version of the ECtHR corpus—a collection of European Court of Human Rights cases—carefully balanced to include equal numbers of male and female applicants. The fine-tuning task is designed as a binary gender classification, leveraging subtle contextual clues in legal text to help the model better understand and neutralize its gender associations.

For comparison, two baseline debiasing methods were also implemented by the authors et al. [1]:

- Gender Preserving Debiasing (GPD) – This method uses a similarity-based loss to reduce gender associations in word embeddings while preserving some gender information.
- GAP Fine-Tuning – Based on the Gendered Ambiguous Pronouns (GAP) dataset from Webster et al., this method fine-tunes the model on gender-neutral coreference tasks. Counterfactual data substitution is applied to further reduce gender skew.

3.3 Experimental Setup and Evaluation

LegalBERT-Small is the framework used in this study for the experiments on finding and lessening bias in the legal domain. Its balance of performance and how light it is on processing resources makes LegalBERT-Small a good model to start with for any legal fine-tuning.

In the beginning, the BEC-Cri dataset is used to learn about present-day gender inequality. With this dataset, one can examine the model's bias before taking any debiasing steps, giving a point of reference to judge improvement against.

Afterward, the study uses two complementary sets of legal documents: rulings from the European Court of Human Rights and another set of Indian cases that have been manually made by experts. The datasets were picked so that crossjurisdictional debiasing strategies can be tried and tested, allowing the approach to be used in different legal systems and languages.

3.3.1. Technologies Used

- The main factor behind using Python in the development was that it is simple, easy to read and many researchers in the field use it. Because of its extensive and helpful library system and dedicated community, developers found it perfect for development.

- Hugging Face Transformers was employed as the main framework to use pre-trained models such as Legal-BERT. It offered smooth tools for handling tokenization, handling models, fine-tuning and evaluating them. It made it much easier to add specific models to our biases process.

- PyTorch played an important role in designing, training and checking neural models, thanks to its strong flexibility. Since the system uses a flexible graph and modular structure, it was easy to introduce new tuning strategies and keep an eye on how the model improved during the debiasing process.

- We used Pandas a lot to handle data manipulation, cleaning and preprocessing. Thanks to it, structured data could be loaded, important features could be picked and outputs were made ready for evaluation and reviewing. It was especially helpful to group model predictions as DataFrames and use them for measuring and understanding bias.

- Generation of graphs to show the results was done with Matplotlib. Histograms, box plots and similar plots were developed to check for possible patterns in the data and to observe how the model behaved before and after debiasing.

- NumPy was an important tool, helping with fast calculations on all the arrays used in the project. It took part in the computations achieved during preprocessing, statistical analysis and calculation of metrics for all the datasets.

3.3.2. Dataset Preparation

Three primary datasets were utilized in this study for bias evaluation and mitigation:

- **BEC-Cri:** BEC-Cri is designed as an evaluation corpus to check whether language related to crime contains gender bias. The terms were taken from the FBI database, so they are based on common patterns found in writing about criminal justice. The method used in this dataset is called masked language modeling (MLM). Here, chosen words or tokens are covered and the model needs to guess what they are. In this way, researchers can work out exactly the level of gender discrimination the model displays. Essentially, the inclusion of the BEC-Cri set made it possible to use an objective standard that is tied to the use of language in the medical field.s

- **ECtHR:** The dataset is constructed from the European Court of Human Rights corpus which gathers many important court case information. For this analysis, a special group of ECtHR cases was put together, with each applied group made up of the same numbers of male and female applicants. Because of this balancing, the model learns about gender equality in training and will be less likely to support unfair gender biases. There are 3,032 records in the dataset and in the supervised fine-tuning setup, the model is taught to do a binary gender classification. Using this strategy helps reduce unfair treatment between genders and at the same time ensures accurate understanding of legal domain vocabulary. Just as BEC-Cri was, the ECtHR dataset was part of this project and gave a solid base for building and testing our models.

- **Indian-Legal-Dataset:** To increase domain diversity and test generalizability, an Indian legal corpus was manually curated. Instead of relying on heuristics or automatic inference (e.g., using name patterns or pronoun frequency to predict gender), which are prone to noise and mislabeling, a manual approach was adopted. This ensured robustness and accuracy in gender labeling:
  - Automatic scripts could have scanned summaries and inferred gender, but such techniques risk injecting subtle biases or errors.
  - Manual curation resulted in a cleaner and more reliable dataset, albeit time-intensive.

  We made sure while making this dataset that there were proper gender clues for a good classification task. This was to avoid gender ambiguity.

  We made use of sources like “indiankanoon.org” [15] to get facts for various Indian legal cases and make our own dataset for binary classification. A total of 500 records with 250 male and 250 female applicant cases was curated for a gender-balanced legal corpus. This would be used like the ECtHR dataset for legal-context-debias (LCD). It has 2 attributes: text (a summary of the facts of the legal case) and applicant gender.

Table 3.2: Sample data from our Indian-Legal-Dataset

text	applicant-gender
The **applicant** was tried for an offence under section 302 Indian Penal Code for the murder of **his** wife. The evidence consisted mainly of the uncorroborated dying declaration of the wife. The Sessions judge accepted the evidence but convicted the applicant under section 304 Part 1 Indian Penal Code.	1
The said FIR dated 8th December, 2019 had been lodged by the **appellant** herein between 23:00 hrs and 23:30 hrs in the night stating that earlier on that day, at about 16:00 hrs, **his** father, aged about 55 years, was attacked by the respondent- accused, at the *Lalpura Pachar* bus stand, with the intention of killing **him**. That the respondent-accused pinned the deceased to the ground, sat on his chest and forcefully strangled **him**, thereby causing **his** death.	1

Each record in the Indian dataset contains a legal case summary with the corresponding applicant gender (1 for male, 2 for female), which is processed and tokenized during model training.

3.3.3 Preprocessing of data

You have to ensure that the text used with the model is properly formatted before starting with any task. LegalBERT and similar models need the input data to be organized in an efficient way for tokenization, encoding and learning tasks.

The preprocessing steps typically convert original legal documents full of complex structure, jargon, citations and unique terms into a format that the model can use.

- Tokens and padding are used in texts. Before anything else, each sentence goes through the model's tokenizer. This takes each part of the text apart and matches those parts to unique IDs from the model's vocabulary. Because BERT works best with even-length input, shorter sequences are extended using special tokens which are usually zeroes. It might be represented as [102, 123, 234, 789, 213] when separated by its tokens [CLS] He commits murder [SEP]

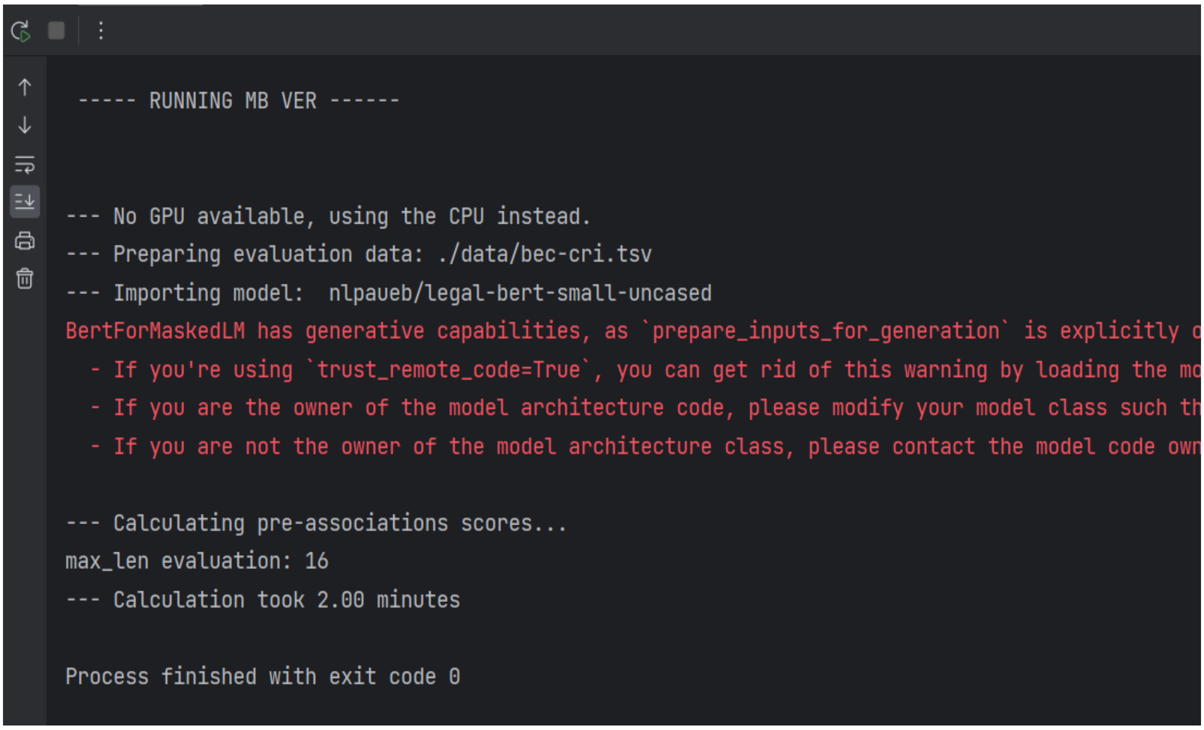
- Following padding, we set up an attention mask letting the model recognize the non-padding tokens. In case padding is added with zeroes in our example, the attention mask would look like [1, 1, 1, 1, 1, 0, 0, 0] with ones representing actual tokens and zeros showing paddings. As a result, the model works on only the significant aspects during training and when inference takes place.

- The tools necessary for image processing such as token IDs and attention masks, are brought together in a TensorDataset.

- The batches are then given to the LegalBERT model. A model provides logits that represent the unnormalized confidence for the model's guess at each masked position.

- Using the softmax function is required to turn logits into easily understandable probabilities. They show how actively the model links specific terms like gendered pronouns to different contexts, for example, crimes. Analyzing these results helps us find out if the model has any gender biases.

Fig 3.2: Screenshot of terminal showing calculation of pre-association scores using Bec-Cri dataset



```
 ----- RUNNING MB VER ------



--- No GPU available, using the CPU instead.
--- Preparing evaluation data: ./data/bec-cri.tsv
--- Importing model:  nlpaueb/legal-bert-small-uncased
BertForMaskedLM has generative capabilities, as `prepare_inputs_for_generation` is explicitly d
  - If you're using `trust_remote_code=True`, you can get rid of this warning by loading the mo
  - If you are the owner of the model architecture code, please modify your model class such th
  - If you are not the owner of the model architecture class, please contact the model code own

--- Calculating pre-associations scores...
max_len evaluation: 16
--- Calculation took 2.00 minutes

Process finished with exit code 0
```

3.3.4 Fine-tuning configuration & significance

LegalBERT-Small is adjusted for a binary gender classification task using a pre-existing dataset called ECtHR [1] and our own manually curated Indian-Legal-Dataset. This type of work helps to spot and, hopefully, reduce the effects of gender bias in the law. This process works better with the AdamW optimizer which notably reduces overfitting for small datasets that we are using. AdamW adjusts weight decay independently from the other optimizer steps which helps the model reach good generalization without hampering optimization.

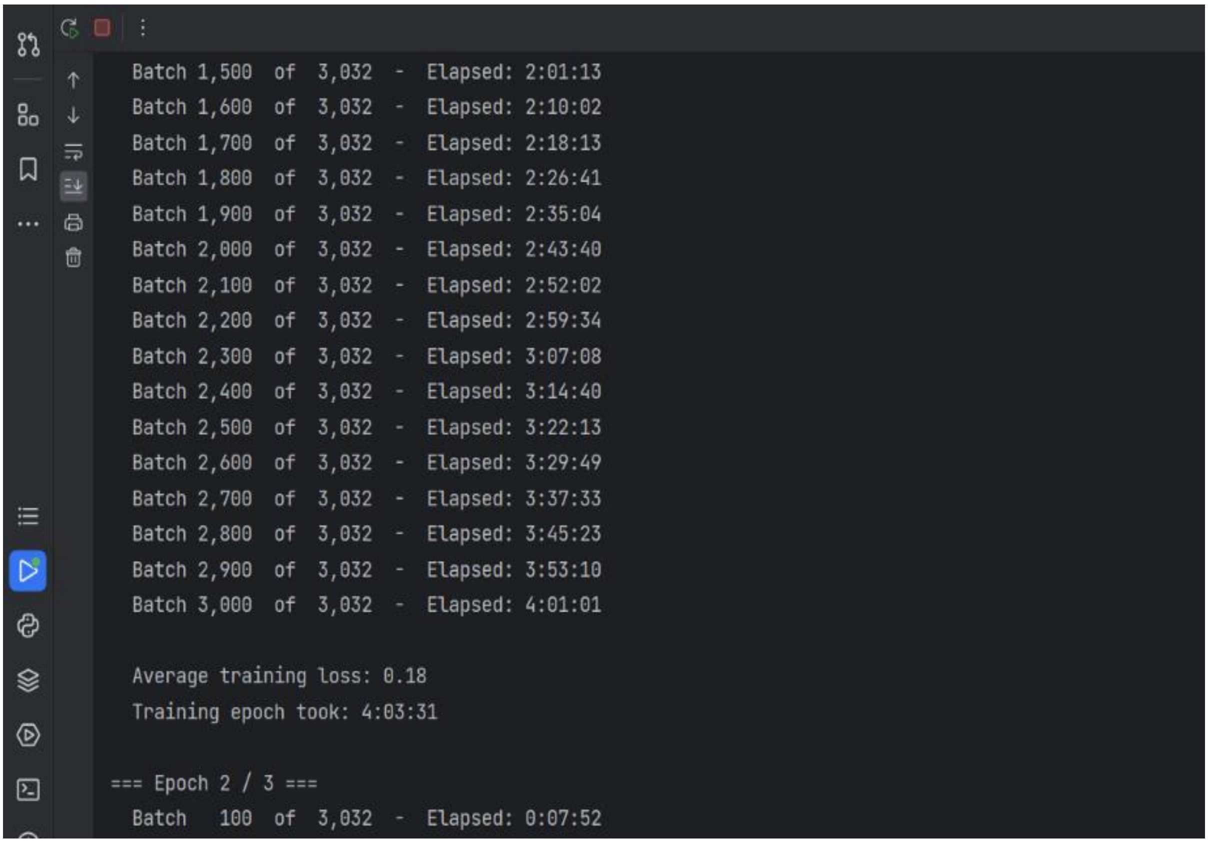
Best practice in BERT fine-tuning recommends a learning rate of 1e-5. As a result, each update to the parameters is small and carefully guided which supports using the model's previous legal knowledge and allowing it to learn the new classification well. Using this conservative approach is key when trying to fine-tune on data that is sensitive or not diverse, since it protects against massive forgetting.

When training the model, 3 epochs are used, each with a batch size of 1 and it is able to process sequences of up to 512 tokens. A linear learning rate scheduler is chosen without adding a warmup phase, because it helps keep early learning conservative. As a result, it helps avoid too fast changes to the pretrained model.

This technique allows LegalBERT-Small to process gender-related terms in law documents without losing its knowledge of specific law terminology. As a result, the model can reduce bias in its output without affecting its performance on legal language tasks.

Adaptation methods here were guided by the steps outlined in [16] which gave us a reliable approach to training transformer models for downstream classification.

Fig 3.3: Screenshot of terminal showing epoch runs while fine-tuning Legal-BERT model on ECtHR dataset

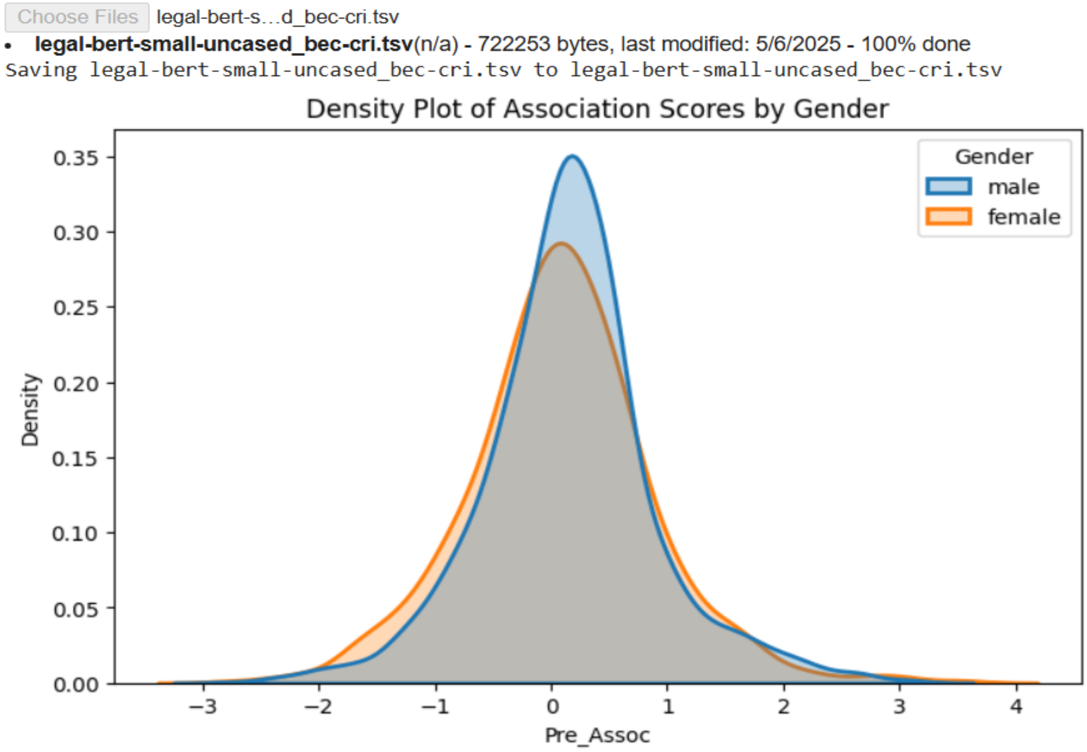


```
  Batch 1,500  of  3,032  -  Elapsed: 2:01:13
  Batch 1,600  of  3,032  -  Elapsed: 2:10:02
  Batch 1,700  of  3,032  -  Elapsed: 2:18:13
  Batch 1,800  of  3,032  -  Elapsed: 2:26:41
  Batch 1,900  of  3,032  -  Elapsed: 2:35:04
  Batch 2,000  of  3,032  -  Elapsed: 2:43:40
  Batch 2,100  of  3,032  -  Elapsed: 2:52:02
  Batch 2,200  of  3,032  -  Elapsed: 2:59:34
  Batch 2,300  of  3,032  -  Elapsed: 3:07:08
  Batch 2,400  of  3,032  -  Elapsed: 3:14:40
  Batch 2,500  of  3,032  -  Elapsed: 3:22:13
  Batch 2,600  of  3,032  -  Elapsed: 3:29:49
  Batch 2,700  of  3,032  -  Elapsed: 3:37:33
  Batch 2,800  of  3,032  -  Elapsed: 3:45:23
  Batch 2,900  of  3,032  -  Elapsed: 3:53:10
  Batch 3,000  of  3,032  -  Elapsed: 4:01:01

  Average training loss: 0.18
  Training epoch took: 4:03:31

=== Epoch 2 / 3 ===
  Batch   100  of  3,032  -  Elapsed: 0:07:52
```

CHAPTER 4

RESULTS & DISCUSSION

Fig 4.1: Density plot of pre-association scores by gender



```
Choose Files legal-bert-s…d_bec-cri.tsv
• legal-bert-small-uncased_bec-cri.tsv(n/a) - 722253 bytes, last modified: 5/6/2025 - 100% done
Saving legal-bert-small-uncased_bec-cri.tsv to legal-bert-small-uncased_bec-cri.tsv
```

The density plot of association scores by gender before debiasing reveals subtle but consistent disparities in how the model associates gender with masked tokens. The x-axis (Pre_Assoc) represents the log-likelihood ratio of gendered token predictions in context, while the y-axis (Density) reflects the probability distribution of these association scores.

- Shape and Centering: Both male (blue) and female (orange) distributions are roughly symmetric and centered around zero, suggesting no extreme bias. However, their modes (peaks) are slightly offset, indicating skewed associations in the pre-trained model.

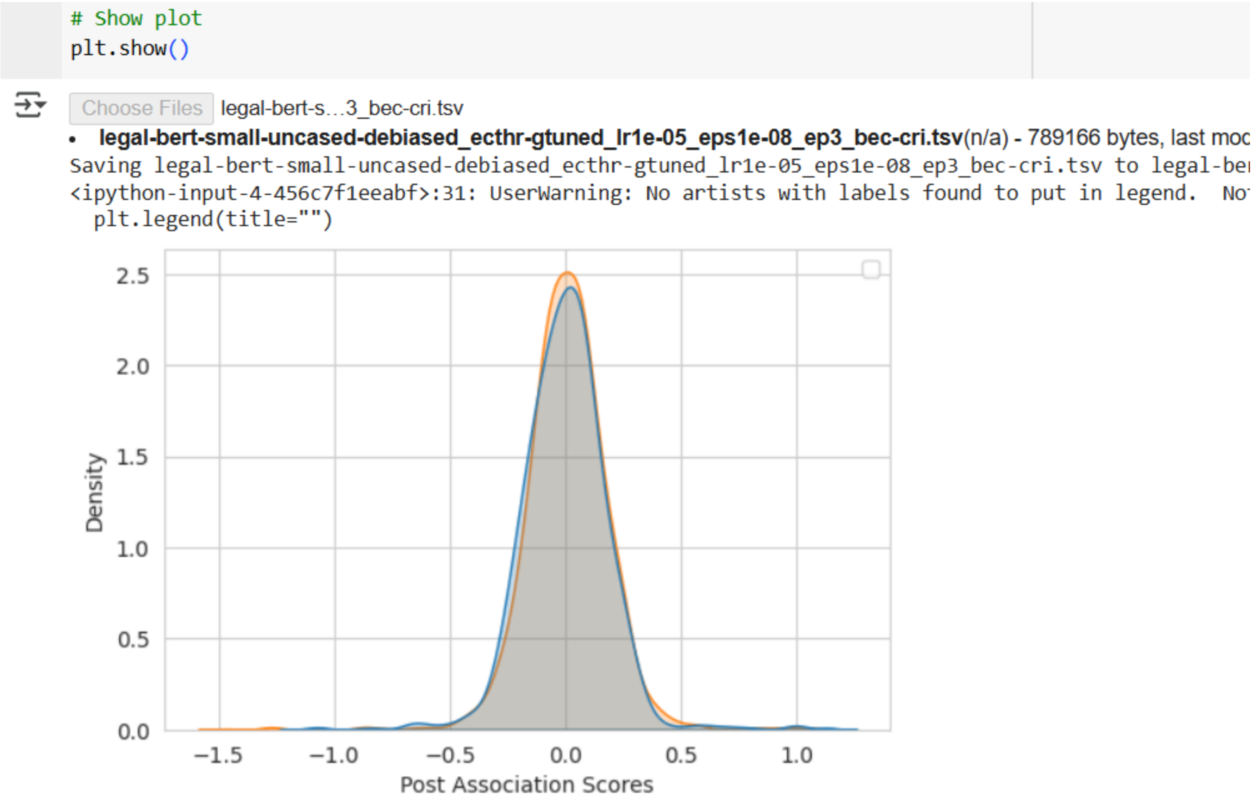
- Distribution Spread: The male curve is slightly taller and narrower, while the female curve is more spread out. This implies that the model's predictions for male-associated contexts are more consistent and confident, whereas its associations with female contexts are more variable and uncertain.

- Intersection and Asymmetry: The curves diverge most noticeably in the positive association range (~0.5 to 2.5), where the male curve dominates. This suggests a systematic preference for male terms in certain masked contexts, potentially reflecting underlying gender bias in the pretraining corpus.

- Overlap: Despite the differences, there is substantial overlap between the two distributions. This indicates that the bias is not overt, but rather subtle and statistical, reinforcing the importance of quantitative evaluation instead of relying on anecdotal evidence.

These pre-debiasing results show that while the model does not exhibit extreme gender bias, it still encodes systematic asymmetries in association scores. Such disparities, if left uncorrected, may influence downstream decisions in sensitive domains like law. This underscores the need for domain-specific debiasing methods to ensure fair and neutral predictions.

Fig 4.2: Density plot of post-association scores by gender after fine-tuning model on ECtHR dataset



```
# Show plot
plt.show()
```

- **legal-bert-small-uncased-debiased_ecthr-gtuned_lr1e-05_eps1e-08_ep3_bec-cri.tsv**(n/a) - 789166 bytes, last moc

```
Saving legal-bert-small-uncased-debiased_ecthr-gtuned_lr1e-05_eps1e-08_ep3_bec-cri.tsv to legal-bei
<ipython-input-4-456c7f1eeabf>:31: UserWarning: No artists with labels found to put in legend.  No
  plt.legend(title="")
```

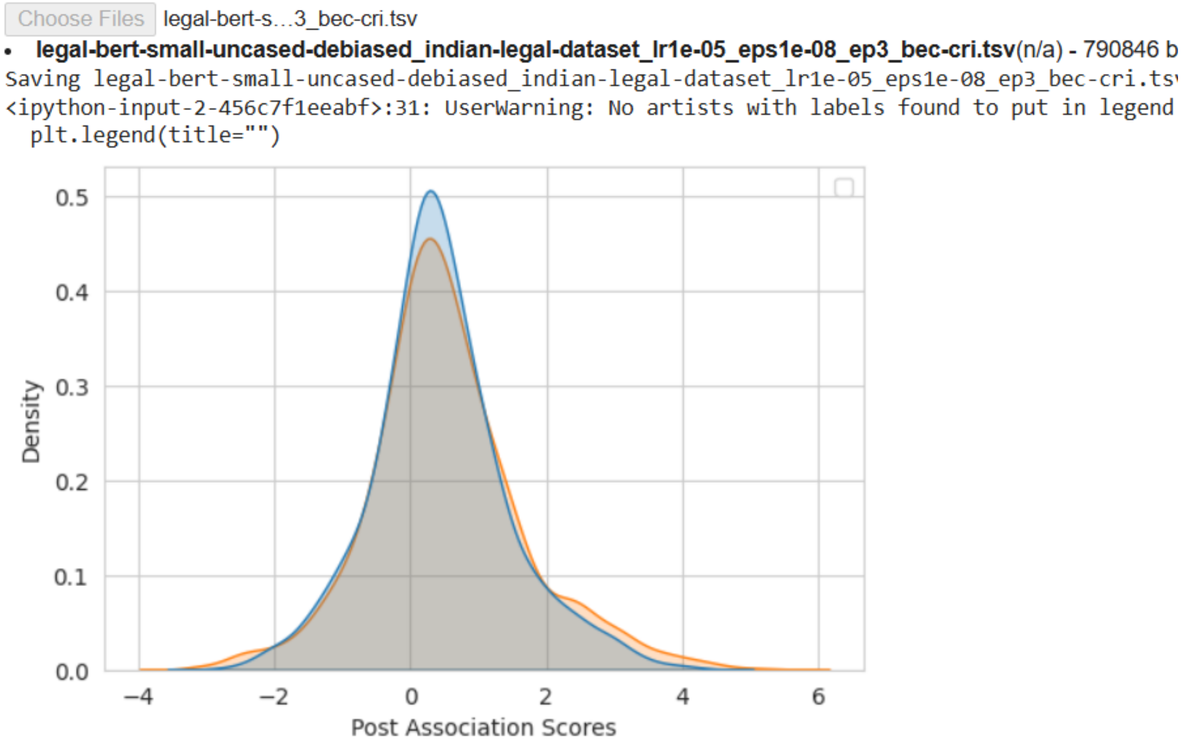
- Centered Distribution: The post-association scores are tightly centered around zero, which indicates that after debiasing, the model no longer strongly associates certain concepts (e.g., gendered words with professions or legal roles). This central alignment is a key sign of neutralization of biased associations.

- Sharp Peak & Symmetry: The density curve is both narrow and symmetric, suggesting that the majority of association scores are close to zero and evenly distributed. This reflects low variance, indicating that the model treats different groups or concepts similarly—an important goal in debiasing.

- Minimal Extremes: The plot shows very few outlier scores far from zero, meaning the model rarely assigns strongly positive or negative associations. This further supports that stereotypical or extreme biases have been significantly reduced.

The results confirm that gender and other social biases in the language model's internal associations have been effectively minimized. By reducing unintended associations, the debiased model is now better suited for sensitive legal tasks, such as judgment prediction, document retrieval, or fairness auditing, where biased outputs could have real-world consequences. The shape and characteristics of the distribution validate that the debiasing technique used was effective—it reduced bias without overly disrupting the model's learned representations.

Fig 4.3: Density plot of post-association scores by gender after fine-tuning model on Indian Legal dataset



```
Choose Files  legal-bert-s…3_bec-cri.tsv
• legal-bert-small-uncased-debiased_indian-legal-dataset_lr1e-05_eps1e-08_ep3_bec-cri.tsv(n/a) - 790846 b
Saving legal-bert-small-uncased-debiased_indian-legal-dataset_lr1e-05_eps1e-08_ep3_bec-cri.ts
<ipython-input-2-456c7f1eeabf>:31: UserWarning: No artists with labels found to put in legend
  plt.legend(title="")
```

After debiasing with the Indian Legal Dataset, the male and female distributions show visibly improved overlap compared to the pre-debiasing plot.

Unlike the debiasing results with ECtHR-Gtuned (which showed sharper peaks and tighter variance), the post-debiasing plot with the Indian dataset retains a wide spread, similar to the original (pre-debiasing) distribution. This indicates that while bias has been reduced, the model's prediction confidence and consistency have not improved significantly — likely due to the limited dataset size (only 500 records). The tails of the distribution suggest that some gender-specific associations still persist in outlier cases.

Despite the broader spread, the peak of both distributions is still centered near zero, showing that on average, the model is no longer strongly skewed toward one gender. This is a positive sign that the core bias has been mitigated.

Table 4.1: Calculation of key metrics for analyzing post-association scores for both datasets

Metrics	**ECtHR dataset**	**Indian-Legal-Dataset**
Mean (μ)	0.171920	0.478970
Std deviation (σ)	0.677826	1.078235
Min value	-2.942653	-3.20374
Max value	3.177733	5.394452

For fair comparison between both datasets we have taken 500 records each and performed fine-tuning.

The mean for the Indian dataset is higher, indicating that the post-association scores are generally more positive compared to the ECtHR dataset. This could suggest that the debiasing effect is less pronounced for the Indian dataset.

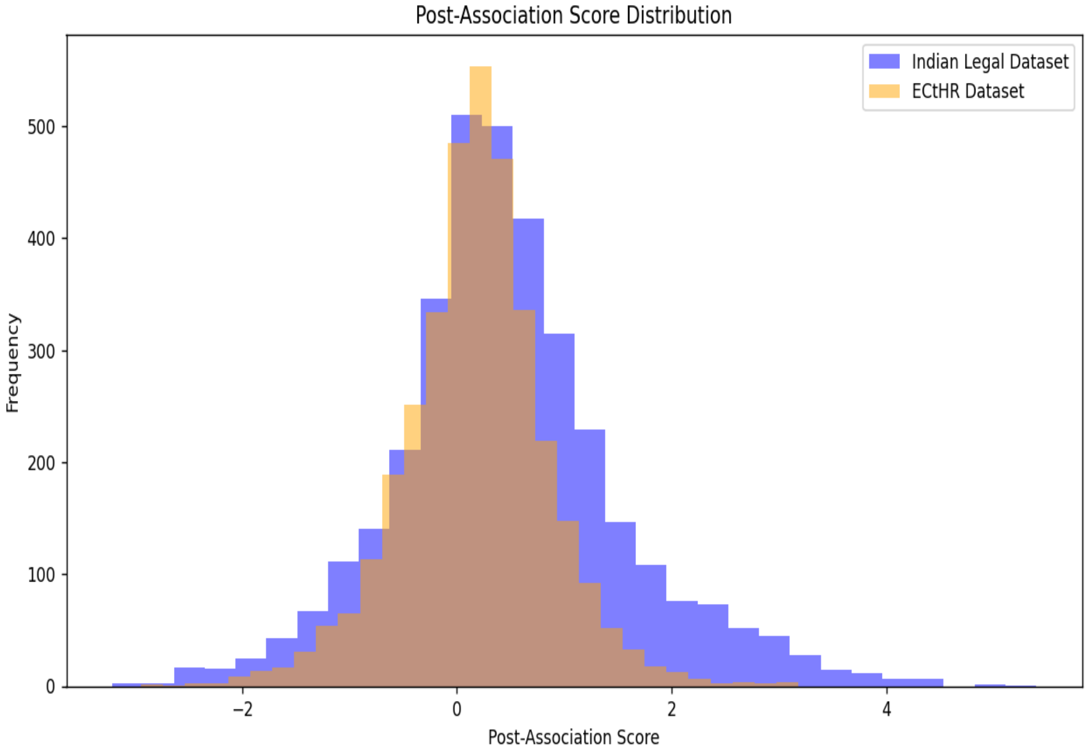
The Indian dataset has a slightly higher standard deviation, meaning the scores are more spread out. Ideally, debiasing should reduce the spread, so this might indicate that the debiasing is not as effective for the Indian dataset.

The Indian dataset has a wider range, with a higher maximum value. This suggests that some associations remain strong even after debiasing.

The Indian dataset has higher quartile values compared to the ECtHR dataset, indicating that a significant portion of the scores are skewed towards higher values. This could mean that the debiasing process is less effective in reducing associations for the Indian dataset.

The metrics suggest that debiasing is less effective for the Indian Legal Dataset compared to the ECtHR dataset. The higher mean, standard deviation, and wider range indicate that associations are not being reduced as much as expected.

Fig 4.4: Histogram plot for post-association scores for ECtHR and Indian-Legal-Dataset



As observed in the plot, we can say that the Indian dataset has higher frequencies for post association scores more than 0 compared to ECtHR. This shows that it's not debiasing the model properly and maybe generating worse scores compared to what we will get before debiasing.

Possible reasons for Indian-Legal-Dataset to not perform so well:

1. Dataset Size:

A smaller dataset (e.g., only 500 records) provides less data for fine-tuning, which can lead to underfitting. The model may not learn the specific patterns or associations effectively.

2. Domain Mismatch:

The Indian dataset focuses on Indian legal cases, which may have linguistic, cultural, or legal nuances different from the datasets (e.g., ECtHR, US contracts) used to pre-train the Legal-BERT model. This domain mismatch can reduce the model's performance. This can be referred to as contextual ambiguity. The facts of our Indian legal cases don't have a proper structure like ECtHR or has references to terms not familiar to the baseline model like FIR, IPC, etc. Even the usage of romanized (Hindi words written in English) words like *Lathi* for stick, *bidi* for cigarette, etc., can cause problems in proper classification task for learning.

3. Pre-trained Model Bias:

Legal-BERT uses a transformer model and was created by studying a range of documents, for example, rulings issued by the European Court of Human Rights and agreements from the United States. As a consequence, the model's representations are mostly affected by the language, legal terms and organization of laws in the regions for which it was trained. Obviously, this early training supports general legal language comprehension, yet it also makes the model focus more on Western legal settings—those in Europe and the US—than on systems with entirely different backgrounds, as seen in India.

Thus, Legal-BERT could not work well for Indian legal language only by fine-tuning on a tiny Indian legal dataset because there are significant differences in style, terms, citation formats and legal settings. Indian laws usually bring together colonial traditions, rules from the region and many languages, meaning stronger efforts are needed for adaptation. If the model doesn't see enough cases and terms similar to Indian laws, it might struggle to understand Indian lawyers' reasoning or manage bias in this area.

4. Data Quality:

The way the Indian legal data is built and organized is important for the success of fine-tuning and debiasing steps. Mistakes in a dataset, whether in formatting, missing judgments or wrong labels of entities, can disrupt the model's ability to study and present legal language in an organized way. Just as, irrelevant content, mistakes in OCR and different legal expressions can introduce confusion and lower the effectiveness of the model's generalization because of the noise in the training data.

5. Fine-tuning Parameters:

How effective LegalBERT is after being fine-tuned depends greatly on the learning rate, batch size, number of training epochs and weight decay. If the parameters are adjusted incorrectly, the model often struggles to adapt, gives unwanted results and shows a drop in performance.

If the learning rate is low, the learning process will move slowly and if it is high, the model could adopt weights that cause the training to stall or wander off the right track. Selecting the wrong batch size can also influence how stable and well the model performs. If the gradient updating is for a small group of weights, it may be too noisy and can damage the accuracy of the model's predictions.

6. Evaluation Dataset:

If the evaluation dataset (e.g., BEC-Cri) is not representative of the Indian legal context, the post-association scores may not accurately reflect the effectiveness of debiasing for the Indian dataset.

After implementing the debiasing techniques, the aim was for these bias scores to shift closer to zero. This shift would reflect a more neutral model with balanced gender associations. Ideally, the scores for male and female targets would also become nearly identical, with minimal differences between them. Such outcomes would signify not only a minimization in gender bias but also the success of the process (debiasing) in achieving fairness while maintaining the semantic and functional effectiveness of the model.

CHAPTER 5

CONCLUSION

Multiple methods were examined in this study for controlling gender bias in law-related language models, especially in the case of LegalBERT. Both using association scores and gender-aligned test sets, we found evidence that LegalBERT shows gender bias which may have come from its previous preparation on common and law-related texts.

To solve this, the researchers developed Legal-Context-Debias (LCD) which trains the model with gender-balanced real-world legal data in a way that focuses on a binary classification problem. As a comparison, LCD was measured against Gender Preserving Debiasing (GPD) and Gender Awareness Preserving Tuning (GAP-based). While GPD succeeded in cutting down some gender differences, it did not greatly enhance results for the overall fairness calculations and operated poorly on a specialized survey. When applying GAP-based fine-tuning to general coreference resolution, we found that it performed unevenly and in some instances led to higher bias, proving that using these methods from different areas can be risky in legal applications.

The study also incorporated a manual collection of just 500 legal records from India. Even though the dataset was not large, it did show a slight improvement in decreasing bias and align the model's actions with gender norms. Still, the scores for post-debiasing association after fixing the biases in the Indian dataset tended to show more variability and occasional confused results compared to ECtHR dataset with the same size. This might be because of different context since our LegalBERT model was pre-trained on a different legal context which is judicial cases from US and Europe. This indicates that we need context aligning data for better results in debiasing.

Even so, the process for improving bias in legal NLP systems is not always straightforward.

- Not Enough Data: Many rules that deal with gender do not clearly label or have enough information on the subject, making it hard to develop fair training data automatically. Even though gender-neutral laws seem fair, models are capable of discovering their understated biases. Legal materials commonly display basic cultural and organizational biases that link certain kinds of crimes or jobs to a single gender. As a consequence, the biased bits of data can unintentionally be built into the models and these biases have a way of increasing stereotypes.
- Formal Laws: Legal documents often use fancy words and, indirectly, refer to gender. Words such as "applicant" or "defendant" have gendered implications that make it more difficult for models to identify and remove bias, unless the context is well understood. The analysis of documents should evaluate bias carefully to keep important legal aspects unchanged. When controlling for bias, the legal information should stay easy to interpret and accurate, while the model treats all individuals fairly.

- Training Bias Mitigation: It costs a lot to get the best results from large transformer models such as LegalBERT; you need GPUs, specific datasets and multiple training phases. Because of these demands, it becomes difficult for ethical AI practices to be carried out widely and equally.

All in all, the results of this study indicate that targeted debiasing approaches should be used in areas such as legal NLP. Most general approaches fail to significantly address bias issues and still maintain high performance. Legal language models should be carefully designed to address bias, by considering how good the data is, how relevant the contexts, how well the meanings fit and if it’s easy for the system to handle. Work going forward must focus on these challenges to create solutions that are fair and can be used on a larger scale in law.

References

1. Mustafa Bozdag, Nurullah Sevim, and Aykut Koc, Dept. of Electrical and Electronics Engineering and UMRAM, Bilkent University, Turkey. Measuring and Mitigating Gender Bias in Legal Contextualized Language Models
2. Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2017. Men Also Like Shopping: Reducing Gender Bias Amplification using Corpus-level Constraints. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, Copenhagen, Denmark, 2979ś2989. https://doi.org/10.18653/v1/D17-1323
3. Aylin Caliskan, Joanna J. Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. Science 356, 6334 (2017), 183ś186. https://doi.org/10.1126/science.aal4230
4. Donghyun Danny Choi, J Andrew Harris, and Fiona Shen-Bayh. 2022. Ethnic Bias in Judicial Decision Making: Evidence from Criminal Appeals in Kenya. American Political Science Review 116, 3 (2022), 1067ś1080.
5. Nurullah Sevim, Furkan Şahinuç, and Aykut Koç. 2023. Gender bias in legal corpora and debiasing it. Natural Language Engineering 29, 2 (2023), 449ś482. https://doi.org/10.1017/S1351324922000122
6. Elliott Ash, Daniel L Chen, and Arianna Ornaghi. 2021. Gender attitudes in the judiciary: Evidence from US circuit courts. Center for Law & Economics Working Paper Series 2019, 02 (2021).
7. Ngo Xuan Bach, Nguyen Le Minh, Tran Thi Oanh, and Akira Shimazu. 2013. A Two-Phase Framework for Learning Logical Structures of Paragraphs in Legal Articles. ACM Transactions on Asian Language Information Processing 12, 1, Article 3 (March 2013), 32 pages.
8. Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. Association for Computational Linguistics. https://doi.org/10.3115/v1/D14-1162
9. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*.
10. Vaswani, Ashish, et al. "Attention is all you need." *Advances in Neural Information Processing Systems* 30 (2017).
11. Radford, Alec, et al. "Improving Language Understanding by Generative Pre-training." *OpenAI preprint* (2018).
12. Peters, Matthew E., et al. "Deep contextualized word representations." *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 2018, pp. 2227–2237.
13. Quantpedia. (n.d.). BERT model – Bidirectional Encoder Representations from Transformers. Quantpedia – The Encyclopedia of Quantitative Trading Strategies. Retrieved May 26, 2025, from https://quantpedia.com/bert-model-bidirectional-encoder-representations-from-transformers
14. Biesialska, K., Biesialska, M., & Rybinski, H. (2020). *Sentiment Analysis with Contextual Embeddings and Self-Attention* [Figure 1: The architecture of ELMo].

ResearchGate. Retrieved May 26, 2025, from https://www.researchgate.net/figure/The-architecture-of-ELMo_fig1_339898853

15. https://indiankanoon.org/
16. https://colab.research.google.com/drive/1Y4o3jh3ZH70tl6mCd76vz_IxX23biCPP#scrollTo=aFbE-UHvsb7-&forceEdit=true&sandboxMode=true
17. https://huggingface.co/nlpaueb/legal-bert-base-uncased

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Shahbad Daulatpur, Main Bawana Road, Delhi-110042, India

PLAGIARISM VERIFICATION

Title of the Thesis: **Measuring and Mitigating Gender Bias in Legal Contextualized Language Models**

Total Pages: **44**

Name of the Student: **Ananya Nayak**

Supervisor: **Prof. Shailender Kumar**

Department of Computer Science & Engineering, Delhi Technological University, Delhi - 110042

This is to report that the above thesis was scanned for similarity detection. Process and outcome is given below:

Software used: **Turnitin,** Similarity Index: **10%,** Total Word Count: **8,492**

Date: 31.05.2025

Candidate's Signature **Signature of Supervisor**

thesis.pdf

Delhi Technological University

Document Details

Submission ID

trn:oid:::27535:98013176

Submission Date

May 27, 2025, 7:48 PM GMT+5:30

Download Date

May 27, 2025, 7:55 PM GMT+5:30

File Name

thesis.pdf

File Size

1.6 MB

44 Pages

8,492 Words

46,690 Characters

10% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 8 words)

Match Groups

40 Not Cited or Quoted 10%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

8%	Internet sources
4%	Publications
7%	Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

40 Not Cited or Quoted 10%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

8%	Internet sources
4%	Publications
7%	Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Submitted works	Delhi Technological University on 2024-05-23	2%
2	Internet	dspace.dtu.ac.in:8080	1%
3	Publication	Sevim, Nurullah. "Analysis of Gender Bias in Legal Texts Using Natural Language ...	1%
4	Internet	www.advocatekhoj.com	<1%
5	Submitted works	Delhi Technological University on 2018-05-17	<1%
6	Submitted works	Delhi Technological University on 2025-05-16	<1%
7	Internet	www.latestlaws.com	<1%
8	Internet	www.coursehero.com	<1%
9	Submitted works	Delhi Technological University on 2025-05-05	<1%
10	Internet	discovery.researcher.life	<1%

11	Publication	"Text Mining Approaches for Biomedical Data", Springer Science and Business Me...	<1%
12	Internet	etd.repository.ugm.ac.id	<1%
13	Internet	tudr.thapar.edu:8080	<1%
14	Submitted works	Cranfield University on 2023-05-14	<1%
15	Submitted works	University of Bristol on 2021-07-10	<1%
16	Internet	github.com	<1%
17	Internet	mro.massey.ac.nz	<1%
18	Submitted works	CSU, San Jose State University on 2025-05-13	<1%
19	Submitted works	dtusimilarity on 2024-05-29	<1%
20	Publication	Bekas, Ioannis. "Overview of Artificial Intelligence Touchpoints With Multi-Criteri...	<1%
21	Internet	arxiv.org	<1%
22	Publication	Hao, Sophie Y.. "Theory and Applications of Attribution for Interpretable Languag...	<1%
23	Internet	m.moam.info	<1%
24	Internet	www.sctce.ac.in	<1%

25	Publication	
Hajiali, Mahdi. "OCR Post-Processing Using Large Language Models", University o...		<1%
26	Publication	
Mustafa Bozdag, Nurullah Sevim, Aykut Koç. "Measuring and Mitigating Gender B...		<1%
27	Submitted works	
Otto-von-Guericke-Universität Magdeburg on 2022-03-17		<1%
28	Submitted works	
University of Durham on 2023-09-18		<1%
29	Internet	
nettt-conference.com		<1%
30	Internet	
www.cnblogs.com		<1%