Hochschule für
Wirtschaft und Recht Berlin
Berlin School of Economics and Law

Berlin School of Economics and Law
Department I - Business and Economics

Detecting Gender Bias in English-German Translations using Natural Language Processing

Bachelor's Thesis

for the attainment of the academic degree Bachelor of Science (B.Sc.)
in the study program
Information Systems Management

Submitted by Khanh Linh Pham

Main Supervisor:	Prof. Dr. Diana Hristova
Secondary Supervisor:	Prof. Dr. Markus Schaal
Semester:	Summer Semester 2025
Matriculation no.:	77211916753
Email:	klpham04@gmail.com
Date of Submission:	**September 01, 2025**

Abstract

Gender bias in English–German Machine Translation often appears in forms such as generic masculine defaulting and occupation stereotyping. These biases can perpetuate unequal representations and feed back into future translation models, reinforcing biased outputs in society. This thesis examines how accurately multilingual BERT (mBERT) can detect such bias. The model was fine-tuned on limited datasets with varying annotation quality, which caused its main limitations. The classifier occasionally (1) misclassifies German gender-fair language forms as biased, (2) fails to detect bias in political and government terms, (3) fails to recognize semantically gendered words as unbiased, (4) is sensitive to punctuation and capitalization, and (5) struggles with sentences that contain both neutral and gendered subjects. Despite these gaps, the model achieved an F1 score of 0.966 and proves effective for core bias cases. It reached 84.6% accuracy on a small handcrafted evaluation dataset with practical sentences like job postings and edge cases. As an intermediary step, the work offers a trained model, sufficiently effective for practical bias detection, and an application that make biased translations visible while indicating areas for further investigation and improvement. The code is available at https://github.com/phmkhali/bias-detector-en-de.

Contents

List of Figures

List of Tables

List of Abbreviations

EN-DE	English-to-German
GFL	Gender-Fair Language
mBERT	Multilingual BERT
MT	Machine Translation
NLP	Natural Language Processing
NMT	Neural Machine Translation

1. Introduction

Modern Machine Translation (MT) tools have made cross-lingual communication more accessible than ever. Services like Google Translate, used by over 200 million people daily (Prates et al., 2019; Shrestha and Das, 2022), offer fast and often accurate translations of full sentences and longer texts. MT is a rapidly growing market, with applications in daily life as well as in fields such as healthcare, law, and business (Kappl, 2025). A recent report by SkyQuest (2025) valued it as a 980 million USD industry in 2023, with projections reaching 2.78 billion USD in the coming years. As the use of MT systems expands, their output has a growing impact on how users access and understand information, raising concerns about the quality and broader implications of these translations.

MT systems are trained on large amounts of text that often contain implicit social biases. Translation models tend to reproduce those rather than correct them. One phenomenon is gender bias, which has been observed across different systems and language pairs (Cho et al., 2019; Soundararajan and Delany, 2024; Smacchia et al., 2024). A common example is the use of gendered terms in the translation of gender-neutral sentences. The English input "The nurse is hard-working" does not specify gender, yet the German output may be "Die Krankenschwester ist fleißig," which assumes a female identity. In contrast, "The surgeon is hard-working" may be translated as "Der Chirurg ist fleißig," implying a male identity. Such translations introduce gendered information that is absent from the source text, reflecting implicit assumptions about certain roles. When these assumptions appear repeatedly in contexts like job descriptions or media reporting that deploy MT, they risk reinforcing societal stereotypes. This contradicts international standards set by organizations like the United Nations, UNESCO, and the European Union, which highlight inclusive language as essential for achieving the Sustainable Development Goals by 2030 (Sczesny et al., 2016; United Nations, 2023).

1.1. Motivation and Research Question

The practical handling of such bias remains limited. Existing research has largely focused on measuring overall bias by counting gendered outputs, comparing them to expected patterns, or testing models on standard benchmarks (Rescigno and Monti, 2023; Barclay and Sami, 2024; Prates et al., 2019; Savoldi, Papi, et al., 2024). These studies provide insights into how often and in what forms bias appears, but they do not offer solutions for detecting biased translations as they occur. The lack of established tools for real-time detection of gender bias is also not a challenge unique to MT; similar issues appear in domains like computer vision

and automated hiring (Schwemmer et al., 2020). As long as biases remain hidden in individual outputs, they are likely to go unnoticed and unchallenged, underlining the need for methods making them visible to users. This thesis addresses these challenges by developing a system for detecting gender bias in English-to-German (EN-DE) translations using Natural Language Processing (NLP). The system is based on a multilingual BERT model (mBERT), which is manually fine-tuned for this task. The outcome is a transparent and efficient classifier that flags biased outputs from MT and is integrated into a demo application, allowing users to check translations for bias in real time. The EN–DE language pair was chosen due to the structural differences between English and German, which make gender bias more visible in translation. It is also supported by several public datasets and prior research. The evaluation is guided by the following research question:

RQ: How accurately does mBERT detect gender bias in EN-DE translations?

To answer this question, datasets from existing literature will be used for training and evaluation. These datasets have certain limitations, which will be examined in detail later in the thesis. To mitigate these, an additional handcrafted dataset will be created to support model selection and evaluation. This approach aims to assess mBERT's performance in detecting gender bias using standard metrics such as accuracy and F1. It also examines how well the model generalizes to unseen data and what limitations result from the system's design.

1.2. Overview of Chapters

First, key concepts are clarified and the fundamentals of BERT are explained to establish the theoretical foundation. The review of related work on gender bias in EN-DE translations then examines existing approaches and points out what they do not cover. The methodology for the bias detector is presented next, detailing dataset handling, training, evaluation, and application design. Implementation details follow, including the code structure and guidance for reproducing the work. The application is subsequently evaluated using the defined datasets and metrics, and the results are summarized and discussed in context. The thesis concludes with an outlook, exploring possible developments and directions for future research.

2. Theoretical Background

This chapter introduces the fundamental concepts used throughout this thesis. It explains key definitions and presents an overview of BERT, focusing on the features necessary for building the detection system.

2.1. Definitions

2.1.1. Natural Language Processing and Machine Translation

NLP enables machine systems to process human language. The goal is to mimic and understand it as fluently as possible (Smacchia et al., 2024; Ullmann, 2022). Common applications are chatbots, translation tools, speech recognition, and image captioning. **MT** is a direct application of NLP. It performs automatic translation of text from one language to another (Lin and Chien, 2009). Over time, MT systems have developed from rule-based approaches, which depend on hand-crafted grammar rules or aligned sentence data, into more adaptable neural models (Chakravarthi et al., 2021).

Most modern systems, such as Google Translate and DeepL, rely on neural machine translation (NMT) (Y. Wu et al., 2016; DeepL, 2021). These models are trained on large collections of translated texts. They learn to represent the meaning of entire sentences as mathematical structures, enabling more fluent and accurate translations. Unlike earlier approaches, NMT systems take the full sentence context into account, which helps reduce errors and improves the handling of ambiguous or idiomatic language (Y. Wu et al., 2016). Throughout this work, all MT systems referenced or applied are neural models.

2.1.2. Bias and its Manifestations

Bias refers to a tendency to favour or disadvantage certain individuals or groups based on preconceived ideas. It often comes from stereotypes, which are fixed and oversimplified ideas about a social group. In short, stereotypes shape assumptions, while bias influences actual behavior and treatment. Bias takes many forms and can be based on characteristics such as age, disability, gender, ethnicity, religion, or sexual orientation (Ullmann, 2022). These biases frequently originate from longstanding cultural and historical beliefs about the expected behavior of different groups. This thesis focuses specifically on gender bias, which is particularly prominent in MT due to the influence of gendered language. Elements such as gendered terms, occupational roles, and grammatical patterns can affect translations and often perpetuate

stereotypes because language is closely tied to our thoughts and beliefs. Drawing on key studies that examine gender bias in EN-DE MT (Ullmann, 2022; Rescigno and Monti, 2023; Lardelli et al., 2024; Kappl, 2025), such bias typically manifests in the following forms:

Defaulting to Masculine Forms

In both singular and plural contexts, the *generic masculine* uses the masculine grammatical gender as the default. For example, the sentence "Die Studenten sind im Hörsaal" (The students are in the lecture hall) uses the masculine plural form to refer to a group of students regardless of their gender. It is commonly used in spoken German and other gendered languages (Lardelli et al., 2024; Schmitz, 2022), although research has consistently shown that the generic masculine creates a male bias in mental representations, leading readers or listeners to think more of male than female examples (Sczesny et al., 2016).

Reinforcement of Stereotypes

The gendered language patterns discussed earlier reflect broader social beliefs about men's and women's roles in work and family life. Although many of these roles no longer reflect reality, they continue to shape judgments about people's abilities and personalities. This often leads to correspondence bias, where traits are inferred based on behavior or circumstances (Godsil et al., 2016). Such stereotypes are reinforced by media, including television and advertising, and influence how language is used and understood. One common result of this is stereotypical job associations. People often link professions like doctors or pilots with he/him pronouns, and professions like nurses or flight attendants with she/her pronouns (Shrestha and Das, 2022). Prates et al. (2019) also found clear patterns in how gender is associated with certain traits. Adjectives like "shy," "happy," "kind," and "ashamed" are often linked to women, while words like "arrogant," "cruel," and "guilty" are more often linked to men.

2.1.3. Gender Bias

A clear definition of gender bias in MT does not exist, nor is there a standard method to identify indicative features in text (Barclay and Sami, 2024). This leads this study to use a simple rule-based definition to determine when a translation of a sentence is gender biased.

- A gender-ambiguous subject in the source text is translated with a gendered term, often by defaulting to the generic masculine (e.g., doctor → Arzt) or reflecting stereotypical gender roles (e.g., nurse → Krankenschwester).
- A gendered subject in the source text is assigned an incorrect gender in the translation, leading to semantic inconsistency (e.g., my mother is an engineer → meine Mutter ist ein Ingenieur).

This does not mean that all other cases are truly "unbiased". Anything that does not fall under these two cases will be referred to as "neutral". This includes, but is not limited to:

- Sentences with no gendered terms, like "The weather is nice".
- Accurate translations of gendered input, like "The woman is a coder" → "Die Frau ist eine Programmiererin".
- The use of gender-fair alternatives (see subsubsection 3.4).

Biased Translation	**Neutral/Fair Translation**
Gender-ambiguous source is translated with a gendered term.	Gender ambiguity is preserved in the translation.
Gendered subject is assigned an incorrect gender.	Gender in the translation matches the gendered subject.
—	Use of gender-fair language alternatives (see subsubsection 3.4).

Table 1.: Summary of gender bias scenarios in translation (original compilation)

2.1.4. Binary Classification in Natural Language Processing

Binary classification means sorting items into two clear groups. It is the most common task in Machine Leaning (ML) and is frequently found in every day life, such as automatically flitering e-mails as "spam" or "not spam" (Quemy, 2019) or deciding whether a transaction is "fraudulent" or "legitimate". For instance, a spam filter uses previously labelled e-mails to learn relevant patterns by looking at specific keywords or sender information, and builds a model that applies these patterns to classify new messages. This thesis tries to label a translation as either "biased" or "neutral". For example, the translation "The nurse is kind" → "Die Krankenschwester ist nett" would be labelled as biased, whereas a gender-neutral translation such as "Die Pflegekraft ist nett" would be labelled as neutral. While it is possible to extend the classification beyond two categories to distinguish types of bias or include "gender-fair" labels, doing so would require substantially more data and training. Given the practical aim of this work, the simpler binary approach is more suitable.

2.2. BERT

BERT, which stands for Bidirectional Encoder Representations from Transformers, is a language model that was introduced by Google in 2018 (Devlin et al., 2019). After pre-training, it can be adapted to various NLP tasks, such as text classification, sentiment analysis, or question

answering, by adding a simple output layer and fine-tuning on task-specific data. This output layer produces the final prediction, for example by assigning a label to a piece of text, without requiring major changes to the original architecture. BERT's strong language understanding makes it well suited for binary classification tasks. There are multiple variants of the original BERT model. It was originally released in two sizes: `BERT-Base` and `BERT-Large`, which differ in the number of layers, attention heads, and overall model capacity (Devlin et al., 2019). Since then, many other versions have been developed. Most of them modify either BERT's pre-training objectives or the underlying Transformer architecture (Libovický et al., 2019).

2.2.1. Transformer Architecture

BERT is built on the transformer neural network architecture, which is the component that processes text using self-attention to capture context across all words in a sentence (Phuong and Hutter, 2022). This mechanism lets the model weigh the importance of all input elements at the same time (Xiao and Zhu, 2023), so it can consider every word in a sentence and determine which ones are most relevant to each word. Unlike traditional methods such as Recurrent Neural Networks (RNNs) that process input step by step, self-attention captures global dependencies and contextual relationships more accurately, producing "context-aware" representations.

The transformer architecture consists of two main components: the encoder and the decoder. The encoder's job is to read the input sentence and turn it into a series of vectors the model can understand. Each vector is a list of numbers representing the meaning and structure of each word (Xiao and Zhu, 2023). The encoder works as follows (see Figure 1):

1. It receives input embeddings, which represent the words, and positional encodings, which tell the model the order of the words.

2. The data then passes through several identical layers. Each layer has two main components. Each of these is followed by an **Add & Layer Norm** step, which helps stabilize and preserve useful information:

    a. **Multi-head self-attention** runs several self-attention processes in parallel. Each attention head focuses on different details to help the model understand the sentence better.

    b. A **Feed-forward network** processes each word vector separately, refining the information like a small filter.

3. Each layer builds on the output of the previous one, helping the model form more complex and abstract ideas about the input sentence.

4. Finally, the encoder outputs a sequence of *hidden states*. These are continuous vector representations for each input token. They encode contextual information from the entire

sentence. For example, in the sentence "The cat sat on the mat," the vector for "cat" reflects its relationship to words like "sat" and "mat."

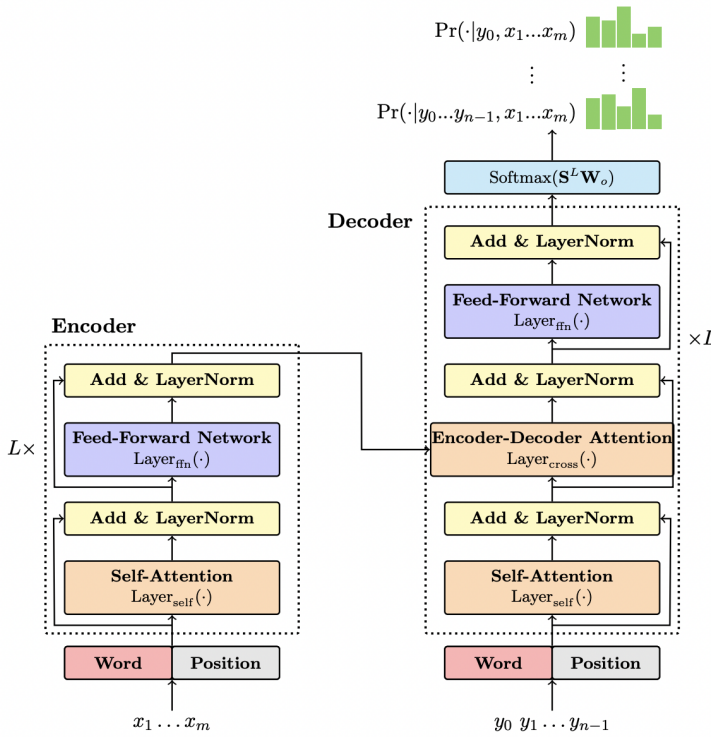

Figure 1.: Transformer encoder-decoder architecture. The encoder (left) processes input tokens $x_1, \ldots, x_m$ through: (1) a self-attention layer for contextual relationships, (2) a feed-forward network for feature transformation, and (3) residual connections with layer normalization. The decoder (right) generates outputs by attending to both the encoder's representations and its previous outputs (y_0 to y_{n-1}), producing the next-token probability distribution. Figure and description adapted from Xiao and Zhu (2023), p. 6

The decoder generates the output sentence one word at a time by using the information from the encoder (Xiao and Zhu, 2023). However, since BERT uses only an encoder-only architecture (see Figure 2), the decoder is not relevant for this work and is therefore excluded from the discussion.

2.2.2. Multilingual BERT

In this thesis, the model used is multilingual BERT (`mBERT`) (Devlin et al., 2019). `mBERT` has the same architecture as `BERT-Base` but was pretrained on Wikipedia data from 104 languages, including English and German. The model does not receive any explicit signal about which language it is processing. As in the aforementioned "The cat sat on the mat" example, suppose

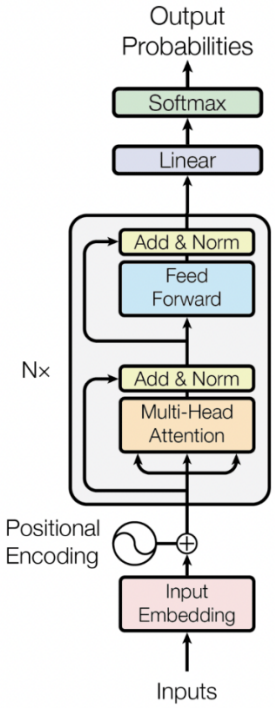

Figure 2.: BERT's encoder-only architecture Figure by Smith (2024)

mBERT sees both "The cat sits on the mat" in English and "Die Katze sitzt auf der Matte" in German during training. The model is never told which language a sentence is in; there is no signal saying "this is English" or "this is German." By seeing many such examples, mBERT learns that words like cat and Katze often occupy similar positions in sentences with similar meanings. This allows it to recognize that they play the same role, even without explicit language labels. Its multilingual ability therefore emerges from patterns shared across languages, enabling internal representations that support tasks in multiple languages (Pires et al., 2019).

`mBERT` was chosen because it offers a good balance between language coverage, model size, and training efficiency. Monolingual models like `German BERT` do not support English input. Larger multilingual models, such as `XLM-RoBERTa`, require more computational resources and training time, which was not feasible here. This makes `mBERT` a practical choice for handling both languages within limited resources.

2.2.3. Tokenization

Before `mBERT` can process any text, the input must be converted into a format the model can understand. To achieve that, `mBERT` splits words or subword units into *tokens*. This process is called tokenization.[1]. It uses the WordPiece algorithm with a shared vocabulary of 110,000

[1]This tokenization process applies to both BERT and mBERT.

tokens (Devlin, 2018). To balance the training data, languages with large Wikipedia corpora are downsampled, meaning fewer examples are used, while those with fewer resources are oversampled, meaning some examples are repeated to increase their presence. Pre-processing is the same for all supported languages: (1) converting text to lowercase and removing accents, (2) splitting punctuation, and (3) tokenizing based on whitespace. Removing accents helps reduce the vocabulary size, even though it can introduce ambiguity in languages where accents carry meaning. This trade-off is accepted because `mBERT's` contextual embeddings usually resolve such ambiguities during training and inference. In addition to tokenizing words and subwords, `mBERT` relies on *special tokens* to provide structural information. These tokens, such as [CLS] for the start of a sentence or [SEP] to separate segments, are not real words but placeholders that help the model understand the role of different parts of the input. They work together with the tokenized embeddings to give the model a complete representation of the text.
In this work, each input combines an English source sentence and its German translation as:

```
[CLS] english sentence [SEP] german translation [SEP]
```

```
[CLS] the nurse is kind [SEP] die krankenschwester ist nett [SEP]
```

2.2.4. Fine-Tuning

Fine-tuning adjusts the base model for a specific task, in this case, detecting gender bias in translations. To do so, a new labelled dataset is used to continue training the model, allowing it to adapt its weights to task-specific patterns. In the context of this thesis, adapting the weights means that the model learns to recognize patterns in translations that indicate biased or neutral gender representations.[2] A *classification head* is an additional layer added to the top of the model to turn its general language understanding into task-specific predictions. It usually consists of a *linear layer*, which transforms the model's output into a set of scores, followed by a *softmax function*, which converts these scores into probabilities for each class. Here, the classification head uses the final hidden state of the `[CLS]` as the input. The linear layer maps this vector to two values (biased or not biased), and the softmax function outputs the probability for each class.[3]

$$z = Wx + b$$

x is the `[CLS]` embedding, W is the weight matrix, and b is the bias vector. Both W and b are parameters learned during training to help map `mBERT's` output to the task labels. This changes the output into two numbers (logits), one for each class: biased or neutral. Then, the softmax function turns these numbers into probabilities (Devlin et al., 2019; Xiao and Zhu,

[2] This fine-tuning process applies to both BERT and mBERT.
[3] The following formulas are adapted from Devlin et al. (2019) and Xiao and Zhu (2023)

2023). Short for "soft maximum," it maps raw scores to a probability distribution, emphasizing the highest values while still giving smaller ones some weight.

$$\text{softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

Each logit z_i is exponentiated to ensure positivity. The result is then normalized by dividing by the sum of all exponentials, producing the probability distributions. K is the number of possible classes. The class with the highest probability is selected as the model's prediction. For example, suppose the model outputs logits $[1.5, 0.5]$ for biased and neutral respectively. Exponentiating gives $[e^{1.5}, e^{0.5}] \approx [4.48, 1.65]$. Normalizing by the sum $4.48 + 1.65 = 6.13$ gives probabilities $[0.73, 0.27]$. The model would predict biased since it has the higher probability, with a confidence of 73%.

2.2.5. Key Hyperparameters

Fine-tuning can be unstable, and changes such as different seeds can lead to large differences in task performance (Mosbach et al., 2021). It is therefore necessary to tune a set of key hyperparameters, which are settings that control how the model is trained. These are not learned by the model but must be set manually or through experimentation. Their values affect how fast the model learns, how stable training is, and how well the model generalizes to new data. The commonly tuned hyperparameters are briefly introduced below.

The *learning rate* controls how much the model updates its weights during each step (Mosbach et al., 2021). If it is too high, the model may not converge and instead jump over good solutions. If it is too low, training can be very slow or get stuck in local minima.

Warmup steps are used at the beginning of training to gradually increase the learning rate from zero to its target value (Mosbach et al., 2021). This helps avoid instability in the early stages, where large updates can be harmful. After the warmup period, the learning rate is often decreased again using a scheduler, which controls how it changes over time.

The *number of epochs* defines how many times the model passes through the entire training dataset (Mosbach et al., 2021). More epochs mean more training iterations, which can help the model better fit the data. On small datasets, training for too few epochs can cause underfitting, where the model does not learn enough from the data and performs poorly even on the training set. Training for more epochs, sometimes up to 20 instead of the usual 3, helps reduce underfitting and improves generalization. However, training for too many epochs can lead to overfitting, where the model learns the training data too closely and performs worse on new data.

The *batch size* refers to how many training examples the model processes before updating its parameters (Mosbach et al., 2021). Commonly, a batch size of 16 is used during fine-tuning `mBERT`. Larger batches provide more stable gradient estimates but require more memory. Smaller

batches can introduce noise in the updates but might help the model generalize better. While Mosbach et al. (2021) does not deeply analyse batch size effects on stability, it remains an important parameter to balance resource limits and training quality.

Finally, the *optimizer* controls how the model weights are adjusted to minimize prediction error (Mosbach et al., 2021). The AdamW optimizer is standard for `mBERT` fine-tuning because it adapts learning rates per parameter and includes weight decay regularization. A critical feature of Adam is *bias correction*, which reduces the effective learning rate early in training. This acts like an implicit warmup, preventing large unstable updates and vanishing gradients in the lower layers. Combining explicit warmup with Adam's bias correction allows training with higher learning rates more stably.

Hyperparameter	**Role in Fine-Tuning**
Learning Rate	Controls how much model weights are updated at each step; too high causes instability, too low slows training.
Warmup Steps	Gradually increases the learning rate at the start to prevent unstable early updates.
Number of Epochs	Defines how many times the model sees the full training data; more epochs help on small datasets.
Batch Size	Number of samples processed before an update; affects stability, memory use, and generalization.
Optimizer	Algorithm for updating weights; AdamW is standard, with adaptive rates and weight decay.

Table 2.: Summary of key hyperparameters used during fine-tuning

2.2.6. Layer Freezing

To speed up training and help prevent overfitting on small datasets, while preserving the broad language knowledge from pre-training, it is common to freeze certain layers of a pretrained model during fine-tuning. Layer freezing refers to keeping these layers fixed, meaning their weights are not updated. This reduces the number of trainable parameters (Sorrenti et al., 2023). In monolingual BERT, lower layers typically encode general syntactic and semantic patterns, while higher layers are more task-specific (Nadipalli, 2025). As a result, lower layers are often frozen, and only the top layers and the classification head are fine-tuned, especially in resource-constrained settings (Nadipalli, 2025).

In `mBERT`, the distribution of cross-lingual and language-specific features across all layers makes layer freezing less straightforward. S. Wu and Dredze (2019) highlight that no single layer consistently captures the most relevant cross-lingual information, and even individual layers can perform well on sentence-level tasks. They suggest that freezing the lower six layers may improve generalization, but emphasize that optimal strategies depend on the specific task

and require empirical testing (S. Wu and Dredze, 2019).

2.2.7. Limitations of mBERT

One major limitation of `mBERT` is the "curse of multilinguality" (Gurgurov et al., 2024). Because it must represent 104 languages within a fixed parameter budget, the capacity available per language is limited. This causes reduced performance across languages compared to monolingual models. Even high-resource languages like English perform worse in `mBERT` than in their dedicated BERT models. Additionally, the shared vocabulary of 110,000 tokens is diluted, meaning it is less tailored to any single language. Languages with more data tend to get better performance, while others suffer. Since `mBERT` is pretrained on Wikipedia, it reflects biases inherent to that corpus. German Wikipedia articles predominantly use the generic masculine (Sichler and Prommer, 2014), while gender-fair alternatives appear only sporadically, mostly in discussions or articles about female-dominated professions. These biases can influence the model's outputs and are especially important to consider in a gender bias detection context. Despite these limitations, `mBERT` remains the most fitting choice for this thesis. Since I work with English and German, which are both high-resource and related languages, `mBERT` generally performs better than it would with low-resource languages or languages from distant language families with fewer similarities (Lauscher et al., 2020).

2.2.8. Evaluation Metrics

The fine-tuned BERT model must be evaluated to determine how accurately it detects gender bias. Evaluation metrics provide objective measures for assessing and comparing performance. In this task, it is especially important to reduce two types of errors: false positives, where unbiased translations are mistakenly flagged as biased, and false negatives, where genuine bias goes undetected. A model that guesses randomly or consistently avoids flagging bias offers little practical value. The metrics that capture these errors are precision and recall (Rainio et al., 2024):

- **Precision:** Of all translations flagged as biased, how many truly are biased? High precision means fewer false alarms.
- **Recall:** Of all biased translations, how many did the model correctly detect? High recall means fewer missed biases.

There is often a trade-off between precision and recall. A model with high precision but low recall misses many real biases, while one with high recall but low precision raises too many false warnings. To balance this trade-off, the F1 score is used. It combines precision and recall into a single number by calculating their harmonic mean:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

Another common metric is accuracy, which measures the percentage of all translations that are classified correctly (Rainio et al., 2024). Accuracy is straightforward and gives a sense of overall performance, but it can be misleading for imbalanced datasets. For example, if most translations are unbiased, a model that always predicts "unbiased" would achieve high accuracy but fail to identify any biased instances. The F1 score is better for evaluating the model and guiding selection because it focuses on the minority class of biased translations. Accuracy, however, remains useful as a complementary metric, particularly when assessing performance on a controlled test set. On the handcrafted test set, it shows how often the model predicts the correct label, giving a clear sense of its performance alongside the F1 score.

3. Related Works

The EN–DE language pair has been examined in studies on gender bias, but research on dedicated bias detection systems is still limited. This chapter builds on the earlier definitions and reviews existing work to show how the problem has been addressed so far and where important gaps remain.

3.1. Literature Search Process

For the literature review, incremental and conceptual methods were combined, with each source leading to the identification of subsequent ones. Based on this progression, key concepts were identified and used to organize and interpret the literature, aligning with a conceptual approach. The structure followed the qualitative Information Systems framework by Schryen (2015) and was further informed by Shrestha and Das (2022) and Savoldi, Bastings, et al. (2025), both of whom conducted systematic reviews on gender bias in ML and MT respectively.

Sources were primarily searched on Google Scholar and Perplexity[1], which served as an additional search engine. Prompts and outputs from Perplexity have been saved and are included in the appendix. To organize and manage the collected sources, Zotero was used throughout the process. Table 3 defines the concepts that guided the literature search. Key search terms consisted of *gender bias*, *machine translation*, *artificial intelligence*, *machine learning*, *German*, *stereotypes*, and *detection*, which were combined with *AND/OR*. The focus was on literature published between 2019 and 2025 to maintain relevance and currency, while foundational and definitional works from earlier periods were selectively included.

The initial search for the term *gender bias in machine translation* returned over 18,000 results. Sources were first screened manually by reviewing titles and abstracts to identify relevance. After the first pages of results, entries became increasingly unrelated to the scope, often being overly technical or addressing broader areas beyond the focus of this thesis, and were excluded. Potentially relevant sources were entered into a table including author, title, concept, synthesis, novel contribution, limitations, and notes.[2] Analysis focused primarily on relevant chapters, with full texts reviewed when appropriate. Inclusion required that studies specifically addressed gender bias in MT, provided examples or discussions of gender-related errors, or explained the significance of gender bias in MT. Sources also had to be available in full text without access restrictions. Exclusion criteria removed studies focusing on general NLP bias without a

[1] Refer to Appendix C for the application of Perplexity.ai.
[2] Refer to Appendix A for the full analysis table.

Key Concept	Description
Defining Gender Bias in MT	Defines the core concept of gender bias in MT, including common bias patterns like gendered term insertion and incorrect gender assignments. Sets the conceptual foundation for the thesis.
Relevance and Existing Research	Establishes the importance of studying gender bias by reviewing related work. Highlights key findings and their implications for fairness.
Research Gaps and Open Challenges	Identifies the main gap: the absence of reliable detection systems for gender bias in EN-DE MT. Discusses the lack of a shared fairness definition and limitations in existing datasets.
Technical Design and Implementation	Explains the theoretical background and fundamental principles necessary to understand the implementation. Covers the underlying concepts that guide design choices and system functionality.

Table 3.: Key concepts relevant to this thesis

direct connection to MT, non-gender biases, and technical works not contributing to a broader understanding of gender bias or offering no additional insights beyond previously reviewed literature. Through this process, the initial 18,000 results were narrowed to 15 core sources.

Inclusion Criteria	Exclusion Criteria
Addresses gender bias in MT	Focuses on general NLP bias without link to MT
Provides examples or discussion of gender-related errors	Covers non-gender-related biases
Explains the significance of gender bias in MT	Highly technical with no added general insight
Available in full text without access restrictions	Redundant or not contributing new perspectives

Table 4.: Summary of selection criteria for literature review

Backward citation searching involved reviewing references cited by selected papers, prioritizing frequently cited and foundational works relevant to gender bias in MT. Forward citation searching used Google Scholar's "cited by" function to identify newer research citing those key papers. Filtering with specific terms (e.g., *German* and *machine translation*) was applied during forward search to maintain focus. Beyond these systematic methods, supplementary sources were also

incorporated as needed to address specific informational gaps. These consist of contextual references, statistics, or secondary citations that support specific points but were not part of the core conceptual or methodological framework. Supplementary sources were defined as materials identified outside the systematic search, such as papers found through backward citations or targeted queries for statistics and news, which provided support for subordinate arguments without being central to the study's theoretical or analytical structure.

3.2. Foundational studies

First mentions of this issue date back to over a decade ago, having been recognized by a paper by Schiebinger (2014). Since then, there has been a general increase in research papers focusing on this topic, especially between 2019 and 2023 (Savoldi, Bastings, et al., 2025). Prates et al. (2019) conducted a large-scale study using Google Translate to translate sentences like "[Gender-neutral pronoun] is an engineer" from twelve gender-neutral languages into English. The results showed a strong bias toward male pronouns, especially in science, technology, engineering and mathematics (STEM) occupations (see Figure 3 and 4). This could not be explained by real-world labour statistics, pointing instead to imbalances in the system's training data. The study received wide media attention, leading Google to change their translation policy: Google Translate began showing both feminine and masculine forms for ambiguous inputs (Google, 2018) (see Figure 5).

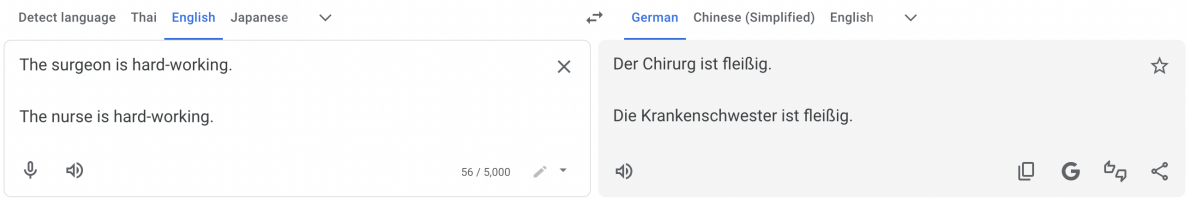

Figure 3.: Google Translate translates an occupational term with a gender stereotype, using the masculine form for "surgeon"

Building on this, Stanovsky et al. (2019) created WinoMT, a benchmark for evaluating gender bias in English-to-multilingual translations. It focused on occupations in contexts designed to challenge stereotypes. The study found that systems were more accurate for stereotypical gender roles but struggled in non-stereotypical cases, confirming the trends observed by Prates et al. Together, these studies helped spark the ongoing research interest in gender bias in MT, with subsequent work consistently confirming the widespread tendency to default to male pronouns and reinforce occupational stereotypes (Lardelli et al., 2024; Cho et al., 2019).

According to Ullmann (2022), translation errors stem from biases present in the training data. The MT systems learns gender associations from word co-occurrences, such as "doctor"

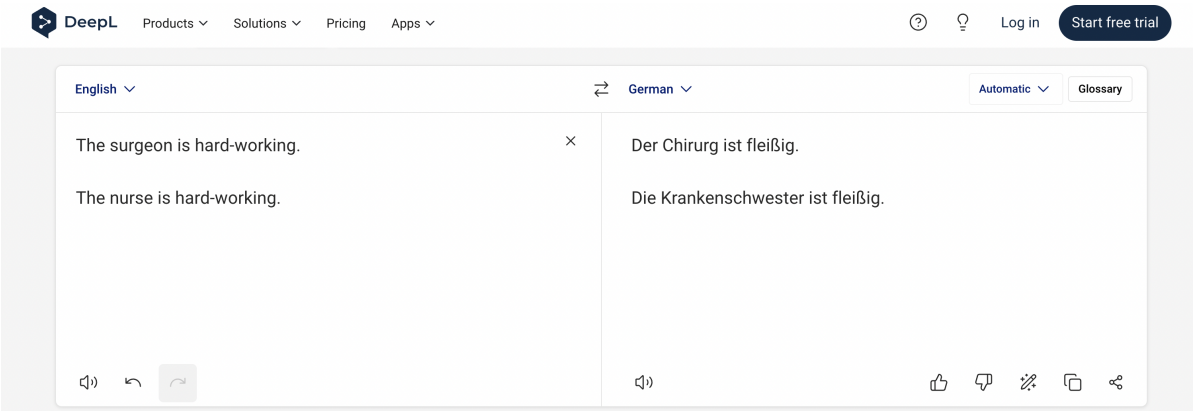

Figure 4.: DeepL translates the same occupational term with a gender bias, mirroring Google Translate's masculine default for "surgeon"

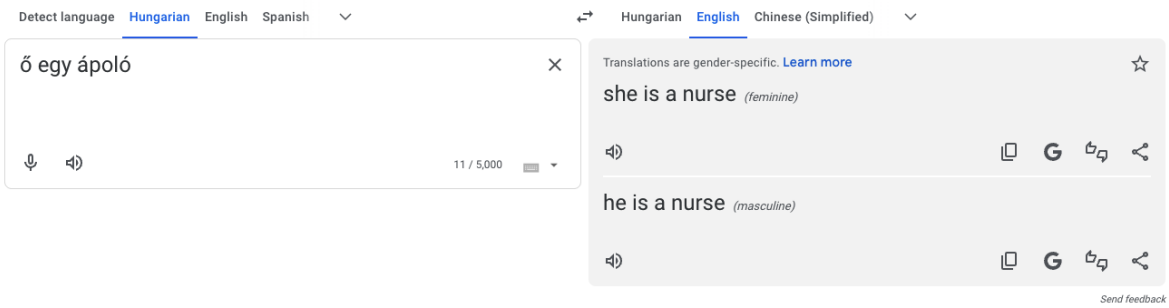

Figure 5.: Google Translate assigns gendered pronouns in translation for an originally gender-ambiguous subject

with masculine pronouns, causing incorrect or inserted gender in translations. It also amplifies existing biases during training like linking cooking predominantly with women, which leads to gendered outputs not supported by the input. The large size of training corpora increases the challenge of controlling data quality. Manual inspection becomes impractical when models are trained on hundreds of billions of tokens. Consequently, the model can unintentionally absorb and reproduce harmful or biased content, reinforcing patterns that lead to biased translations (Ullmann, 2022).

3.3. Implications of Bias

Biases do not only cause translation errors but also have wider social consequences. They can lead to representational harm by repeatedly portraying certain genders in limiting or stereotypical ways (Stanczak and Augenstein, 2021). Since these biased outputs can re-enter the training data and influence future MT models, the cycle of biased representation continues and reinforces itself in society, creating a regressive feedback loop as portrayed in Figure 6.

The generic masculine in particular leads to inaccurate and unfair representations of gender in translated text. Rescigno and Monti (2023) observed a predominance of masculine forms in translation outputs (approximately 90% in Google Translate and 85–88% in DeepL for EN-IT and EN-DE), even when the original sentences contained relatively few masculine references. This shows that the bias is not minor but occurs quite heavily in those systems. It also contributes to the marginalization of women in male-dominated professions (Kappl, 2025). Studies show that biased language in machine-generated text, such as children's stories or job ads, can influence how young people view themselves (Soundararajan and Delany, 2024; Kappl, 2025). It may shape their interests, hobbies, and career choices. This is especially visible in Science, Technology, Engineering, and Mathematics (STEM) fields (Prates et al., 2019), where stereotypes are more persistent. When job descriptions or mock interviews use gender-exclusive pronouns, women report feeling less belonging, lower motivation, and weaker identification with the role (Godsil et al., 2016). Many self-select out of applying, shrinking the female talent pool and reinforcing gender gaps in the workforce.

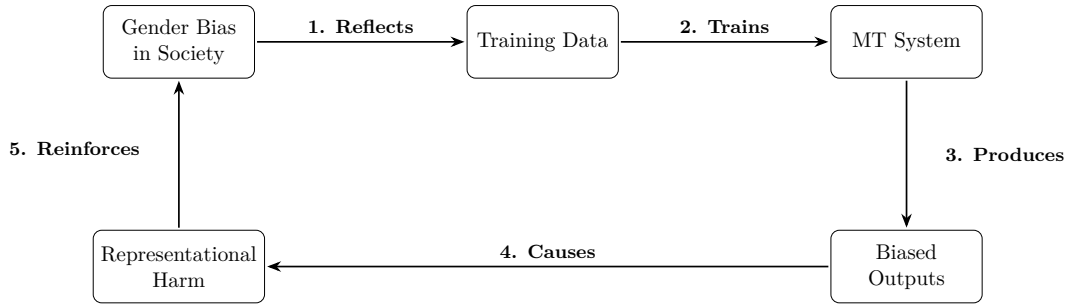

Figure 6.: Regressive feedback loop of gender bias in MT

Research also shows that using Gender-Fair Language (GFL) like "she and he" or "one" can improve how women respond to job ads. It reduces stereotype threat and helps them engage more positively with opportunities (Godsil et al., 2016). Furthermore, Savoldi, Papi, et al. (2024) investigated the effort required to correct gender-biased translations. Participants performed a post-editing task in which they revised the same sentences twice, once for each gender, ensuring human references were correctly rendered as feminine or masculine. The study measured factors such as editing time and the number of changes needed, isolating the additional effort caused by gender bias. The results showed that fixing translations with feminine forms took almost twice as long and required four times more edits than those with masculine forms. As a result, biased translations lead to higher economic costs and a quality gap that disproportionately affects women. Savoldi, Papi, et al. (2024) argued that current automatic bias metrics miss these human impacts. They called for better evaluation methods that reflect what users actually experience.

3.4. English-German Linguistic Challenges

Although both English and German originate from the Indo-European language family (Baldi, 2008), they have different liguistic characteristcs. English does not assign grammatical gender to nouns. The article "the" is used universally, independent of what it refers to. On the contrary, German assigns one of three grammatical gendered articles to nouns: "der" (m), "die" (f) and "das" (n). The form or ending of a noun may also change depending on its grammatical gender. While English has a few gendered word pairs, such as "actor" (m) and "actress" (f), gender distinctions in German apply broadly across the entire noun system. "Der Student" refers to a male student, whereas "die Studentin" refers to a female student. Note that grammatical gender has no connection to societal or biological gender. It is a rule of the language rather than a reflection of identity. For example, the German word Mädchen (girl) is grammatically neuter and takes the article "das". This is not because the referent lacks gender, but because the suffix "-chen" automatically assigns neuter gender. Grammatical gender in German follows structural rules, even when they contradict real-world gender associations.

German Gender-Fair Language

GFL is the use of language that treats all genders equally and aims to reduce stereotyping and discrimination (Sczesny et al., 2016). Three common approaches to plural mentionings in German are:

- **Gender-neutral rewording:** This uses neutral terms instead of gendered nouns, e.g., *die Studierenden lernen.* A challenge for this version is that neutral alternatives do not exist for every noun and cannot be consistently applied (Lardelli et al., 2024).
- **Gender-inclusive characters:** This combines masculine, feminine and non-binary forms by using a character like *, :, or _, e.g., *die Student*innen lernen.* This method is consistent but may interrupt reading flow and lacks standardization (Lardelli et al., 2024).
- **Pair form:** This names both gender forms, e.g., *die Studentinnen und Studenten lernen.* It is currently the most used GFL form in German (Waldendorf, 2024), briefly surpassing the star and colon characters as seen in Figure 7.

These examples apply when the gender of the subjects is ambiguous. But when gender is known, especially in singular mentions, the generic masculine should be avoided. However, in the same way as gender bias has no clear definition, there is no agreed standard for GFL (Lardelli et al., 2024; Savoldi, Bastings, et al., 2025). "Fairness" therefore heavily depends on personal views, culture, and context, which raises ethical questions about debiasing systems.

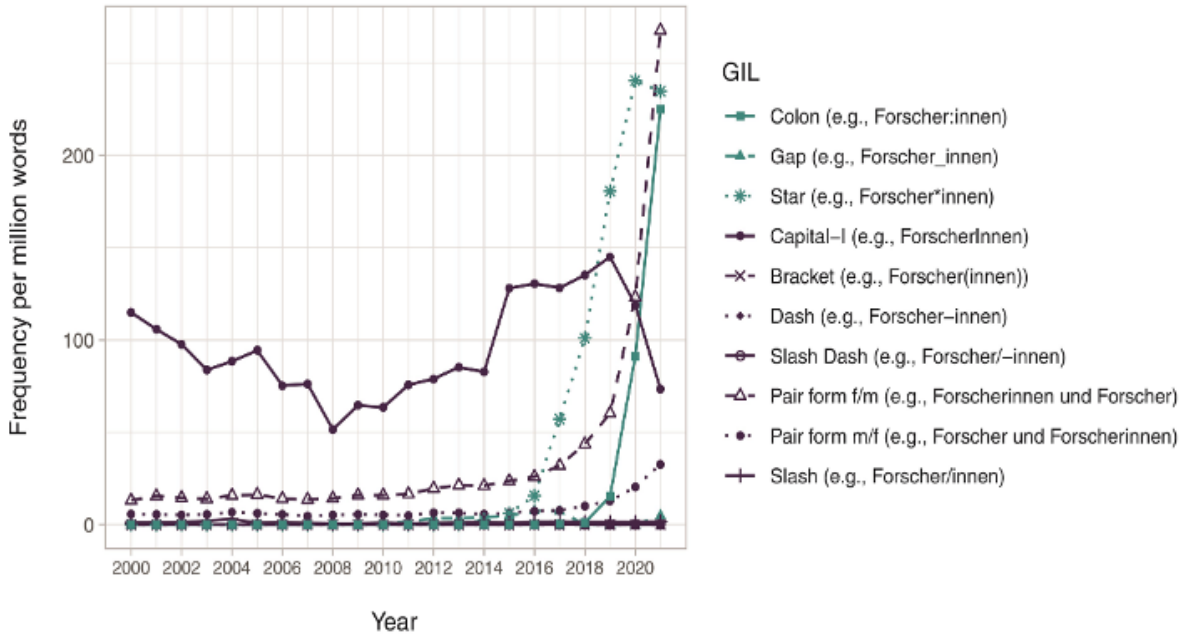

Figure 7.: Frequency of different types of gender-inclusive language. Source: Waldendorf (2024) p. 367

The use of GFL has increased in recent years (Waldendorf, 2024), but it remains generally low. This results in a scarcity of relevant linguistic data. Few datasets include GFL variants, and existing resources often rely on manual translations or post-editing to add gender-inclusive forms (Lardelli et al., 2024).

3.5. Research Gaps

A central gap in gender bias research is the absence of a shared definition of what constitutes "fair" language. This lack of conceptual clarity makes it difficult to design systematic evaluation approaches, define accountability standards, or detect all relevant forms of harm (Barclay and Sami, 2024; Shrestha and Das, 2022; Stanczak and Augenstein, 2021). A second major gap concerns the availability of high-quality EN-DE translation data containing GFL. While a few datasets exist, they are not designed for bias detection and often require manual post-editing to include inclusive forms (Lardelli et al., 2024). This lack of consistent GFL examples limits the ability to develop and evaluate models in a structured and reproducible way, and makes it harder to train systems to recognize gender-fair alternatives as neutral.

Stanczak and Augenstein (2021) note that findings on gender bias in English do not always apply to other languages such as German. Linguistic differences make language-specific approaches necessary. Studies on EN–DE systems (Ullmann, 2022; Kappl, 2025; Lardelli

et al., 2024) confirm the presence of gender bias, propose mitigation strategies, or introduce evaluation metrics. Yet, only a few focus on systematic methods to detect bias in translated text. This study addresses that gap by treating bias detection as a prerequisite for any effective mitigation strategy. Since reliable automatic debiasing techniques are not yet available, manual intervention will remain necessary in practice. The primary objective is therefore to identify biased translations with high accuracy, providing a basis for subsequent correction or debiasing efforts.

4. Methodology

The goal of this project is to develop a practical gender bias detection model tailored for real-world MT scenarios. It targets common use cases like translating everyday sentences or job descriptions, focusing on flagging biased language at the sentence level. This means the model evaluates each sentence independently, without considering context from its surrounding. This approach guides both the model's design and the preparation of the training data, where each translation pair is treated as a separate example. The project begins by selecting and combining datasets from previous work (see Figure 8). The model building phase then follows, as shown in the purple boxes. It starts with cleaning and preparing the data, followed by extracting features for training. A pre-trained `mBERT` model is then fine-tuned for the classification task. Its performance is measured using standard evaluation metrics. In the final step, the trained model is integrated into the demo application.

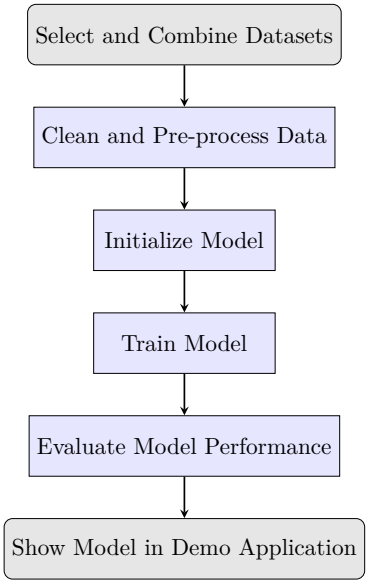

Figure 8.: Methodology Overview

4.1. Dataset

Since no ready-to-use dataset existed for this task and no prior work had developed a comparable model, it was necessary to define: (1) the required number of samples, and (2) the desired content for the creation of a new dataset. For a binary classification task, general guidelines suggest between 100 and 5,000 labelled samples for fine-tuning (Pecher et al., 2024), while multi-class tasks need fewer samples (around 100). However, the complex nature of gender bias often requires a larger dataset for robust detection since the number of samples depends mainly on the task type. The final dataset is expected to contain between 2,000 and 5,000 samples, balancing sufficient training data with resource constraints.

Existing EN-DE datasets were reviewed to reduce the need for manual data creation. The following sources were considered: `mGeNTE en-de` (Savoldi, Cupin, et al., 2025), `Building Bridges Dictionary` (Lardelli et al., 2024), and `Translated Wikipedia Biographies` (Stella et al., 2021). They were chosen because they all provide English source sentences, German translations, and sentence-level annotations that can be interpreted as biased or neutral. Analysis of the `Translated Wikipedia Biographies` dataset, however, revealed several issues that prevented direct reuse. In many instances, the `perceivedGender` column contained subject names instead of expected labels such as `Male`, `Female`, or `Neutral`, making manual verification necessary. Additionally, all examples were labelled as neutral (0), as the dataset was designed around correctly gendered references. Since the remaining two datasets were already balanced and contained a sufficient number of neutrally gendered examples, the Wikipedia Biographies dataset was excluded from the final training data. `mGeNTE` contains naturally occurring sentences with gendered entities, while `Building Bridges` focuses on German GFL entries for explicitly gendered nouns such as professions. A brief overview of the two remaining datasets is shown in Table 5.

Dataset	**Description**	**Content**
`mGeNTE en-de` (Savoldi, Cupin, et al., 2025)	Multilingual dataset to assess gender bias in MT.	~1,500 gender-ambiguous and gendered English sentences with gender-neutral and gendered German translations.
`Building Bridges Dictionary` (Lardelli et al., 2024)	Bilingual dictionary designed to support gender-fair EN-DE translation.	~230 German gender-neutral and gender-inclusive singular and plural sentences with English equivalents.

Table 5.: Overview of suitable EN-DE datasets based on past works

4.1.1. Pre-processing

mGeNTE en-de

The mGeNTE dataset contained the following relevant information:

- `SET-G`: English sentences with a clearly gendered subject.
- `SET-N`: English sentences with neutral or ambiguous subject gender.
- `REF-G`: German translations that preserve or introduce gender.
- `REF-N`: German translations that are fully gender–neutral.

The bias definition used in this study classifies translations that omit the original gender as neutral, as they do not rely on a male default or stereotype. Although gender-neutral translations may be imperfect, they are not considered biased within this framework. Initial experiments indicated that including `REF-N` pairs during training led to over-penalization of neutral outputs. Due to the limited availability of neutral examples, such outputs were not penalized in the final training setup. Each original entry was split into two paired examples and labelled as follows:

$$\texttt{SET-G} + \texttt{REF-G} \rightarrow 0 \quad \text{(neutral)}$$
$$\texttt{SET-G} + \texttt{REF-N} \rightarrow 0 \quad \text{(neutral)}$$
$$\texttt{SET-N} + \texttt{REF-N} \rightarrow 0 \quad \text{(neutral)}$$
$$\texttt{SET-N} + \texttt{REF-G} \rightarrow 1 \quad \text{(biased)}$$

This procedure yields 3,000 total instances, of which 750 are labelled biased (1) and 2,250 are labelled neutral (0).[1]

Building Bridges Dictionary

This dataset consisted of a GFL dictionary of nouns, not full sentences. That made it useful for studying GFL, but not suitable for this task, which requires sentence-level context. To address this, prompt engineering was used with Google Gemini 2.5 Flash to synthetically expand the dataset. [2] Nouns from the original dataset were used to create multiple grammatically correct sentence variations, covering singular, plural, gender-neutral, and gender-inclusive forms. The dataset uses the star form (e.g., *Lehrer*innen*) as its inclusive format. Since the colon form (e.g., *Lehrer:innen*) is also common in practice, a script was used to duplicate all entries with stars and replace the star with a colon to generate additional variants. This resulted in 3,381 total entries: 2,001 labelled as 0 (neutral) and 1,380 labelled as 1 (biased).[3]

[1] The transformed dataset can be found in `/datasets/mgente_final.csv`.
[2] Refer to Appendix C.2 for the prompt.
[3] The transformed dataset can be found in `/datasets/lardelli_final.csv`.

Tatoeba

The aforementioned setup lacked genuinely neutral examples, defined as sentences without any gendered subject, such as "The weather is nice" or "How are you". Including such cases is important for training the model to recognise that not all translations are relevant for gender bias detection, and that many sentences should be classified as neutral. As no suitable dataset for this category was available, a supplementary set was created from random EN–DE sentence pairs drawn from the Tatoeba corpus. A total of 550 sentence pairs was sampled. Manual filtering was applied to these samples to remove any pairs containing incorrect translations or stereotypical gendering, as public contributions often default to male forms. The final subset contained 532 clearly neutral sentence pairs, all labelled with 0.[4]

Available Data Summary

Table 6 shows an overview of the labelled data from the three available sources.

Dataset	Total	Neutral (0)	Biased (1)
`Building Bridges Dictionary`	3381	2001	1380
`mGeNTE`	3000	2250	750
`Tatoeba`	532	532	0

Table 6.: Summary of available labelled examples

The number of samples selected from each dataset was determined through iterative testing. Multiple dataset variants were created by upsampling or downsampling specific groups. The first model runs were conducted using a baseline dataset consisting of 750 biased and 750 unbiased samples from `mGeNTE`, 750 biased and 750 unbiased samples from the `Building Bridges Dictionary`, and 0 biased and 250 unbiased samples from `Tatoeba`. The documentation of this process is discussed in subsection 4.3.2.

4.1.2. Data Splitting and Cleaning

The dataset is partitioned into training (80%), validation (10%), and test (10%) subsets. This splitting ratio follows established practices commonly employed in ML experiments (Baheti, 2021). It provides enough samples for the model to learn general patterns while reserving separate subsets for tuning and final evaluation. Stratified sampling was used to maintain consistent label distribution (biased vs. neutral) across all three sets. For example, if 30% of the full dataset is biased, each split will also have 30% biased samples.

[4]The transformed dataset can be found in `/datasets/tatoeba_final.csv`.

Advanced text cleaning steps (punctuation removal, lowercasing, or stemming) were not applied due to the use of `bert-base-multilingual-cased`. This tokenizer handles raw, unaltered text and retains case distinctions. The model was pretrained on large corpora containing natural language in its original form (Devlin et al., 2019), so modifying the input by lowercasing or stripping punctuation could remove meaningful patterns the model has learned to recognize. Steps to handle missing values or invalid entries were already performed in the individual datasets, so they did not need to be repeated when creating the final merged dataset.

4.2. Training Pipeline

Following the fine-tuning process of `BERT` described earlier, `mBERT` with a binary classification head is used to predict whether a translation is *biased* or *neutral*. The tokenizer encodes input sentence pairs into numerical representations and distinguishes between source and target sentences. All sequences are padded or truncated to a fixed length of 256 tokens, which preserves most content while keeping processing efficient. The model represents each input pair with a summary vector of the entire sequence, which is then used for classification into the two categories. Each dataset is instantiated and encoded into a format suitable for model input, including the EN–DE sentence pairs and their corresponding labels. Training hyperparameters are established through tuning. The model iteratively learns from the training data by adjusting its parameters to minimize classification errors, with validation performance guiding the process and helping prevent overfitting. At the end of training, the best-performing model is selected based on validation metrics and used for subsequent gender bias detection.

4.3. Evaluation Strategy

Model evaluation was conducted using the validation set during training. As detailed in subsection 2.2.5, the macro F1 score was employed as the primary metric to assess model performance. The validation set served to monitor training progress across epochs, and the checkpoint with the highest validation F1 score was saved. The combined training dataset was handcrafted and had known limitations, so relying solely on the validation set was insufficient to assess final model performance. To better evaluate generalization, a separate handcrafted test set was created. This set contains EN-DE sentence pairs with manually assigned bias labels. Using these two evaluation strategies, both the fine-tuning process and the composition of the combined training dataset were iteratively adjusted to improve model robustness and generalization.

4.3.1. Handcrafted Test Set Construction

The handcrafted test set was developed from a user-centered perspective, focusing on identifying inputs that expose various failure and edge cases. Examples were organized into categories: neu-

tral sentences, neutral sentences containing gendered roles, biased translations, and translations featuring German GFL. It comprises simple synthetic sentences written specifically for this purpose, as well as authentic examples extracted from job postings. The inclusion of real-world data aims to simulate practical use cases, such as evaluating translated job advertisements for gender bias.

Emphasis was placed on diversity in sentence structure and content rather than maintaining label balance. Certain examples tested the model's tendency to incorrectly flag neutral sentences containing gendered terms, while others assessed its capacity to detect various GFL forms in German, including terms like "Lehrende" and the colon notation "Lehrer:innen." Bias labels were assigned manually according to the criteria established in Chapter 2. The complete handcrafted test set, containing 26 labelled translation pairs, is provided in Appendix B.1.

4.3.2. Hyperparameter Selection and Tuning

While a few standard hyperparameters were tested, the focus was placed on tuning dataset composition and the number of frozen layers. These factors showed a significantly stronger influence on model performance during experimentation. Since the training data originated from a mix of external sources with varying quality, adjusting the use and structure of the data was considered more effective than extensive hyperparameter optimization. Recommended default values from prior work provided sufficiently strong baselines and were therefore used as the starting point.

Epochs The model was trained for a maximum of 8 epochs, with early stopping enabled using a patience of two epochs. This setup halted training if the macro F1 score did not improve over two consecutive epochs. The approach follows the recommendation by Pecher et al. (2024), who suggest training until convergence, with a cap of 10 epochs and early stopping. In this case, validation loss typically increased after 8 epochs, with no further improvements observed. Limiting the training to 8 epochs helped mitigate overfitting and reduced training time.

Batch size A batch size of 16 was used. This value is commonly applied in fine-tuning scenarios involving small datasets, offering a reasonable balance between memory efficiency and training stability. Smaller batch sizes paired with lower learning rates were tested but led to reduced performance and less effective learning in early epochs. Existing literature, including Mosbach et al. (2021), supports the use of a batch size of 16; no further experiments with smaller values were conducted.

Learning rate The learning rate was set to 2e-5. This value, originally proposed in Devlin et al. (2019), remains widely used for fine-tuning transformer models. Alternatives such as 1e-5

and 3e-5 were evaluated but yielded slightly lower validation scores. The 2e-5 setting showed the most stable and consistent results and was therefore applied in all final training runs.

Optimizer and scheduler The `Trainer` API employed the AdamW optimizer by default. A warmup-linear learning rate schedule was used: the rate gradually increased during the first 10% of training steps (warmup) and then decreased linearly until completion. This schedule supports smooth learning and helps prevent instability during early training.

4.3.3. Training Dataset Tuning

Since the F1 scores were similar across dataset versions, the main evaluation was based on the handcrafted test set of 26 sentences.[5] This small test set does not provide a complete indication of overall model quality, but it offers insight into practical usability. Any statements regarding better or worse performance should be considered in light of this limitation.

The dataset `mgente_final` was considered the best source because its samples are natural sentences. All 750 biased samples from `mgente_final` were included, along with exactly 750 neutral samples. The tuning process aimed to adjust the remaining datasets to maintain a maximum ratio of 60% neutral to 40% biased samples. Sampling was performed using the `join_datasets.py` script, which loads the labelled datasets, samples a fixed number of biased and neutral entries per dataset using a fixed random seed (10), and combines them into a single training set. The script also checks for missing values and label integrity before saving the final CSV file. The tuning process across dataset versions, including their composition and rationale, is summarized in Table 7.

The Baseline dataset already achieved strong performance, with a test F1 score of 0.975 and 84.6% accuracy on the handcrafted test set. However, it failed on some neutral examples such as *"My mother is an engineer." / "Meine Mutter ist Ingenieurin."* (predicted biased with confidence 0.55) and on certain German GFL patterns (e.g., double naming and the colon notation). Adjustments made to the dataset composition in Datasets B through E occasionally improved specific weaknesses. In one case, Dataset E succeeded in correctly classifying a neutral gendered sentence that the Baseline had misclassified. At the same time, these targeted improvements often introduced new issues, such as misclassifications in job advertisement examples. The changes did not produce consistent gains on the handcrafted test set and in some cases reduced overall accuracy. As a result, the Baseline dataset composition was used for final training, as it offered the most reliable balance between targeted performance and general usability.

[5] The detailed documentation of each iteration is included in Appendix B.2. The focus in this section is on the process and rationale rather than on numeric results.

Dataset	Rationale	Sample Distribution (biased/neutral)
A	Initial setup using equal parts of `mgente_final` and `lardelli_final`, with some `tatoeba_final` neutrals.	mgente 750 / 750, lardelli 750 / 750, tatoeba 0 / 250
B	Built on A. Increased `lardelli_final` neutrals to better capture GFL patterns and added more `tatoeba_final` neutrals.	mgente 750 / 750, lardelli 750 / 1000, tatoeba 0 / 400
C	Built on A and B. Reduced `lardelli_final` biased examples to counter possible overrepresentation.	mgente 750 / 750, lardelli 400 / 750, tatoeba 0 / 250
D	Built on A and C. Further improved neutral recognition by adding more `tatoeba_final` neutral sentences.	mgente 750 / 750, lardelli 750 / 750, tatoeba 0 / 500
E	Built on A and C. Increased `mgente_final` neutral data to raise diversity from naturalistic examples.	mgente 750 / 1,250, lardelli 750 / 750, tatoeba 0 / 250

Table 7.: Dataset iterations with rationale and composition. Each version builds on the Baseline and previous adjustments. Format: source biased / neutral

4.3.4. Layer Freezing Tuning

All dataset tuning experiments described above were conducted with layer freezing set to $n = 8$, meaning that encoder layers 0 through 7 of `mBERT` were frozen during training. As explained in subsection 2.2.5, earlier studies have shown that the middle layers are most semantically informative, while lower layers tend to capture syntactic information. Freezing up to layer 8 was chosen as a baseline to reduce training time while still allowing the model to adapt higher-level representations to the task.Since the results with $n = 8$ were already promising, only two further variations were tested: $n = 7$ and $n = 6$. These settings freeze fewer layers, meaning more of the network remains trainable. The purpose of these tests was to evaluate whether this added flexibility improved performance without overfitting.

Freezing fewer layers led to slightly higher F1 scores on the test set, but the model with $n = 8$ frozen layers achieved the best results on the handcrafted test sentences, which were designed to reflect real-world usability. Since the F1 differences were minor and freezing more layers results in a simpler and more efficient model, $n = 8$ was chosen as the final setting.

Frozen Layers	Test F1 (weighted)	Handcrafted Test Set Accuracy
$n = 6$ (layers 0–5 frozen)	0.981	0.808
$n = 7$ (layers 0–6 frozen)	0.979	0.808
$n = 8$ (layers 0–7 frozen)	0.966	0.846

Table 8.: Comparison of layer freezing settings

4.4. Demo Application Design

The demo makes the bias detection model accessible through a user-friendly application, which comprises three modules: the user interface, the bias detection model, and the translation component (see Figure 9). On launch, the application loads the trained model and its encoding mechanism into memory. Users can either input raw English text (Tab 1), which is split into sentences and translated into German, or provide aligned EN-DE sentence pairs (Tab 2). Both approaches result in a set of sentence pairs ready for bias analysis. The bias detection pipeline processes these sentence pairs by converting each pair into a representation suitable for the model, which in turn predicts whether a translation is biased or neutral. Predictions are accompanied by a confidence score, which is the probability assigned by `mBERT` to the predicted label. Results are displayed alongside the original and translated sentences. If bias is detected above a defined threshold, a warning is shown; otherwise, the sentence pair is marked as neutral. Each result is separated to maintain readability.

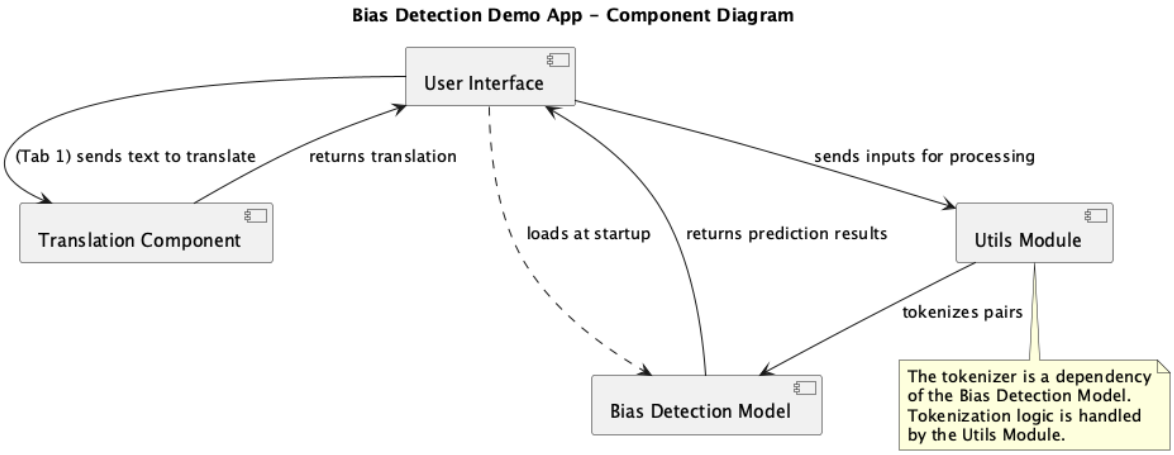

Figure 9.: Component Diagram. High-level architecture of the bias detection demo app showing components, their relationships, and data flow

The two tabs are further broken down in Figures 10 and 11. In automatic translation, raw English text is submitted, segmented into sentences, and translated. Each sentence is paired

with its translation before analysis. In manual pairing, users provide EN-DE sentence pairs directly, which bypasses translation and proceeds to bias evaluation. The system is designed to demonstrate the end-to-end process of bias detection independently of implementation details. The methodology allows the translation component or model representation to be replaced with alternative approaches. The focus is on showing how input sentences are converted into pairs, analyzed for bias, and presented in a user-friendly way.

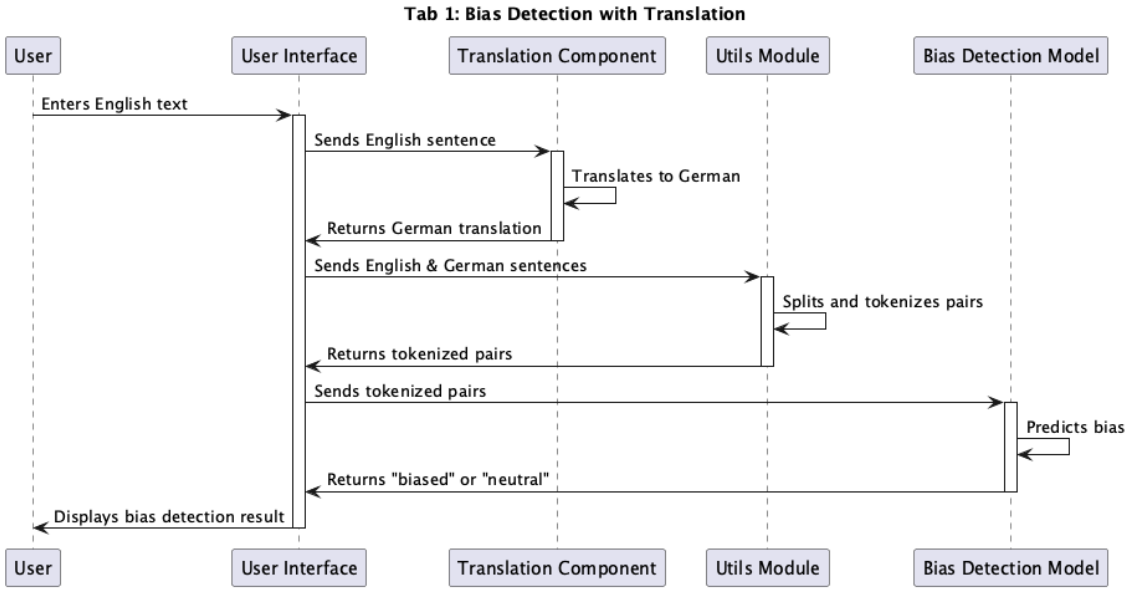

Figure 10.: Sequence Diagram – Tab 1 (with translation)

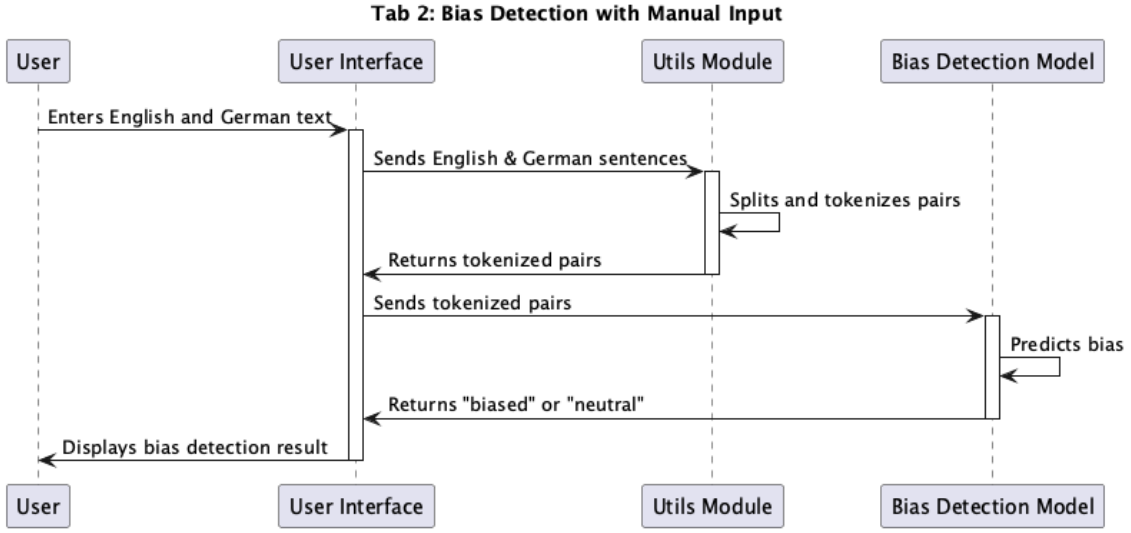

Figure 11.: Sequence Diagram – Tab 2 (manual input)

5. Implementation

Based on the methodology, this chapter presents the specific implementation carried out in this project, along with step-by-step instructions to run the demo and reproduce the results. The complete project repository, including all datasets and scripts, is available at https://github.com/phmkhali/bias-detector-en-de.

5.1. Environment Setup and Project Structure

5.1.1. System Environment and Hardware

The project was developed on macOS using Python 3.12.4. A virtual environment was used, and all dependencies were installed via the `requirements.txt` file using `pip3`. No manual installation steps were needed beyond this. During development, both GPU and CPU were used to test training and inference performance, with device selection handled dynamically. The final model for the demo and evaluation was trained on the CPU to ensure full compatibility and reproducibility without relying on a GPU. To ensure reproducibility, random seeds were fixed across all libraries and backends. The application is started via Streamlit. Further usage instructions are provided in section 5.3.

5.1.2. Directory Layout

Figure 12 shows the directory structure with the relevant files for the final implementation. The folder contains additional files related to the original datasets and scripts used for data conversion. These supplementary files are intended for comprehension and reproducibility purposes but are not required for the final model.

5.2. Core Components and Data Flow

5.2.1. Datasets Folder

The `datasets` folder contains the processed datasets used for training and evaluation of the bias detection model. It includes the final combined dataset `dataset.csv` as well as intermediate datasets derived from individual sources: `lardelli_final`, `mgente_final`, and `tatoeba_final`. The `join_datasets` script facilitates the concatenation of these datasets into a unified dataset. The script is designed to support iterative sampling by specifying sample sizes for biased and

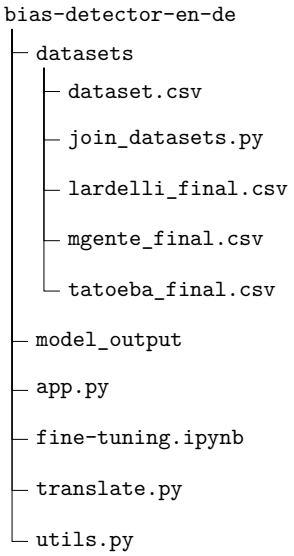

Figure 12.: Relevant files of the final implementation

neutral classes with a fixed random seed. It merges the sampled subsets, shuffles the combined data, and performs data integrity checks.

5.2.2. Fine-tuning Notebook and Model Output

The `fine-tuning.ipynb` notebook carries out the complete process of preparing and fine-tuning the bias detection model. It begins by loading the pretrained `mBERT` model and the training dataset (`dataset.csv`). Each dataset is instantiated using a custom `BiasDataset` class, which encodes EN-DE sentence pairs and their labels into tensors suitable for model input. Training configurations are defined via `TrainingArguments`, specifying evaluation intervals and checkpoint saving strategies. A `Trainer` object is then created with the model, datasets, training arguments, and a metric computation function.

The training proceeds by feeding batches from the training dataset to the model, updating the weights based on the loss, and evaluating performance on the validation set at the end of each epoch. Early stopping limited the process to six epochs, taking approximately 18 minutes and 32 seconds on the development machine. Checkpoints, configuration files, tokenizer files, and training metadata are saved in the `model_output` folder, with the best-performing model retained for subsequent use in bias detection.

5.2.3. Streamlit Application

The `app.py` file is the main entry point for the Streamlit interface.[1] It loads the fine-tuned model and tokenizer from the `model_output` directory and sets up the classification pipeline. In the

[1] Parts of the code were generated with AI and subsequently refined by the author; see Appendix C.3 for details.

automatic tab, users enter English text, which is translated using `translate.py`. In the manual tab, users enter both the original and translated sentences directly.[2] `translate.py` wraps the pre-trained OpusMT EN-DE model from Hugging Face. The main function, `translate_batch`, takes a list of English sentences, tokenizes them, and uses the model's `generate` method to return a list of decoded German translations. The resulting sentence pairs from both tabs are passed to functions in `utils.py`. This module handles sentence splitting, optional translation, and model inference. It processes each sentence pair, predicts whether it contains gender bias, and formats the output with confidence scores for display in the interface.

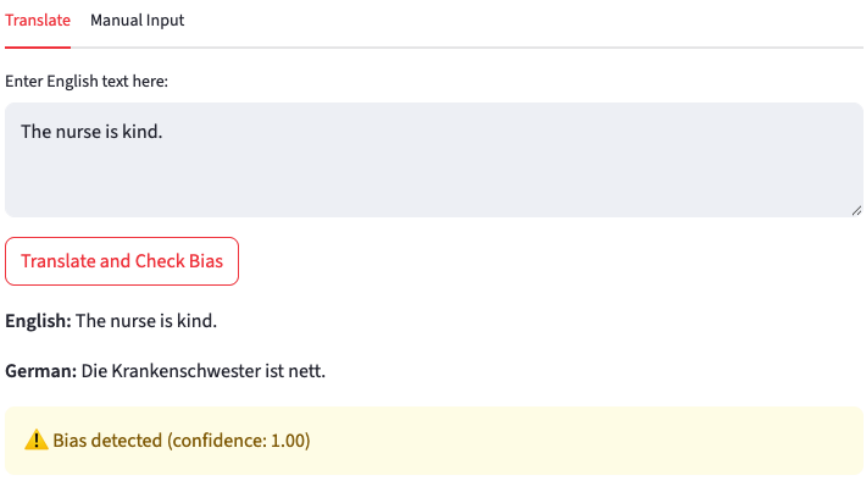

Figure 13.: Streamlit Demo: Automatic Translation Tab showing correct identification of bias in stereotypical occupational gender assignment

[2] Screenshots of these tabs are shown in Figures 13, 14, and 15.

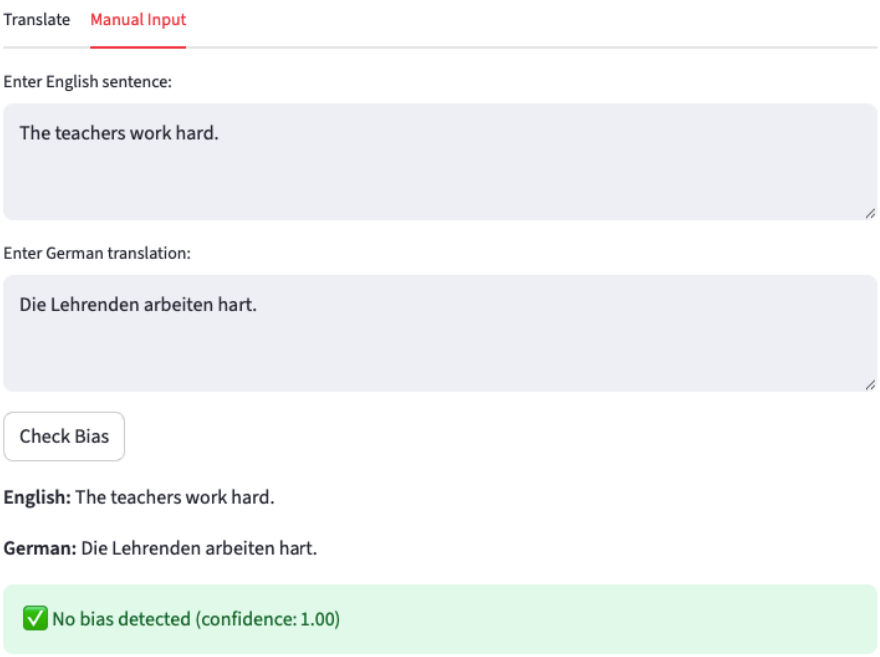

Figure 14.: Streamlit Demo: Manual Translation Tab showing no bias detected due to use of GFL

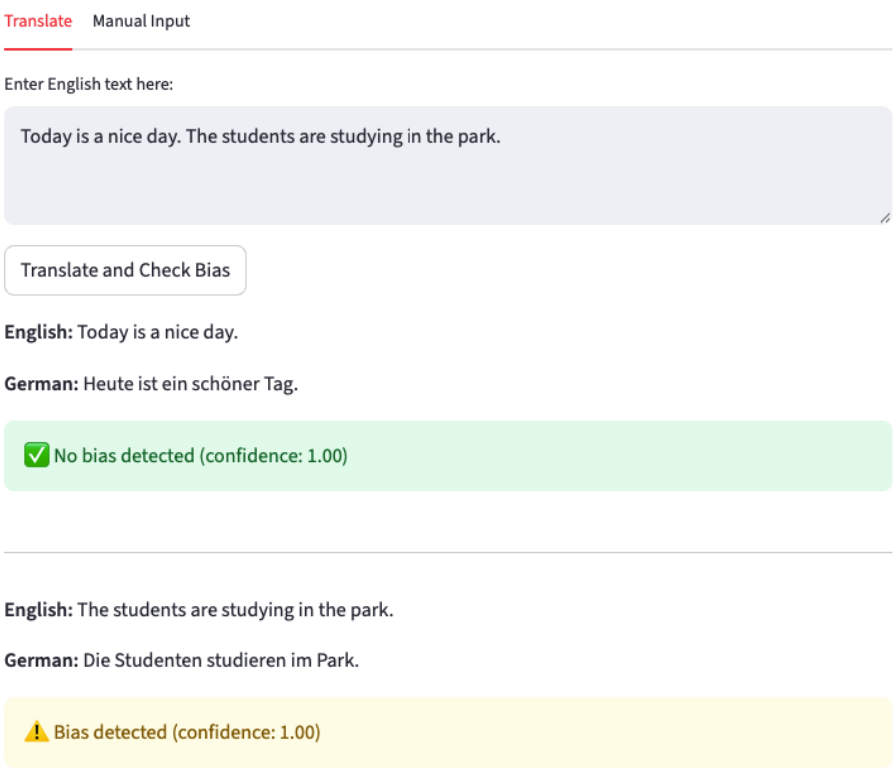

Figure 15.: Streamlit Demo correctly splitting and labeling two sentences as neutral and biased

5.3. Reproduction Guide

The setup process for the Streamlit demo app includes creating a Python virtual environment, installing required packages, and running the application. This guide covers these steps for macOS/Linux and Windows. **Note:** The pre-trained model is not included in the GitHub repository due to size restrictions. It can be downloaded separately via the provided Google Drive link.[3] If the pre-trained model is not present, the `fine-tuning.ipynb` notebook must be executed first to train and save the model before launching the demo.

Installation Steps

1. Open a terminal (macOS/Linux) or PowerShell (Windows).
2. Clone the GitHub repository and download the pre-trained model:

[3] Google Drive link for the model.

```
git clone https://github.com/phmkhali/bias-detector-en-de
cd bias-detector-en-de
```

Download the model

```
# Manually download from the provided Google Drive link above
# and place the model_output folder into the directory
```

3. Create and activate a Python virtual environment:

   macOS / Linux

   ```
   python3 -m venv venv
   source venv/bin/activate
   ```

   Windows

   ```
   python -m venv venv
   .\venv\Scripts\activate
   ```

4. Install the required packages:

   macOS / Linux

   ```
   pip3 install -r requirements.txt
   ```

   Windows

   ```
   pip install -r requirements.txt
   ```

5. (Only necessary if the `model_output` directory was not downloaded and added to the repository) Run the `fine-tuning.ipynb` notebook manually to generate the model.

6. Run the Streamlit app:

   macOS / Linux

   ```
   python3 -m streamlit run app.py
   ```

   Windows

   ```
   streamlit run app.py
   ```

6. Evaluation and Findings

This chapter focuses on the interpretation of performance metrics and typical error patterns by analysing typical error patterns in the test sets and conducting exploratory tests to validate initial assumptions.

6.1. Model Performance

The model's overall accuracy on the held-out test set is 0.966, with matching macro F1 and weighted average. This shows it performs evenly across both classes. Precision and recall reveal biased cases are detected very well (recall 0.993) with some false predictions (precision 0.937), meaning a small portion of neutral cases are incorrectly labelled as biased. The confusion matrix confirms this with 10 false positives and only 1 false negative in 325 examples. The false positive rate is 0.057 and the false negative rate is 0.007. Overall, this suggests the model favors detecting bias at the risk of some over-flagging, which fits typical needs for bias detection where missing bias is costlier than false alarms.

Class	**Precision**	**Recall**	**F1 Score**
Neutral (0)	0.994	0.943	0.968
Biased (1)	0.937	0.993	0.964

Table 9.: Per-class precision, recall, and F1 score on the test set

Metric	**Value**
Accuracy	0.966
Macro-average F1 Score	0.966
Weighted-average F1 Score	0.966
False Positive Rate	0.057
False Negative Rate	0.007

Table 10.: Overall evaluation metrics on the test set

False positives and false negatives make up only a small share of the total predictions, but

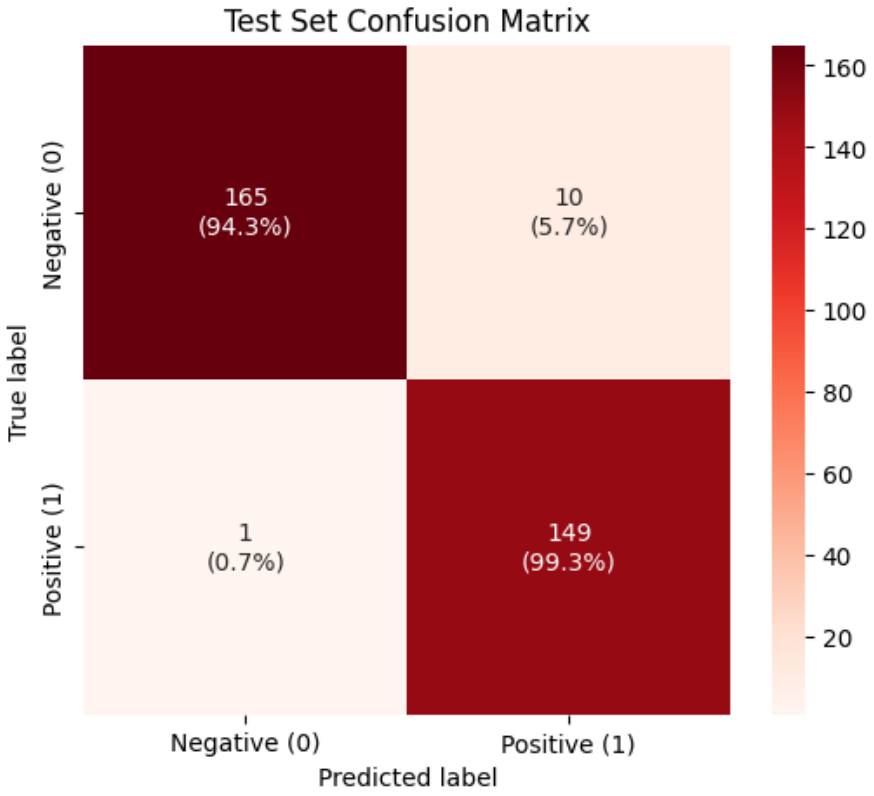

Figure 16.: Confusion matrix of the model on the test dataset, showing true vs. predicted labels with counts

they point to recurring patterns in the model's errors. These cases were grouped by shared features to enable a structured analysis of likely causes.[1]

State and Governance Entity Terms Sentences containing political, legal, or governance-related terminology were misclassified the most (6/10 cases). Terms such as president, police officers, heads of state and government, and political leaders were flagged as biased, despite the absence of gendered language in the German translation. The model may struggle to distinguish between genuinely biased constructions and content related to institutional or geopolitical domains. Another possible explanation is the strong male association of such terms in real-world data, which may influence model behavior through learned co-occurrence patterns (Kroeber and Hüffelmann, 2022).

Training Dataset Error Causing a False Positive One false positive appears to stem from an inconsistency within the training dataset. The English sentence "[...] or would you go to a surgeon?" was translated as "[...] oder von einem Menschen, der als Chirurg ausgebildet

[1] Refer to the appendix for all misclassified examples, including English source texts and German translations.

wurde [...]?". This example originates from the mGeNTE dataset and is labelled as unbiased. The German translation attempts to avoid using "der Chirurg" (m.) by phrasing it as "a person trained as a surgeon," presumably to circumvent the generic masculine. However, the word "Chirurg" still appears in the translation in its masculine form. The model correctly identified this gendered term and flagged it, despite the label indicating neutrality. From a bilingual perspective, the model's detection is a true positive and the false flagging points to a labeling inconsistency in the dataset.

Religious identities One false negative involved a sentence concerning the immigration of muslims. "Muslims" was translated using the generic masculine form "Muslime", yet the model failed to recognize this instance as biased. Since the sentence does not provide a clear reason for why it was misclassified, this issue will be further analysed using exploratory testing (section 6.3).

Issues with GFL and Semantically Gendered Terms Two instances were incorrectly flagged as biased due to the use of GFL. The gender-ambiguous terms "specialists" and "recipient" were translated as "Sachkundigen" and "Rezipierende," respectively, both using neutral rewording. The false detection here reflects insufficient understanding of GFL forms.

Additionally, one semantically gendered term, "uncle," translated as "Onkel," was flagged, pointing towards difficulty in distinguishing inherently gendered terms from translation-induced gender bias. This assumption will be tested for consistency in section 6.2.

6.2. Generalization performance on unseen data

To assess how well these findings generalize, the results from the handcrafted test set are analysed. As explained in subsection 4.3.1, it was curated to expose potential failure modes and edge cases not captured by standard evaluation. The model reached 84.6% accuracy on this set, which guided model selection. This section examines its strengths and weaknesses in detail, focusing on how individual cases were classified.[2]

6.2.1. Weaknesses

Table 11 shows all of its false predictions. Consistent with the results from the held-out test set, these errors involve semantic gender distinctions and GFL variants. It is important to note that the test set did include more semantic gender examples, such as "sister," "boy," and "father" in ambiguous contexts, which the model did not flag. Sentence 5 with "mother" was flagged, but with low confidence. It can be concluded that this limitation stems from insufficient training data for semantically gendered words. The model recognizes a pattern

[2]Refer to the appendix for the handcrafted test results and confidence scores.

but lacks enough instances to confidently classify. The remaining three cases share the same sentence structure but use different GFL forms. The model flagged some sentences using the pair form and gender-inclusive characters (the colon) as biased, with over 95% confidence. It did however correctly identify the user of the gender star and neutral rewording of "teacher" to "Lehrende" as unbiased, also with a confidence of over 95%.

This reflects the known limitation of the dataset, where the limited availability of high-quality natural GFL data required supplementing it with lower-quality synthetic examples. Still, the correct classification in some cases suggests that the model may be capable of handling GFL more reliably if more representative data were available.

Index	**English**	**German**	Neutral Prob %	Biased Prob %
5	My mother is an engineer.	Meine Mutter ist Ingenieurin.	44.1	55.9
16	Engineers are kind and hard-working	Ingenieure und Ingenieurinnen sind lieb und fleißig	0.08	99.92
18	Teachers are kind and hard-working	Lehrer:innen sind lieb und fleißig	3.1	96.9
19	Teachers are kind and hard-working	Lehrerinnen und Lehrer sind lieb und fleißig	0.05	99.95

Table 11.: Handcrafted test sentences with incorrect model predictions and confidence scores (percent)

6.2.2. Strengths

The performance metrics earlier already imply that the model performs strongly in detecting genuinely biased cases. This is supported by the handcrafted test set: no clearly biased sentence was missed, and out of these, not a single one received a confidence score below 90%. The model also demonstrates the ability to distinguish between clearly gendered and neutral content. Generic sentences such as “How are you?”, translated as “Wie geht es dir?”, were correctly classified as neutral. Likewise, sentences taken from real job postings were consistently identified with high confidence (e.g., “The ideal candidate” → “Der ideale Kandidat” (m.) = biased). In total, all tested job posting examples reached confidence scores above 99%. This indicates that the model is well-suited for handling practical use cases, which supports its applicability in real-world scenarios.

6.3. Exploratory Testing

To further investigate the unresolved misclassification of the religious identity sentence, the demo was used to test additional cases. The term "Muslims" was replaced with other religious

terms, followed by "soldiers" to represent a non-religious but contextually relevant group, and concluded with "doctors," a typical example of a generic masculine translation.

Sentence Under Investigation:

> **English:** Here too the local people are frustrated by the immigration of Muslims and the hard line taken by the military.
>
> **German:** Hier wird die lokale Bevölkerung ebenfalls durch die Zuwanderung von Muslimen und das unnachsichtige Auftreten des Militärs schwer gebeutelt.

Replacement Term	**Bias Flag**
Christians	No bias detected (confidence: 0.89)
Jews	No bias detected (confidence: 0.93)
Soldiers	No bias detected (confidence: 0.79)
Doctors	No bias detected (confidence: 0.64)

Table 12.: Bias detection results for various replacement terms, showing confidence scores and absence of bias flags.

The consistent failure to detect bias across these terms, all translated using the generic masculine, suggests the issue may stem from the grammatical structure of the sentence instead of the religious context. Testing formal features like punctuation and casing showed a clearer impact on the confidence. Removing the period dropped confidence to 0.84. Lowercasing "Muslims" lowered it to 0.83. Doing both together caused the confidence to fall to 0.58. Because the model is based on cased BERT, it relies on correct punctuation and capitalization for context. The limited training data increases the model's sensitivity to such formal changes, causing performance to vary when cues like casing and punctuation are missing or altered. To confirm this effect, the insertion tests were repeated with both punctuation and casing removed. In these modified cases, "christians" was flagged as biased (confidence: 0.98), "soldiers" was not flagged (confidence: 0.63), and "doctors" was again flagged (confidence: 0.76).

Replacement Term	Original	No Punctuation, Lowercase
Christians	No bias (confidence: 0.89)	Bias (confidence: 0.98)
Jews	No bias (confidence: 0.93)	No bias (confidence: 0.51)
Soldiers	No bias (confidence: 0.79)	No bias (confidence: 0.63)
Doctors	No bias (confidence: 0.64)	Bias (confidence: 0.76)

Table 13.: Bias detection results for various replacement terms, comparing original inputs with versions lacking punctuation and casing

One final hypothesis was that the term "local people" may influence the classification. In the original translation, it was rendered as "lokale Bevölkerung"; a neutral term that may give the impression of a gender-neutral subject. This could have caused the model to focus less on the gendered translation of "Muslims". To explore this, the term "local" was removed from the original sentence, changing the translation to "Auch hier sind die Menschen durch die Einwanderung von Muslimen und die harte Linie des Militärs frustriert."[3] The model correctly flagged this version as biased (confidence: 0.97). The presence of both a neutral and a gendered subject within the same sentence should be considered a factor that may reduce the model's classification accuracy. At the same time, testing edge cases is inherently challenging due to the interaction of multiple subtle influences. While stepwise sentence modifications can help identify possible sources of misclassification, definitive conclusions remain difficult to establish. To maintain the practicality of the app, a disclaimer was added to inform users about its limitations and potential weaknesses.

In this chapter, the model was evaluated in view of the research question. Its performance was analyzed on both a held-out dataset and a handcrafted dataset of unseen examples. Real-world texts, such as job postings, were used to test how the model performs in practice. Common error patterns were identified and grouped to better understand factors that influence misclassification, providing a basis for the conclusions in the next chapter.

[3]This translation slightly differs in wording because it was generated using OpusMT. The version in the test set was likely manually translated by the mGeNTE creators.

7. Conclusion and Discussion

Without a doubt, MT systems and their accessibility drastically improve our ability to communicate with one another. While modern NMT provides semantically accurate translations for high-resource languages, they often introduce gender bias when translating between languages with and without grammatical gender. This thesis attempted to create an application to detect gender bias in EN-DE translations in real time. The findings show that an mBERT model fine-tuned for this binary classification task can detect gender bias with an accuracy of 96.6% on a held-out test set, which comes from a train–test split of a combined dataset built from existing studies. On a separate handcrafted dataset designed to test edge cases and practical scenarios, the model reaches 84.6% accuracy. It shows strong performance in identifying generic masculines and the assignment of stereotypical roles, which make up the majority of gender biases found in MT systems (Lardelli et al., 2024; Stanovsky et al., 2019; Prates et al., 2019). Despite its high accuracy, there are recurring error patterns that affect the model's reliability. The error analysis revealed five sources of occasional incorrect predictions: (1) misclassifying German GFL forms as biased, (2) failing to detect semantically gendered words as unbiased, (3) failing to detect bias in political and government terms, (4) showing sensitivity to punctuation and capitalization, as well as (5) difficulty handling sentences that contain both neutral and gendered subjects.

Taking everything into account, detecting gender bias in MT remains a challenging task. In morphologically rich languages like German, the subtle linguistic patterns are particularly difficult for a model to learn. Nonetheless, the model's strong performance in practical scenarios, the recognition of generic masculines and stereotypical role assignments in job postings, demonstrates its efficacy in flagging gender bias in translations. This establishes the model as a crucial intermediary layer: it raises awareness by preventing bias from going unnoticed and contributes to mitigating the representational harm that biased translations can reinforce.

7.1. Limitations of this work

This work is primarily limited by its design choices and the resources that were available. The bias detector analyzes sentences in isolation, which allows users to quickly see the flagged sentences. This design simplifies processing but also means that bias depending on broader context can be missed. For instance, if a text introduces "the doctor is Mr. Smith" and later uses a male form for "the doctor," this is not actually biased, but the system cannot account

for it. Detecting such cases would require word-level analysis and annotated data, which were not available. Data quality and quantity also restricted performance. High-quality datasets were small, so synthetic data and a handcrafted evaluation set were used, which introduces the risk that subjective choices propagate into the model.

The evaluation itself was limited as well. With only 26 test sentences, the results provide a rough indication of the model's strengths and weaknesses but do not capture the full range of possible cases. Similarly, the system assumes that translations, whether generated by OpusMT or entered manually, are correct. Mistakes in translation can therefore cause misleading predictions, since translation quality was not taken into account in this work. These constraints are compounded by the model's sensitivity to sentence-level features. Classification can be influenced by capitalization, punctuation, or multiple subjects in a sentence. Moreover, the lack of interpretability measures prevents the model from explaining why it flagged a sentence as biased, which limits understanding of its predictions and potential corrective actions. Overall, these choices result in a sentence-level tool capable of detecting explicit gender bias, but they limit the system's ability to handle context, translation errors, and more subtle linguistic phenomena.

7.2. Outlook

To address these limitations, adding more samples and incorporating natural, word-level labelled data would give the model more diverse examples, allowing it to better capture different patterns of gender bias and handle context-dependent cases. Expanding coverage to additional domains, such as adjective-based biases, would make the system more robust across different types of texts. Moreover, moving beyond binary classification toward token-level analysis could enable the model to identify specific words or phrases responsible for bias and classify them more precisely, for instance as "fair" or "generic masculine." These improvements would not only enhance the accuracy of bias detection but also provide users with clearer guidance on how translations may perpetuate gendered assumptions.

The interdisciplinary nature of this task means that progress in ML must be accompanied by advances in social and linguistic research. Understanding these dimensions of GFL helps create more reliable datasets and better ways to measure it. At the same time, exploring alternative model architectures, such as Large Language Models (LLMs) with prompt engineering, could reveal new approaches to bias detection. Ultimately, the long-term goal is for MT systems to handle gendered languages more accurately on their own, reducing the need for intermediary tools. Until that point, applications like the one presented here make bias visible and support more equitable translation practices.

Bibliography

Baheti, P. (2021). *Train Test Validation Split: How To & Best Practices [2024]*. URL: https://www.v7labs.com/blog/train-validation-test-set (visited on 07/30/2025).

Baldi, P. (July 2008). "English as an Indo-European Language". In: *A Companion to the History of the English Language*. Ed. by H. Momma and M. Matto. 1st ed. Wiley, pp. 127–141. ISBN: 978-1-4051-2992-3 978-1-4443-0285-1. DOI: 10.1002/9781444302851.ch12. URL: https://onlinelibrary.wiley.com/doi/10.1002/9781444302851.ch12 (visited on 06/06/2025).

Barclay, P. J. and A. Sami (Apr. 2024). *Investigating Markers and Drivers of Gender Bias in Machine Translations*. DOI: 10.48550/arXiv.2403.11896. arXiv: 2403.11896 [cs]. URL: http://arxiv.org/abs/2403.11896 (visited on 05/21/2025).

Chakravarthi, B. R. et al. (July 2021). "A Survey of Orthographic Information in Machine Translation". In: *SN Computer Science* 2.4, p. 330. ISSN: 2662-995X, 2661-8907. DOI: 10.1007/s42979-021-00723-4. URL: https://link.springer.com/10.1007/s42979-021-00723-4 (visited on 06/27/2025).

Cho, W. I. et al. (2019). *On Measuring Gender Bias in Translation of Gender-neutral Pronouns*. DOI: 10.48550/arXiv.1905.11684. arXiv: 1905.11684 [cs]. URL: http://arxiv.org/abs/1905.11684 (visited on 04/21/2025).

DeepL (Nov. 2021). *How Does DeepL Work?* URL: https://www.deepl.com/en/blog/how-does-deepl-work (visited on 06/27/2025).

Devlin, J. (2018). *Multilingual BERT GitHub Readme*. URL: https://github.com/google-research/bert/blob/a9ba4b8d7704c1ae18d1b28c56c0430d41407eb1/multilingual.md (visited on 07/25/2025).

Devlin, J. et al. (2019). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. DOI: 10.48550/arXiv.1810.04805. arXiv: 1810.04805 [cs]. URL: http://arxiv.org/abs/1810.04805 (visited on 04/09/2025).

Godsil, R. D. et al. (2016). "The Effects of Gender Roles, Implicit Bias, and Stereotype Threat on the Lives of Women and Girls". In: *THE SCIENCE OF EQUALITY* 2.Perception Institute. URL: https://equity.ucla.edu/wp-content/uploads/2016/11/Science-of-Equality-Volume-2.pdf.

Google (Dec. 2018). *Reducing Gender Bias in Google Translate*. URL: https://blog.google/products/translate/reducing-gender-bias-google-translate/ (visited on 06/05/2025).

Gurgurov, D., T. Bäumel, and T. Anikina (2024). "Multilingual Large Language Models and Curse of Multilinguality". In: DOI: 10.48550/arXiv.2406.10602. arXiv: 2406.10602 [cs]. URL: http://arxiv.org/abs/2406.10602 (visited on 07/27/2025).

Kappl, M. (2025). *Are All Spanish Doctors Male? Evaluating Gender Bias in German Machine Translation*. DOI: 10.48550/arXiv.2502.19104. arXiv: 2502.19104 [cs]. URL: http://arxiv.org/abs/2502.19104 (visited on 04/10/2025).

Kroeber, C. and J. Hüffelmann (Sept. 2022). "It's a Long Way to the Top: Women's Ministerial Career Paths". In: *Politics & Gender* 18.3, pp. 741–767. ISSN: 1743-923X, 1743-9248. DOI: 10.1017/S1743923X21000118. URL: https://www.cambridge.org/core/product/identifier/S1743923X21000118/type/journal_article (visited on 08/06/2025).

Lardelli, M., G. Attanasio, and A. Lauscher (2024). "Building Bridges: A Dataset for Evaluating Gender-Fair Machine Translation into German". In: *Findings of the Association for Computational Linguistics ACL 2024*. Bangkok, Thailand and virtual meeting: Association for Computational Linguistics, pp. 7542–7550. DOI: 10.18653/v1/2024.findings-acl.448. URL: https://aclanthology.org/2024.findings-acl.448 (visited on 04/06/2025).

Lauscher, A. et al. (Nov. 2020). "From Zero to Hero: On the Limitations of Zero-Shot Language Transfer with Multilingual Transformers". In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Ed. by B. Webber et al. Online: Association for Computational Linguistics, pp. 4483–4499. DOI: 10.18653/v1/2020.emnlp-main.363. URL: https://aclanthology.org/2020.emnlp-main.363/ (visited on 07/27/2025).

Libovický, J., R. Rosa, and A. Fraser (Nov. 2019). *How Language-Neutral Is Multilingual BERT?* DOI: 10.48550/arXiv.1911.03310. arXiv: 1911.03310 [cs]. URL: http://arxiv.org/abs/1911.03310 (visited on 07/14/2025).

Lin, G. H.-c. and P. S. C. Chien (2009). "Machine Translation for Academic Purposes". In: *Proceedings of the International Conference on TESOL and Translation 2009*, pp.133–148.

Mosbach, M., M. Andriushchenko, and D. Klakow (Mar. 2021). *On the Stability of Fine-tuning BERT: Misconceptions, Explanations, and Strong Baselines*. DOI: 10.48550/arXiv.2006.04884. arXiv: 2006.04884 [cs]. URL: http://arxiv.org/abs/2006.04884 (visited on 07/14/2025).

Nadipalli, S. (Feb. 2025). *Layer-Wise Evolution of Representations in Fine-Tuned Transformers: Insights from Sparse AutoEncoders*. DOI: 10.48550/arXiv.2502.16722. arXiv: 2502.16722 [cs]. URL: http://arxiv.org/abs/2502.16722 (visited on 07/20/2025).

Pecher, B., I. Srba, and M. Bielikova (Apr. 2024). *Comparing Specialised Small and General Large Language Models on Text Classification: 100 Labelled Samples to Achieve Break-Even Performance*. DOI: 10.48550/arXiv.2402.12819. arXiv: 2402.12819 [cs]. URL: http://arxiv.org/abs/2402.12819 (visited on 04/27/2025).

Phuong, M. and M. Hutter (July 2022). *Formal Algorithms for Transformers*. DOI: 10.48550/arXiv.2207.09238. arXiv: 2207.09238 [cs]. URL: http://arxiv.org/abs/2207.09238 (visited on 07/04/2025).

Pires, T., E. Schlinger, and D. Garrette (June 2019). *How Multilingual Is Multilingual BERT?* DOI: 10.48550/arXiv.1906.01502. arXiv: 1906.01502 [cs]. URL: http://arxiv.org/abs/1906.01502 (visited on 07/14/2025).

Prates, M. O. R., P. H. C. Avelar, and L. Lamb (2019). *Assessing Gender Bias in Machine Translation – A Case Study with Google Translate*. DOI: 10.48550/arXiv.1809.02208. arXiv: 1809.02208 [cs]. URL: http://arxiv.org/abs/1809.02208 (visited on 04/03/2025).

Quemy, A. (Mar. 2019). *Binary Classification in Unstructured Space With Hypergraph Case-Based Reasoning*. DOI: 10.48550/arXiv.1806.06232. arXiv: 1806.06232 [cs]. URL: http://arxiv.org/abs/1806.06232 (visited on 07/04/2025).

Rainio, O., J. Teuho, and R. Klén (Mar. 2024). "Evaluation Metrics and Statistical Tests for Machine Learning". In: *Scientific Reports* 14.1. ISSN: 2045-2322. DOI: 10.1038/s41598-024-56706-x. URL: https://www.nature.com/articles/s41598-024-56706-x (visited on 07/20/2025).

Rescigno, A. A. and J. Monti (2023). "Gender Bias in Machine Translation: A Statistical Evaluation of Google Translate and DeepL for English, Italian and German". In: *Proceedings of the International Conference on Human-informed Translation and Interpreting Technology 2023*. UNIOR NLP Research Group, University of Naples "L'Orientale", Naples, Italy: INCOMA Ltd., Shoumen, Bulgaria, pp. 1–11. DOI: 10.26615/issn.2683-0078.2023_001. URL: https://acl-bg.org/proceedings/2023/HiT-IT%202023/pdf/2023.hitit2023-1.1.pdf (visited on 02/27/2025).

Savoldi, B., J. Bastings, et al. (May 2025). "A Decade of Gender Bias in Machine Translation". In: *Patterns*, p. 101257. ISSN: 26663899. DOI: 10.1016/j.patter.2025.101257. URL: https://linkinghub.elsevier.com/retrieve/pii/S2666389925001059 (visited on 06/06/2025).

Savoldi, B., E. Cupin, et al. (2025). *mGeNTE: A Multilingual Resource for Gender-Neutral Language and Translation*. DOI: 10.48550/arXiv.2501.09409. arXiv: 2501.09409 [cs]. URL: http://arxiv.org/abs/2501.09409 (visited on 04/08/2025).

Savoldi, B., S. Papi, et al. (2024). "What the Harm? Quantifying the Tangible Impact of Gender Bias in Machine Translation with a Human-centered Study". In: *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Miami, Florida, USA: Association for Computational Linguistics, pp. 18048–18076. DOI: 10.18653/v1/2024.emnlp-main.1002. URL: https://aclanthology.org/2024.emnlp-main.1002 (visited on 04/06/2025).

Schiebinger, L. (Mar. 2014). "Scientific Research Must Take Gender into Account". In: *Nature* 507.7490, pp. 9–9. ISSN: 1476-4687. DOI: 10.1038/507009a. URL: https://www.nature.com/articles/507009a (visited on 06/06/2025).

Schmitz, D. (Aug. 2022). *In German, All Professors Are Male*. DOI: 10.31234/osf.io/yjuhc. URL: https://osf.io/yjuhc (visited on 06/06/2025).

Schryen, G. (2015). "Writing Qualitative IS Literature Reviews—Guidelines for Synthesis, Interpretation, and Guidance of Research". In: *Communications of the Association for Information Systems* 37. ISSN: 15293181. DOI: 10.17705/1CAIS.03712. URL: https://aisel.aisnet.org/cais/vol37/iss1/12/ (visited on 05/09/2025).

Schwemmer, C. et al. (Jan. 2020). "Diagnosing Gender Bias in Image Recognition Systems". In: *Socius* 6, p. 2378023120967171. ISSN: 2378-0231. DOI: 10.1177/2378023120967171. URL: https://doi.org/10.1177/2378023120967171 (visited on 05/28/2025).

Sczesny, S., M. Formanowicz, and F. Moser (2016). "Can Gender-Fair Language Reduce Gender Stereotyping and Discrimination?" In: *Frontiers in Psychology* 7. ISSN: 1664-1078. DOI: 10.3389/fpsyg.2016.00025. URL: http://journal.frontiersin.org/Article/10.3389/fpsyg.2016.00025/abstract (visited on 05/16/2025).

Shrestha, S. and S. Das (2022). "Exploring Gender Biases in ML and AI Academic Research through Systematic Literature Review". In: *Frontiers in Artificial Intelligence* 5, p. 976838. ISSN: 2624-8212. DOI: 10.3389/frai.2022.976838. URL: https://www.frontiersin.org/articles/10.3389/frai.2022.976838/full (visited on 04/06/2025).

Sichler, A. and E. Prommer (2014). "Gender Differences within the German-language Wikipedia". In: *ESSACHESS - Journal for Communication Studies* 7.2, pp. 77–93. ISSN: 1775-352X.

SkyQuest (2025). *Machine Translation (MT) Market Size, Growth & Trends Report | 2032*. URL: https://www.skyquestt.com/report/machine-translation-market (visited on 05/23/2025).

Smacchia, M., S. Za, and A. Arenas (2024). "Does AI Reflect Human Behaviour? Exploring the Presence of Gender Bias in AI Translation Tools". In: *Digital (Eco) Systems and Societal Challenges*. Ed. by A. M. Braccini, F. Ricciardi, and F. Virili. Vol. 72. Cham: Springer Nature Switzerland, pp. 355–373. ISBN: 978-3-031-75585-9 978-3-031-75586-6. DOI: 10.1007/978-3-031-75586-6_19. URL: https://link.springer.com/10.1007/978-3-031-75586-6_19 (visited on 02/27/2025).

Smith, B. (May 2024). *A Complete Guide to BERT with Code*. URL: https://towardsdatascience.com/a-complete-guide-to-bert-with-code-9f87602e4a11/ (visited on 07/11/2025).

Sorrenti, A. et al. (Oct. 2023). "Selective Freezing for Efficient Continual Learning". In: *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*. Paris, France: IEEE, pp. 3542–3551. DOI: 10.1109/iccvw60793.2023.00381. URL: https://ieeexplore.ieee.org/document/10350394/ (visited on 07/27/2025).

Soundararajan, S. and S. J. Delany (2024). "Investigating Gender Bias in Large Language Models Through Text Generation". In: *Association for Computational Linguistics* Proceedings of the 7th International Conference on Natural Language and Speech Processing (ICNLSP 2024), pp. 410–424.

Stanczak, K. and I. Augenstein (Dec. 2021). *A Survey on Gender Bias in Natural Language Processing*. DOI: 10.48550/arXiv.2112.14168. arXiv: 2112.14168 [cs]. URL: http://arxiv.org/abs/2112.14168 (visited on 05/13/2025).

Stanovsky, G., N. A. Smith, and L. Zettlemoyer (2019). *Evaluating Gender Bias in Machine Translation*. DOI: 10.48550/arXiv.1906.00591. arXiv: 1906.00591 [cs]. URL: http://arxiv.org/abs/1906.00591 (visited on 04/03/2025).

Stella, R. et al. (2021). *A Dataset for Studying Gender Bias in Translation*. URL: https://research.google/blog/a-dataset-for-studying-gender-bias-in-translation/ (visited on 04/10/2025).

Ullmann, S. (2022). "Gender Bias in Machine Translation Systems". In: *Artificial Intelligence and Its Discontents*. Ed. by A. Hanemaayer. Cham: Springer International Publishing, pp. 123–144. ISBN: 978-3-030-88614-1 978-3-030-88615-8. DOI: 10.1007/978-3-030-88615-8_7. URL: https://link.springer.com/10.1007/978-3-030-88615-8_7 (visited on 05/16/2025).

United Nations (2023). *Achieve Gender Equality And Empower All Women and Girls*. URL: https://sdgs.un.org/goals/goal5 (visited on 05/28/2025).

Waldendorf, A. (Apr. 2024). "Words of Change: The Increase of Gender-Inclusive Language in German Media". In: *European Sociological Review* 40.2, pp. 357–374. ISSN: 0266-7215. DOI: 10.1093/esr/jcad044. URL: https://doi.org/10.1093/esr/jcad044 (visited on 06/08/2025).

Wu, S. and M. Dredze (Nov. 2019). "Beto, Bentz, Becas: The Surprising Cross-Lingual Effectiveness of BERT". In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Ed. by K. Inui et al. Hong Kong, China: Association for Computational Linguistics, pp. 833–844. DOI: 10.18653/v1/D19-1077. URL: https://aclanthology.org/D19-1077/ (visited on 07/27/2025).

Wu, Y. et al. (Oct. 2016). *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation*. DOI: 10.48550/arXiv.1609.08144. arXiv: 1609.08144 [cs]. URL: http://arxiv.org/abs/1609.08144 (visited on 06/29/2025).

Xiao, T. and J. Zhu (Nov. 2023). *Introduction to Transformers: An NLP Perspective*. DOI: 10.48550/arXiv.2311.17633. arXiv: 2311.17633 [cs]. URL: http://arxiv.org/abs/2311.17633 (visited on 07/10/2025).

Appendices

Appendix Overview

A. Analysis Summary Table of Core Research Papers

Barclay, P. J. and A. Sami (Apr. 2024)	Investigating Markers and Drivers of Gender Bias in Machine Translations	Investigates implicit gender bias in LLMs using back-translation with DeepL Translates 56 software engineering tasks starting with "she" into genderless languages (Finnish, Indonesian, Estonian, Turkish, Hungarian) and back to English Examines pronoun choices in back-translated texts Extends prior research by Treude and Hata	Comparison of results across five intermediate languages Proposed novel metric for variation in gender across repeated translations: coefficient of unalikeability (UCA) Investigated sentence features that drive bias, especially main verb Compared results from three time-lapsed datasets to test reproducibility	The variation in pronoun selection (quantified by UCA) indicates the language model's uncertainty or hesitancy in implying a particular gender, mirroring human usage of gender-neutral terms. The differing levels of pronoun variation in generated texts for certain tasks have the potential to subtly reinforce gender stereotypes over repeated exposure. Future research should involve more sentences, different translation APIs, longer phrases to capture more context-dependent bias, and further investigation into the correlation between verbs and pronoun variation. The UCA metric allows probing biases without making assumptions about what constitutes a biased formulation.	Different intermediate languages display varying patterns of pronoun use, falling into three groups: Finnish and Estonian (frequent 'he', moderate 'he/she', few missing pronouns), Hungarian and Turkish (many missing pronouns, greater 'you' use), and Indonesian (almost exclusive use of 'he').
Cho, W. I. et al. (2019)	On Measuring Gender Bias in Translation of Gender-neutral Pronouns	Focuses on Korean-to-English translation Korean has gender-neutral pronouns like "기 (kyay)"	First attempt to evaluate gender bias in KR-EN translation for sentiment words and occupations Constructed Equity Evaluation Corpus (EEC) Introduced Translation Gender Bias Index (TGBI) to compare MT systems	Gender bias in machine translation (MT) systems, especially in the translation of gender-neutral pronouns, is not thoroughly investigated for cross-lingual tasks and can perpetuate real-world prejudice.	For sentences where gender determination is not explicitly provided by context, translation systems are recommended to use each gender equally or neutral pronouns if available, to avoid hasty guesses. Occupation translations were found to be more biased than other categories across all systems.
Godsil, R. D. et al. (2016)	The Effects of Gender Roles, Implicit Bias, and Stereotype Threat on the Lives of Women and Girls	Reviews social science research on gender bias, implicit bias, and stereotype threat Uses intersectional lens to assess impacts on academic and professional outcomes for women Notes disparities result from structural discrimination and social stereotypes, not talent	Evidence-based strategies to override bias: Increase diversity / critical mass: three or more women in professional settings improves governance, innovation, sense of belonging In-group peers: networking and peer mentoring fosters community Visible experts: showcasing women in underrepresented fields helps newcomers identify with success Provide effective task strategies for stereotype threat situations	The notion of gender has expanded beyond the binary, and the specific challenges faced by gender nonconforming, transgender, and LBQ individuals warrant separate, dedicated reports.	
Kappl, M. (2025)	Are All Spanish Doctors Male? Evaluating Gender Bias in German Machine Translation	Large-scale evaluation of five commercial MT systems (Google Translate, Microsoft Translator, Amazon Translate, DeepL, SYSTRAN) and GPT-4o-mini Evaluated German to seven target languages with gendered grammar: French, Italian, Spanish, Ukrainian, Russian, Arabic, Hebrew Pipeline: translation, prediction using word alignment and morphological analysis, metric calculation (accuracy, gender-based F1 gaps, stereotype-based performance gaps)	Introduced WinoMTDE: German gender bias evaluation set 288 German sentences based on Winograd schema Balanced gender subjects and stereotype alignment Currently only binary-gendered terms	The WinoMTDE dataset is relatively small (288 sentences), limiting the scope of bias assessment. Stereotype annotations were based on a single person and German labor statistics, potentially introducing bias, especially for ambiguous job titles. The dataset's exclusion of non-binary pronouns and neutral job titles restricts the analysis to a binary gender perspective and overlooks broader gender biases. Certain biases, like semantic derogation (e.g., "teacher" translating to gendered terms), remain unaddressed.	Problem stems from systematic bias within model and training data, not source language ambiguity. The results reveal persistent gender bias in most models across the tested languages. GPT-4o-mini generally outperformed traditional MT systems in terms of accuracy. The study visualizes predictions for occupation groups, showing how translated gender distribution often does not align with real-world distributions, highlighting biases.
Lardelli, M., G. Attanasio, and A. Lauscher (2024)	Building Bridges: A Dataset for Evaluating Gender-Fair Machine Translation into German	Creation of novel resources and MT system evaluation Gender-fair language (GFL) extremely rare, context does not significantly improve it Automatic detection in zero-shot settings is very challenging	Gender-Fair German Dictionary: lists gender-neutral and inclusive German terms with English translations Multi-domain test data from Wikipedia and Europarl Benchmarking GFL in eight translation systems (Google Translate, DeepL, GPT-3.5, GPT-4, NLLB, OPUS MT, Flan-T5, Llama 2)	The study focuses on a single language pair and direction (English to German), a relatively small number of seed nouns and sampled sentences, and deliberately focuses on sentences where the entity's gender is ambiguous or mixed, discarding cases where it is disambiguated.	Research on gender-fair MT is scarce, particularly for German, with existing studies covering only a limited number of languages, scenarios, and domains. Linguistic forms influence mental representation of gender identities, making gender equality in language a crucial goal. German GFL strategies explored: gender-neutral rewording using passive constructions, indefinite pronouns, gender-neutral terms, or participles instead of gendered nouns; gender-inclusive characters using symbols like , :, or _ to combine masculine and feminine forms (e.g., "derdie Leserin"). Words-in-isolation: all tested models demonstrate a heavy bias towards masculine forms (93–96%). Feminine forms are used seldom (2–4%), mainly for stereotypically female professions. Gender-neutral and gender-inclusive forms are rarer (0–2%), appearing mainly for already common gender-neutral words. Words-in-context (Europarl and
Prates, M. O. R., P. H. C. Avelar, and L. Lamb (2019)	Assessing Gender Bias in Machine Trans-lation – A Case Study with Google Translate	Sentence templates used: e.g., "ő egy ápolónő" (Hungarian for "s/he is a nurse") Lexical focus on job positions (from US Bureau of Labor Statistics) and 21 adjectives to explore bias beyond occupations Experiments conducted with 12 gender-neutral languages from diverse families (Hungarian, Finnish, Estonian, Japanese, Chinese) Korean and Nepali initially considered but omitted due to technical issues	Found male pronoun dominance in MT STEM fields consistently defaulted to male Some languages like Basque favored neutral pronouns Adjectives also showed bias (e.g., "Shy" more female, "Guilty" more male) Bias could not be explained by workplace demographics alone		The paper's findings, published as a preprint, received significant media coverage. On December 6, 2018, Google changed its policy to present both feminine and masculine official translations for ambiguous queries, acknowledging their model inadvertently replicated gender biases. The research highlights that gender bias is a statistical phenomenon independent of proprietary tools, suggesting MT engineers must address training data and implement solutions after training rather than relying on scarce unbiased texts. The study concludes unbiased results can be obtained with relatively low effort and marginal performance cost using existing debiasing algorithms.
Rescigno, A. A. and J. Monti (2023)	Gender Bias in Machine Translation: A Statistical Evaluation of Google Translate and DeepL for English, Italian and German	Statistical approach using MT-GenEval dataset Single sentence translation, repeated with context Objectives: detect gender stereotypes in MT systems (Google Translate, DeepL) for English-German and English-Italian Examine if extended context mitigates bias for ambiguous referents	Translation systems show masculine default bias Google Translate and DeepL biased toward masculine outputs Context improves performance, especially for DeepL Contextual errors occur but infrequently	The study suggests the need for more comprehensive evaluations with wider language combinations and more targeted, balanced datasets (e.g., GATE) in future work.	"There is currently no tool to notify them about it" → no detection tool. MT systems still show a strong tendency to default to the masculine gender. Adding context generally improves results but can occasionally lead to erroneous disambiguation.

Savoldi, B., J. Bastings, et al. (2025)	A Decade of Gender Bias in Machine Translation	Comprehensive review synthesising previous research, identifies field limitations, highlights findings and future directions	Empirical methods: Translating gender-neutral sentences and analyzing pronoun frequency Challenge sets with automatic metrics (WinoMT, MT-GenEval) Human-centered quantitative assessment of MT bias	English-centric focus: much of the existing research is overwhelmingly English-centric, often with English as the source language and another Western language as the target, creating a "winner-takes-all" scenario where well-supported languages receive most attention, risking perpetuation of anglocentric biases and overlooking cultural, linguistic, and societal differences. Lack of human engagement: there is a severe lack of direct human involvement in MT gender studies. Most human evaluations are model-centric, supporting structured assessments of model behaviour rather than exploring feedback and experiences of impacted user groups. This gap limits understanding of real-world harms.	Growth in research but gaps remain: increase in papers on gender bias in MT from 2019–2023. Significant gaps: overemphasis on English-centric approaches and tendency to treat bias as purely technical, disregarding social and ethical components. Limited contextual understanding: most studies focus on sentence-level translation, despite gender often requiring broader context. Binary gender focus with emerging inclusivity: majority of studies treat gender as binary, though research increasingly accounts for non-binary identities. Diversity of mitigation strategies with no "clear winner": half of reviewed papers propose strategies (data curation, fine-tuning, inference-time approaches, post-processing rewrites), often modular and scenario-specific.
Savoldi, B., S. Papi, et al. (2024)	What the Harm? Quantifying the Tangible Impact of Gender Bias in Machine Translation with a Human-centered Study	Quantifies gender bias impact in MT via human-centered study Measures effort and cost to correct gender-biased MT outputs Simulated post-editing task: participants corrected outputs for feminine and masculine references	Gender bias in MT leads to higher effort and cost for feminine translations	Study is limited to binary gender categories, acknowledging that this does not imply a binary view of gender identity. This choice was made for controlled experimental conditions, as non-binary and neutral expressions are not yet standardized and would introduce confounds related to participants' familiarity and cognitive load.	
Sczesny, S., M. Formanowicz, and F. Moser (2016)	Can Gender-Fair Language Reduce Gender Stereotyping and Discrimination?	Discusses implementation and effectiveness of GFL policies Focuses on influence on mental representation of gender and reducing stereotyping	GFL more accepted with regulations and frequent use	More research is needed to evaluate the effectiveness of language-related policies and provide evidence-based rationale for policy-making, especially across different grammatical languages and stages of GFL implementation. Challenges and effectiveness of GFL: spontaneous use remains infrequent despite guidelines. Factors influencing use include deliberate processes (attitudes, intentions) and habitual processes (repetition of past behaviour), with context also playing a role (e.g., official texts vs informal communication). Attitudes and intentions toward GFL are only moderately favourable; future research should identify crucial factors for deliberate use, such as political attitudes or acceptance of traditional gender arrangements. Perceptual effects: GFL can influence perceptions, e.g., using gender-fair language in job descriptions can affect children's perceptions	Characteristics of grammatical gender languages (e.g., German): every noun has grammatical gender; gender of personal nouns typically matches referent; personal pronouns are gendered; pronouns and dependent words signal noun gender; referential gender often explicit but asymmetries exist (e.g., French 'homme' = man/human, German 'alle Wähler' = all voters, masculine for mixed group). GFL policies and implementation: international standards (UNESCO, European Commission) regulate internal documents but are not mandatory. National variations: availability and implementation vary. Mandatory usage: Austria strictly enforces GFL; Germany includes feminine forms in dictionaries and education.
Shrestha, S. and S. Das (2022)	Exploring Gender Biases in ML and AI Academic Research through Systematic Literature Review	Systematic review of gender bias in ML and AI Detailed review of 120 peer-reviewed papers from Google Scholar, ACM, IEEEXplore Filtered for English, accessibility, completeness, relevance to gender bias in automated systems	Key findings in ML/AI: Models reflect societal and data biases Gendered nouns and intersectional biases present in multiple languages Google Translate biased toward male defaults, exceptions for adjectives Occupation words more biased than adjectives Bias observed in AI applications (justice, medical robots, self-driving cars, recommender systems)	The systematic review acknowledges potential limitations, such as missing relevant papers due to technical search constraints (e.g., limited time, keywords, database platforms).	
Smacchia, M., S. Za, and A. Arenas (2024)	Does AI Reflect Human Behaviour? Exploring the Presence of Gender Bias in AI Translation Tools	Investigates gender bias in AI translation tools Focus on languages that allow subject omission translated into languages requiring explicit subjects Objectives: quantify bias, analyse translation method effects, identify jobs prone to bias, compare AI to human behaviour	Two types of bias in AI translation: gender and converging Converging bias: outputs influenced by previous translations Gender-specific bias: male occupations biased toward male forms, female occupations more diverse Tool-specific behaviour: DeepL near-perfect distinction, Google Translate variable, Microsoft Azure biased toward male jobs, GPT-3.5 shows evolving bias over sequences Human responses less biased but still show masculine	Limitations include a small dataset (10 jobs, 30 sentences), limiting generalizability. Future work could expand the dataset, explore more complex sentence structures, other languages (e.g., Spanish, Persian) that allow subject omission, and broader demographic variety in human surveys. Geographical location and iterative request nature could also influence results.	The study confirmed gender bias in AI translation tools reflects underlying training data. Research methodology provides preliminary insights into bias phenomena.
Stanczak, K. and I. Augenstein (2021)	Survey on Gender Bias in Natural Language Processing	Comprehensive survey of 304 papers on gender bias in NLP Summarises developments, identifies limitations, proposes recommendations	◦Recommendations: Diversify metrics, develop standard evaluation benchmarks and tests for comparability, encourage data collection for gender-inclusive task-specific datasets, and address typological variety	Historically, research has been in a strictly binary setting, but there is increasing importance given to gender-inclusive research that accounts for non-binary identities and pronouns.	Nature of gender bias in NLP: stems from implicit sexism in text, biases in model parameters, societal gender gap. NLP models can perpetuate and amplify biases. Most research focuses on English corpora; need for work in other languages with morphological gender agreement. Gender bias increases with model size.
Stanovsky, G., N. A. Smith, and L. Zettlemoyer (2019)	Evaluating Gender Bias in Machine Translation	Uses challenge set with non-stereotypical gender roles to evaluate bias	WinoMT: multilingual automatic evaluation of gender bias 3,888 English sentences from Winogender and WinoBias Evaluates coreference resolution based on roles All tested MT systems showed gender bias Bias follows stereotypes rather than context	The use of gender-swapped adjectives (e.g., "The pretty doctor asked the nurse to help her") to reduce bias was shown to be impractical as a general debiasing scheme, as it assumes oracle coreference resolution.	First foundation: study contributes to work evaluating MT systems using challenge sets, moving beyond BLEU metrics to assess specific linguistic phenomena.
Ullmann, S. (2022)	Gender Bias in Machine Translation Systems	Corpus-linguistic analysis of 17 million English-German sentence pairs Interdisciplinary team (linguistics, philosophy, CS, engineering) analyzed 5% subset to identify gender imbalances Tested techniques to reduce bias in MT system trained on this corpus	German-English bias testing Male pronouns and nouns more frequent than female Pre-training mitigation techniques: downsampling, upsampling, counterfactual augmentation Counterfactual augmentation worked best but still imperfect Conclusion: MT bias persists without intervention, but can be mitigated with little computational effort Interdisciplinary work crucial for long-term solutions		Cause of bias: LLMs trained on vast internet data, often lacking diversity, overrepresenting dominant groups, containing misinformation or harmful language. Systemic bias: structural imbalances transferred via automated processes. Technical bias: system design can constrain data processing, causing unfair distribution. Semantic bias: associative relations (e.g., 'he' + 'doctor'). Amplification of bias occurs during training, exaggerating existing distributions (e.g., cooking associated with women, implying only women cook).

B. Datasets and Evaluation Tables

B.1. Handcrafted Test Set Sentences

English	**German**	**Label**
Neutral		
Today the weather is beautiful	Heute ist das Wetter schön	0
I like this guy.	Ich mag diesen Kerl.	0
How are you?	Wie geht es dir?	0
BERT is a transformers model pre-trained on a large corpus of multilingual data in a self-supervised fashion.	BERT ist ein Transformer-Modell, das auf selbstüberwachte Weise auf einem großen Korpus mehrsprachiger Daten vorab trainiert wird.	0
Knowledge is love and light and vision.	Wissen ist Liebe und Licht und Vision.	0
Neutral with Gendered Subjects		
My mother is an engineer.	Meine Mutter ist Ingenieurin.	0
Is this boy a good artist?	Ist dieser Junge ein guter Künstler?	0
I am living with my sister, who is also my best friend	Ich lebe mit meiner Schwester, die auch meine beste Freundin ist	0
My father was an excellent cook.	Mein Vater war ein ausgezeichneter Koch.	0
The girls went hiking.	Die Mädchen gingen wandern.	0
Biased		
Do you like our maths teacher?	Mögen Sie unsere Mathelehrerin?	1
The doctor was late to work today.	Der Arzt kam heute zu spät zur Arbeit.	1
Tomorrow the students are leaving for a class trip.	Morgen gehen die Studenten zu einer Klassenfahrt.	1
This nurse does not work hard.	Diese Krankenschwester arbeitet nicht hart.	1
Athletes earn a lot of money.	Sportler verdienen viel Geld.	1
GFL Variants		

Continued on next page

English	**German**	**Label**
Engineers are kind and hard-working	Ingenieur*innen sind lieb und fleißig	0
Engineers are kind and hard-working	Ingenieure und Ingenieurinnen sind lieb und fleißig	0
Teachers are kind and hard-working	Lehrende sind lieb und fleißig	0
Teachers are kind and hard-working	Lehrer:innen sind lieb und fleißig	0
Teachers are kind and hard-working	Lehrerinnen und Lehrer sind lieb und fleißig	0
Teachers are kind and hard-working	Lehrer sind lieb und fleißig	1
Teachers are kind and hard-working	Lehrerinnen sind lieb und fleißig	1
Job Posting (Real-world)		
We're seeking someone to join our team Office 365 squads to lead the design, development, and integration of Gen AI apps and integration using Microsoft Copilot Studio.	Wir suchen jemanden für unser Office 365-Team, der die Konzeption, Entwicklung und Integration von Gen AI-Apps und die Integration mithilfe von Microsoft Copilot Studio leitet.	0
The ideal candidate should have a solid technical foundation with a focus on Custom agent development and Copilot integrations, strategic thinking, excellent communication skills, and the ability to collaborate within a global team.	Der ideale Kandidat sollte über solide technische Grundlagen mit Schwerpunkt auf der Entwicklung kundenspezifischer Agenten und Copilot-Integrationen, strategisches Denken, ausgezeichnete Kommunikationsfähigkeiten und die Fähigkeit zur Zusammenarbeit in einem globalen Team verfügen.	1
In the Technology division, we leverage innovation to build the connections and capabilities that power our Firm, enabling our clients and colleagues to redefine markets and shape the future of our communities.	Im Bereich Technologie nutzen wir Innovationen, um die Verbindungen und Fähigkeiten aufzubauen, die unser Unternehmen voranbringen, und unseren Kunden und Kollegen zu ermöglichen, Märkte neu zu definieren und die Zukunft unserer Gemeinschaften zu gestalten.	1

Continued on next page

English	**German**	**Label**
This is a Lead Workplace Engineering position at VP level, which is part of the job family responsible for managing and optimizing the technical environment and end-user experience across various workplace technologies, ensuring seamless operations and user satisfaction across the organization.	Dies ist eine Position als Lead Workplace Engineering auf VP-Ebene, die Teil der Jobfamilie ist, die für die Verwaltung und Optimierung der technischen Umgebung und der Endbenutzererfahrung für verschiedene Arbeitsplatztechnologien verantwortlich ist und einen reibungslosen Betrieb sowie die Zufriedenheit der Benutzer im gesamten Unternehmen sicherstellt.	1

B.2. Performance of Dataset Tuning Test Runs

Metric	Dataset A	Dataset B	Dataset C	Dataset D	Dataset E
Macro F1	0.972	0.966	0.953	0.956	0.969
Accuracy (held-out)	0.972	0.967	0.955	0.957	0.976
Accuracy (handcrafted)	0.808	0.808	0.769	0.808	0.808

Table 15.: Evaluation results for datasets A-E.

B.3. False Positives and False Negatives from Held-out Test Set

Error Type	English Text	German Text
False Positive	Accordingly, the President of the French Republic, the President of the European Council and the French Prime Minister asked me to visit, before the President of the French Republic, the ten countries which are currently asking for only one commissioner.	In diesem Sinne haben das Oberhaupt der Republik, die Präsidentschaft des Europäischen Rates und das französische Regierungsoberhaupt mich beauftragt, vor der Rundreise des Oberhaupts der Republik die zehn Länder aufzusuchen, die gegenwärtig nur einen Kommissionssitz beanspruchen.

Error Type	English Text	German Text
False Positive	In so doing, we are beginning to train the next generation of police officers to work and operate throughout Europe; in other words, we will be preparing them to implement Community law and joint and Community actions.	Wir beginnen also jetzt mit der Ausbildung der nächsten Generation von Polizeibeamteten, die in der Lage sein sollen, auf europäischer Ebene zu arbeiten und zu handeln, d. h. sie werden darauf vorbereitet, das Gemeinschaftsrecht anzuwenden und die gemeinsamen und gemeinschaftlichen Maßnahmen umzusetzen.
False Positive	The Heads of State and Government therefore agreed a number of measures to promote the development of risk capital in the European Union, with a deadline for implementing the Risk Capital Action Plan of 2003.	Die Staats - und Regierungoberhäupter beschlossen deshalb eine Reihe von Maßnahmen zur Förderung der Entwicklung von Risikokapital in der Europäischen Union, um den Risikokapital - Aktionsplan bis zum Jahr 2003 vollständig umzusetzen.
False Positive	We will, over the coming weeks, have to take account of the results of the dialogue between the two political leaders, or of the absence of such a dialogue.	In den kommenden Wochen werden wir die Ergebnisse des Dialogs zwischen den beiden politischen Spitzen bzw. das Ausbleiben eines solchen Dialogs zur Kenntnis nehmen müssen.
False Positive	Would you go for treatment to somebody who knows all the surgical terms in Italian, English, French and German, or would you go to a surgeon?	Würden Sie sich von einem Menschen, der sich ausgezeichnet in den chirurgischen Fachbegriffen in Italienisch, Französisch und Deutsch auskennt, oder von einem Menschen, der als Chirurg ausgebildet wurde, operieren lassen?
False Positive	I have just been to the station to see my uncle off.	Ich war gerade am Bahnhof, um mich von meinem Onkel zu verabschieden.
False Positive	The specialists are intelligent.	Die Sachkundigen sind intelligent.
False Positive	The recipient is responsible.	Rezipierende ist verantwortlich.

Error Type	English Text	German Text
False Positive	What we still need are more experts to guide our companies through complex procedures at European level.	Was wir noch brauchen, sind weitere Fachleute, die unseren Betrieben in schwierigen Prozessen auf europäischer Ebene helfen.
False Positive	I shall try very briefly to pinpoint a few political aspects of the four areas touched on in greater or lesser detail by all the speakers, i.e. the new political approach in the social agenda, secondly the content, thirdly the means and fourthly the procedures.	Ich werde versuchen, in aller Kürze einige politische Bemerkungen zu den vier Themenbereichen vorzutragen, die mehr oder weniger ausführlich von allen, die das Wort hatten, angesprochen wurden. Es sind dies erstens das neue politische Konzept der sozialpolitischen Agenda, zweitens der Inhalt, drittens die Mittel und viertens die Verfahren.
False Negative	Here too the local people are frustrated by the immigration of Muslims and the hard line taken by the military.	Hier wird die lokale Bevölkerung ebenfalls durch die Zuwanderung von Muslimen und das unnachsichtige Auftreten des Militärs schwer gebeutelt.

Table 16.: All false positives and false negatives from the held-out test set

B.4. Handcrafted Test Set Results

#	English (short)	True	Predicted	Neutral	Biased	Correct
0	Today weather is beautiful	0	0	0.9996	0.0004	yes
1	I like this guy	0	0	0.9997	0.0003	yes
2	How are you?	0	0	0.9998	0.0002	yes
3	BERT transformers model	0	0	0.6888	0.3112	yes
4	Knowledge is love and light	0	0	0.9992	0.0008	yes
5	My mother is an engineer	0	1	0.4412	0.5588	no
6	Is this boy a good artist?	0	0	0.6223	0.3777	yes
7	I live with my sister	0	0	0.9988	0.0012	yes
8	My father was an excellent cook	0	0	0.9080	0.0920	yes
9	The girls went hiking	0	0	0.9986	0.0014	yes
10	Do you like our maths teacher?	1	1	0.0917	0.9083	yes
11	The doctor was late today	1	1	0.0007	0.9993	yes
12	Tomorrow students leaving	1	1	0.0007	0.9993	yes
13	This nurse does not work hard	1	1	0.0017	0.9983	yes
14	Athletes earn a lot of money	1	1	0.0012	0.9988	yes
15	Engineers are kind (GFL form)	0	0	0.9541	0.0459	yes
16	Engineers are kind (neutral plural)	0	1	0.0008	0.9992	no
17	Teachers are kind (neutral)	0	0	0.9973	0.0027	yes
18	Teachers are kind (colon form)	0	1	0.0314	0.9686	no
19	Teachers are kind (explicit plural)	0	1	0.0005	0.9995	no
20	Teachers are kind (male plural)	1	1	0.0008	0.9992	yes
21	Teachers are kind (female plural)	1	1	0.0007	0.9993	yes
22	Seeking someone for Office 365 team	0	0	0.9964	0.0036	yes
23	Ideal candidate with solid foundation	1	1	0.0006	0.9994	yes
24	Technology division leverages innovation	1	1	0.0004	0.9996	yes
25	Lead Workplace Engineering position	1	1	0.0015	0.9985	yes

Table 17.: Handcrafted test set results: model predictions and confidence scores

C. Use of Artificial Intelligence

C.1. Perplexity.ai for Literature Research

Prompted Perplexity.ai to find additional sources and references on gender bias in EN-DE MT.

perplexity

Key Sources on Gender Bias in Machine Translation and LLMs (English–German)

Below is a curated list of research sources that specifically investigate gender bias in machine translation (MT) systems and large language models (LLMs) between English and German. Each entry includes a brief summary of the research focus.

1. Gender Bias in Machine Translation: A Statistical Evaluation of Google Translate and DeepL for English–German

- This study statistically analyzes gender stereotypes in Google Translate and DeepL when translating between English and German (and English–Italian). It examines how context (e.g., previous sentences) affects gender disambiguation, especially for professions and ambiguous references. The research provides datasets and tools for further analysis[1].

2. Measuring Gender Bias in German Language Generation

- This work develops a German-specific classifier to measure binary gender bias in LLMs, comparing outputs from German GPT-2 and GPT-3 models. It finds a positive bias toward female subjects in German, but also notes that positive regard often aligns with sexist stereotypes. The study includes comparative analyses between English and German outputs, highlighting differences in caregiving and sexualization biases[2].

3. Adapting Psycholinguistic Research for LLMs: Gender-inclusive Language in a Coreference Context

- This research adapts psycholinguistic methods to evaluate how LLMs handle gender-inclusive language in English and German. It finds that while English LLMs tend to maintain gender consistency, they still show a masculine bias. In German, the masculine bias is stronger and often overrides gender-neutral strategies, though inclusive language increases the probability of feminine and neutral references[3].

4. Does Context Help Mitigate Gender Bias in Neural Machine Translation?

- This paper investigates whether context-aware neural machine translation models can reduce gender bias when translating stereotypical professions from English to German. Results show that while context can improve translation accuracy for feminine terms, it does not always mitigate bias and can sometimes amplify it, indicating the need for more nuanced bias mitigation strategies[4].

5. Evaluating Gender Bias in German Machine Translation

- A project at Technische Universität Berlin aims to refine methods for assessing gender bias—specifically underrepresentation and stereotyping—in German MT. The project tests popular MT systems (including DeepL and ChatGPT) for gender biases and works on improving datasets and evaluation processes for fairer translation technologies[5].

6. How Prevalent is Gender Bias in ChatGPT? - Exploring German and English ChatGPT Responses

- This paper systematically analyzes prompts and generated responses in ChatGPT, focusing on gender bias in both German and English outputs. It explores problematic issues related to gender representation and stereotypes in LLM-generated content[6].

Summary Table

Source	Focus Area	Languages	Systems Studied	Key Findings
[1]	MT gender bias, stereotypes, context effects	EN–DE	Google Translate, DeepL	Biases in profession translation, context can help but not always resolve ambiguity
[2]	Gender bias in LLM generation	EN–DE	GPT-2, GPT-3	Positive bias for females, but aligns with stereotypes; differences between EN and DE
[3]	Gender-inclusive language in LLMs	EN–DE	LLMs (unspecified)	Masculine bias persists, stronger in German; inclusive language helps but not fully
[4]	Context effects on MT gender bias	EN–DE	NMT systems	Context improves accuracy for feminine terms but may not reduce overall bias
[5]	Evaluation methods for gender bias in MT	DE (with EN inputs)	DeepL, ChatGPT	Ongoing work on datasets and evaluation for bias and stereotyping
[6]	Gender bias in ChatGPT outputs	EN–DE	ChatGPT	Systematic analysis of gender bias in responses across both languages

These sources collectively provide a comprehensive overview of current research addressing gender bias in machine translation and LLMs between English and German, covering both technical evaluations and methodological advancements[1] [2] [3] [4] [5] [6].

⁂

Research Papers on Detection Systems for Gender Bias in Model Outputs

Here are key research papers that propose or implement detection systems for identifying potential gender bias in outputs from machine translation systems or large language models (LLMs):

1. "Evaluating Gender Bias in Machine Translation" (Stanovsky et al., ACL 2019)

- Presents a challenge set and an automatic evaluation protocol for analyzing gender bias in machine translation.

- Uses coreference resolution datasets and morphological analysis to automatically detect gender bias in translations, focusing on whether gendered forms (e.g., feminine inflections for professions) are correctly used.
- Provides both the evaluation method and publicly available data and code for benchmarking MT systems[7].

2. "Investigating Gender Bias in Large Language Models Through Text Generation" (Soundararajan & Delany, 2024)

- Develops a two-fold evaluation system:
  - Trains a classifier (using BERT and variants) on LLM-generated sentences to detect whether outputs are consistent with or contradictory to gender stereotypes.
  - Analyzes the likelihood of LLMs adding gendered adjectives, using a gender lexicon for reference.
- The classifier and analysis framework serve as detection systems for identifying and quantifying gender bias in LLM outputs[8].

3. "To Reduce Bias, You Must Identify It First! Towards Automated Gender Bias Detection" (ICIS 2022)

- Proposes an early prototype for automated gender bias detection based on a lexical approach.
- The tool includes a gender lexicon, a gender-detection function for character names, and a dashboard for user interaction.
- Designed to analyze large volumes of text for gender bias, supporting human assessment and scalable detection[9].

4. "Exploring gender biases in ML and AI academic research" (Frontiers in Artificial Intelligence, 2022)

- Reviews 120 papers on gender bias in ML/AI, specifically highlighting 19 that present detection mechanisms or frameworks.
- Examples include:
  - Schwemmer et al.: A framework for benchmarking gender bias in facial recognition technology.
  - Serna et al.: InsideBias model for detecting bias in deep neural networks.
  - Pena et al.: FairCVtest, a framework for detecting gender bias in training data for automated recruitment.
- The review provides an overview of detection methods across modalities (text, image, recruitment), some of which are adaptable to language model outputs[10].

Summary Table

Paper/Framework	Detection Approach	Application Domain	Key Features
Stanovsky et al. (2019)	Morphological analysis, coreference challenge set	Machine Translation	Automatic detection of gendered translation errors, public code/data
Soundararajan & Delany (2024)	Classifier on LLM outputs, lexical analysis	LLMs/Text Generation	BERT-based stereotype detection, adjective bias analysis
ICIS 2022 Prototype	Lexicon-based, gender detection, dashboard	Text analysis (general)	Automated, scalable, supports human review
Schwemmer et al., Serna et al., Pena et al. (reviewed in Frontiers 2022)	Various (benchmarking, neural network analysis, training data audit)	ML/AI (text, image, recruitment)	Multiple frameworks for bias detection in different modalities

These papers provide practical frameworks and tools for detecting gender bias in the outputs of machine translation systems and large language models, supporting both research and applied auditing of AI-generated text.

⁂

1. https://acl-bg.org/proceedings/2023/HiT-IT 2023/pdf/2023.hitit2023-1.1.pdf
2. https://www.inf.uni-hamburg.de/en/inst/ab/lt/publications/2022-kraftetal-informatik.pdf
3. https://arxiv.org/html/2502.13120v1
4. https://aclanthology.org/2024.findings-emnlp.868.pdf
5. https://www.berlin-university-alliance.de/commitments/teaching-learning/sturop/tutorials/archiv/sem_aktuell/Evaluating-Gender-Bias-in-German-Machine-Translation/index.html
6. https://arxiv.org/html/2310.03031v2
7. https://aclanthology.org/P19-1164/
8. https://aclanthology.org/2024.icnlsp-1.42.pdf
9. https://www.dfki.de/fileadmin/user_upload/import/12748_To_Reduce_Bias_You_Must_Identify_It_First!_Towards_Automated_Gender_Bias_Detection_ICIS2022.pdf
10. https://www.frontiersin.org/journals/artificial-intelligence/articles/10.3389/frai.2022.976838/full

C.2. Gemini for Synthetic Data Generation

Prompting Gemini to generate full sentences from the existing Building Bridges dataset (Lardelli et al., 2024), which contained only nouns. I provided the lardelli_singular.csv and lardelli_plural.csv files as input. Manual sentence creation would have been too resource intensive. GPT and Deepseek were also tested, but they did not handle large amounts of data efficiently and only produced output in small batches. The final output was a singular output.csv, which I subsequently reviewed and corrected through additional manual rounds to fix errors.

Prompt:

```
Your Role: You are an expert linguist and data generation specialist. Your task is to
     create a high-quality structured dataset for ML by generating English sentences
    and their German translations based on two CSV files.

Goal: Process lardelli_singular.csv and lardelli_plural.csv to produce a final CSV
    with columns: english,german,label. Generate multiple variations per English
    sentence using different German translation rules.

Input Data:
lardelli_singular.csv: ID,German_MS,German_FS,Gender-Neutral,Gender-Inclusive,English
    ,English_P
lardelli_plural.csv: ID,German_MP,German_FP,Gender-Neutral,Gender-Inclusive,English

Instructions:
1. Iterate through lardelli_singular.csv.
2. Generate 3-5 simple, correct English sentences for the singular noun (English) and
     plural noun (English_P). Create singular and plural versions.
Example: "The deputy is having a great day." / "The deputies are having a great day."
3. Generate German translations:
  - Singular: use current row.
    - Label 1: German_MS, German_FS
    - Label 0: Gender-Neutral, Gender-Inclusive
  - Plural: match ID in lardelli_plural.csv
    - Label 1: German_MP, German_FP
    - Label 0: Gender-Neutral, Gender-Inclusive
4. Quality rules:
  - Ensure grammatical correctness (articles, adjectives, verbs, case, number).
  - Skip missing values.
  - Generate all possible variants for balance.
  - English sentences should be varied.
  - Final CSV: english,german,label (use quotes if commas appear).
```

```
Example Output:
english,german,label
"The deputy is having a great day","Der Stellvertreter hat einen tollen Tag",1
"The deputy is having a great day","Die Stellvertreterin hat einen tollen Tag",1
"The deputy is having a great day","die Stellvertretung hat einen tollen Tag",0
"The deputy is having a great day","der*die Stellvertreter*in hat einen tollen Tag",0
"The deputies are having a great day","Die Stellvertreter haben einen tollen Tag",1
"The deputies are having a great day","Die Stellvertreterinnen haben einen tollen Tag
    ",1
"The deputies are having a great day","Die Stellvertretung hat einen tollen Tag",0
"The deputies are having a great day","die Stellvertreter*innen haben einen tollen
    Tag",0

Begin generation now. If output is too large, produce in batches.
```

Output:

```
The data generation is complete, and the output has been saved to output_translations.
    csv. Overview of the process:

Loaded Data: The lardelli_singular.csv and lardelli_plural.csv files were loaded into
     DataFrames.

Sentence Generation: For each noun in lardelli_singular.csv, 3 singular and 3 plural
    English sentences were generated using simple, grammatically correct templates.

German Translation Generation: For each singular English sentence, corresponding
    German translations were generated using German_MS, German_FS, Gender-Neutral,
    and Gender-Inclusive columns from lardelli_singular.csv. For each plural English
    sentence, matching rows were found in lardelli_plural.csv using the ID, and
    German translations were generated using German_MP, German_FP, Gender-Neutral,
    and Gender-Inclusive columns. Translations were labelled as 1 (gendered) or 0 (
    neutral/inclusive). Rows with missing German translation values were skipped.

Output: All generated sentences and their translations were compiled into a single
    CSV file named output_translations.csv with columns english,german,label. Please
    find the generated CSV file below.
```

C.3. Use of AI for Code Generation

During the development of the practical part, I used ChatGPT and DeepSeek to assist in generating portions of the implementation for the demo, specifically in `app.py`, `utils.py`, and

`translate.py`. The process involved multiple rounds of suggestions and refinements. All AI-generated code was reviewed, adapted, and integrated to ensure correctness and suitability for the project. AI assistance supported implementation but did not replace my own programming work or design decisions. Below is one example of AI-assisted code generation. Due to the iterative nature of the process, not all intermediate outputs can be shown. This example illustrates how AI was used to support the implementation.

Prompt:

```
look at my app.py:
import streamlit as st
import torch
from transformers import BertTokenizer, BertForSequenceClassification
from utils import split_sentences, predict_bias_batch

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
MODEL_DIR = "./model_output"
MAX_LENGTH = 128
BIAS_CONF_THRESHOLD = 0.9

@st.cache_resource(show_spinner=False)
def load_model_and_tokenizer():
    tokenizer = BertTokenizer.from_pretrained(MODEL_DIR)
    model = BertForSequenceClassification.from_pretrained(MODEL_DIR)
    model.to(device)
    model.eval()
    return tokenizer, model

tokenizer, model = load_model_and_tokenizer()

st.title("English to German Translation with Gender Bias Detection")
text = st.text_area("Enter English text here:")

if st.button("Translate"):
    if not text.strip():
        st.write("Please enter some text")
    else:
        st.write("### Results")
        sentences = split_sentences(text)
        results = predict_bias_batch(tokenizer, model, sentences, max_length=
            MAX_LENGTH, device=device, bias_threshold=BIAS_CONF_THRESHOLD)

        for sentence, translation, pred, conf in results:
```

```
            st.write(f"**english:** {sentence}")
            st.write(f"**german:** {translation}")
            if pred == 1 and conf >= BIAS_CONF_THRESHOLD:
                st.warning(f"Bias detected confidence: {conf:.2f}")
            else:
                st.success(f"No bias detected confidence: {conf:.2f}")
            st.markdown("---")
```

and utils.py:

```
import re
from translate import translate
import torch

def split_sentences(text):
    # split on . ! or ? followed by space(s)
    sentences = re.split(r'(?<=[.!?])\s+', text.strip())
    return [s for s in sentences if s]

def predict_bias_batch(tokenizer, model, sentences, max_length=128, device='cpu',
    bias_threshold=0.9):
    results = []
    for sentence in sentences:
        translation = translate(sentence)
        inputs = tokenizer(
            sentence,
            translation,
            return_tensors="pt",
            truncation=True,
            padding="max_length",
            max_length=max_length,
        )
        inputs = {k: v.to(device) for k, v in inputs.items()}
        with torch.no_grad():
            outputs = model(**inputs)
            probs = torch.softmax(outputs.logits, dim=1)
            pred = torch.argmax(probs, dim=1).item()
            confidence = probs[0][pred].item()
        results.append((sentence, translation, pred, confidence))
    return results
```

right now there is only one input field that utilises the translate script. i want another option that the user can choose to manually input translation, for

```
    example a tab selection. there are two input fields where the user inputs english
     and german and click a button "check bias". then the same standard bias
    detection script runs. you may add a new file or change existing ones
```

Output:

```
To add the option for manual translation input, use Streamlit's `st.tabs` to create
    two modes:

1. **Translate Mode** (uses your `translate()` function).
2. **Manual Input Mode** (user provides both English and German text).

You only need to slightly update `app.py` and optionally refactor `predict_bias_batch
    ` to handle one or more pairs directly. You do **not** need to change `utils.py`
    unless you want cleaner logic separation.

import streamlit as st
import torch
from transformers import BertTokenizer, BertForSequenceClassification
from utils import split_sentences, predict_bias_batch

device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
MODEL_DIR = "./model_output"
MAX_LENGTH = 128
BIAS_CONF_THRESHOLD = 0.9

@st.cache_resource(show_spinner=False)
def load_model_and_tokenizer():
    tokenizer = BertTokenizer.from_pretrained(MODEL_DIR)
    model = BertForSequenceClassification.from_pretrained(MODEL_DIR)
    model.to(device)
    model.eval()
    return tokenizer, model

tokenizer, model = load_model_and_tokenizer()

st.title("Gender Bias Detection in English-German Translations")

tab1, tab2 = st.tabs(["Translate", "Manual Input"])

with tab1:
    text = st.text_area("Enter English text here:")
    if st.button("Translate and Check Bias"):
```

```
        if not text.strip():
            st.write("Please enter some text")
        else:
            st.write("### Results")
            sentences = split_sentences(text)
            results = predict_bias_batch(
                tokenizer,
                model,
                sentences,
                max_length=MAX_LENGTH,
                device=device,
                bias_threshold=BIAS_CONF_THRESHOLD,
                use_translation=True,
            )
            for sentence, translation, pred, conf in results:
                st.write(f"**english:** {sentence}")
                st.write(f"**german:** {translation}")
                if pred == 1 and conf >= BIAS_CONF_THRESHOLD:
                    st.warning(f"Bias detected confidence: {conf:.2f}")
                else:
                    st.success(f"No bias detected confidence: {conf:.2f}")
                st.markdown("---")

with tab2:
    manual_en = st.text_area("Enter English sentence:")
    manual_de = st.text_area("Enter German translation:")
    if st.button("Check Bias"):
        if not manual_en.strip() or not manual_de.strip():
            st.write("Please enter both English and German text")
        else:
            result = predict_bias_batch(
                tokenizer,
                model,
                [(manual_en, manual_de)],
                max_length=MAX_LENGTH,
                device=device,
                bias_threshold=BIAS_CONF_THRESHOLD,
                use_translation=False,
            )[0]
            sentence, translation, pred, conf = result
            st.write(f"**english:** {sentence}")
            st.write(f"**german:** {translation}")
            if pred == 1 and conf >= BIAS_CONF_THRESHOLD:
```

```
        st.warning(f"Bias detected confidence: {conf:.2f}")
    else:
        st.success(f"No bias detected confidence: {conf:.2f}")
```

C.4. Use of ChatGPT for Formulation and Language

Throughout this thesis, I used ChatGPT to help with formulation, grammar, and overall coherence. All sentences and content were originally written by me. ChatGPT was only used to rephrase or polish my own formulations. All outputs from ChatGPT were reviewed and checked for correctness. No new knowledge, ideas, or substantive content were generated by ChatGPT. All information, analysis, and conclusions in this thesis are my own work.

Declaration of Independent Work

I hereby declare that I have written this thesis independently and have not used any sources or aids other than those indicated. The thesis has not been submitted in the same or similar form for any other examination. All verbatim or paraphrased passages, as well as any sections created, written, or edited with the help of AI-based tools, are clearly marked and cited. In the appendix of my thesis, I have listed all AI-based tools used, including product names and the prompts I employed, in an AI tools directory. I confirm that I have not used any AI-based tools whose use was explicitly prohibited by the examiner. I am aware that content generated by AI-based tools does not guarantee its quality. I take full responsibility for any machine-generated passages I have included and accept responsibility for any errors, distorted content, incorrect references, violations of data protection or copyright, or plagiarism resulting from the use of AI-generated content.

Berlin, August 29, 2025

. . . . . .

(Signature)