Universiteit Leiden
The Netherlands

Computer Science
Bachelor Thesis

On frogs and swings:
Analyzing gender differences in language use of children
by examining the focus of a gender-predicting BERT model

Sander Honig
s2954508

Supervisors:
Dr. M.J. van Duijn & Dr. G.J. Wijnholds

BACHELOR THESIS

Leiden Institute of Advanced Computer Science (LIACS)
www.liacs.leidenuniv.nl

19/07/2024

Abstract

Many advancements have been made in Computational Linguistics and Natural Language Processing over the past decade, opening up great potential to gain more insight into longer existing linguistic questions. A popular topic of research within the field has been the differences in language use between genders due to its compellingness to a general audience. In this research, we used newly emerging Artificial Intelligence technology to investigate a novel technique to research these differences. For this, we finetuned an Artificial Intelligence model to predict a storyteller's gender based on their language use, measured its *performance*, after which we examined the model's *focus*. We then *compared* our findings with pre-existing related research to analyze if this novel technique can provide any insight into the differences in language use between (binary) genders. Concretely, we used a subset of the ChiSCor dataset with gender labels containing 240 freely told stories by 145 children, which we used to finetune the pre-trained Dutch BERTje model for gender label prediction. To measure performance we used 10-fold cross validation, after which we used the integrated gradients technique to produce attribution scores which we combined in rankings. We performed a manual analysis on these rankings to conclude what the model focussed on. Finally, we compared our model's focus with relevant pre-existing research to analyze whether this technique indeed can provide any insight into the differences in language use between (binary) genders. Our results showed a performance with an average F1-score of 0.56 and an average accuracy of 0.63, however, we argued this modest result might not be limited by the capabilities of the model itself, but instead by the limited dataset. We continued our research on the model's focus with the best performing 10-fold cross validation iteration which showed a performance of an F1-score of 0.79 and an accuracy of 0.79. Our model largely considered the presence of verbs, nouns, and adverbs in domains of adventure and technology as indicators of a male-told story, while it considered the presence of nouns and personal pronouns in the domains of personal relations and explicit female characters as indicators of a female-told story. These results were in line with relevant pre-existing research. This showed the potential one can indeed successfully finetune a BERT model to predict a storyteller's gender with relatively high accuracy to then examine what parts of language it focuses on as a way to gain insight into differences in language use between the (binary) genders.

Acknowledgements

First and foremost, we like to express a deep gratitude towards all members of the thesis group led by Dr. Anna N. van der Meulen: Raaf van Beusekom, Hugo Hulsebosch, and Hanna Straathof. They were always there for us with wise advice –requested or unrequested, upon hitting obstacles or when steady progress was made– and a true source of motivation in the journey this thesis was. Invaluable!

Furthermore, we would like to sincerely thank Dr. Veronica Liesaputra and Dr. Lech Szymanski during our exchange in New Zealand, and Dr. Francien Dechesne here in Leiden for their overwhelmingly passionate and inspiring lectures. These people led us to discover the fascinating and multifaceted topic that Artificial Intelligence is nowadays and inspired us for our graduate studies. Thank you!

Last but not least, we would like to thank our supervisors Dr. Max J. van Duijn and Dr. Gijs J. Wijnholds for their useful suggestions and feedback along the way, even in times when their busy schedules barely allowed. Special credit should be given towards Max for his linguistic expertise and general guidance in the research setup, and Gijs for his technical expertise, valuable feedback on the report, and thinking along with the general outline of this research.

Contents

1 Introduction

Language and technology are all around us. It was only quite recently, in the 1950s, that advancements in computing power led people to start analyzing human language by technology in the emerging fields of Computational Linguistics and Natural Language Processing (NLP) [Jon94] [Sch20]. Many advancements have been made since then in both fields to process and analyze language data, perhaps one of the most notable being the introduction of Artificial Intelligence (AI). These advancements open up great potential to gain more insight into many longer existing linguistic questions. One particularly popular topic within the field is one about whether and what language use differences exist between male and female individuals (one that indeed overlaps with psychology as one's language use relates to one's perception and behavior [Who97]).

We aim to investigate a novel technique to research these differences to shed new light on this century-old question. Our idea is not to analyze text data itself, but rather to *finetune a gender predicting AI model to predict a storyteller's gender based on their language use, to then examine what this model focuses on*. The reasoning being that if an AI model can determine a storyteller's gender with relatively high accuracy by solely focusing on parts of language (i.e. no extracted features), it must focus on relevant parts of speech input to gain such accuracy. By examining differences in focus, we can analyze whether clear differences exist and what they are. This limits the tasks in which humans are involved, therefore having the potential to mitigate –not fully remove– human interpretation bias. For the creation of such an AI model that can predict a storyteller's gender, the popular pre-trained BERT model seemed a promising commencement given its outstanding performance regarding (con)textual interpretation. To investigate the potential of this approach we stated the following research question: *How successful can a finetuned BERT model distinguish informal Dutch speech between boys and girls, what parts of language does it focus on, and can this provide insight into differences in language use between (binary) genders?*[1]

To answer this research question, we first look in Section 2 at other work related to gender classification in general and discuss findings and general concerns present in the field. In Section 3, we discuss certain techniques, introduced in other papers, we used to conduct our research. The discussion of how we conducted our own research to answer the research question is discussed in Section 4. In short, we use a subset of the ChiSCor dataset containing 240 freely told stories by 145 children with gender labeled data of the storytellers provided by their parents. We then preprocess these data to finetune the pre-trained Dutch BERTje model for prediction of the gender label. Additionally, we search for the most optimal training hyperparameters. To test performance we use 10-fold cross validation, after which the integrated gradients technique is used to produce an attribution score per token per story of what the model focuses on. Subsequently, we rank the acquired attribution scores per token in various ways and perform a manual analysis on these

[1] In this research, we are not looking to challenge whether gender is a binary construct or not. Butler [But90] and others have already shown how a strictly binary approach to gender identity in society sells short of capturing the nuances this social construct contains. That is why for our research, as discussed in more detail later, we only train our model on stories of children whose parents considered their child to be of either male or female gender at the time of data collection when an option to leave the question blank was given. This way we hope to gain insight into language differences between people who wish to fall into one of these categories. Although indeed, the participants did not choose this gender categorization themselves, we assume the categorization their parents made coincided given their age.

rankings. To mitigate the problems our limited size test set creates, we create an additional extended test set to gain better insight into the model's focus, although introducing unlabeled data does introduce other additional problems. We discuss the findings of model performance and focus in Section 5. In Section 6 we deeper analyze these findings and discuss our model's generalizability by comparing it to other relevant pre-existing research to gain an answer to all three parts of our research question. Finally, in Section 7 we discuss some potential points of improvement and opportunities for future research we gathered in the process of this research. We end with a conclusion and summary of our entire paper in Section 8.

2 Related work

Research on variances in language use between genders has long been a popular topic of research. This popularity can partly be explained by the topic's compellingness to a general audience [KC17] caused by numerous factors such as curiosity (people are naturally curious about how gender might influence behavior and aspects of life), relatability (everyone can relate to the topic as it pertains to fundamental aspects of human identity), cultural relevance (gender roles are a relevant topic in today's society), combined with the perceived objectivity of scientific articles.

A widely known article –because of its rigorous scope– on the topic of gender variances in language was published in 2008 by Newman et al. [New+08]. For their research, 14,000 text samples were combined, capturing both spoken and written texts from 70 separate studies to conduct a robust analysis using LIWC software[2]on differences in message content, word usage, phrase usage, and sentence structure. Over all documents regarding message content, they found women were more likely to discuss internal processes with others, including doubts, thoughts, emotions, senses, and other people. Conversely, men were more likely to discuss external events, such as objects, processes, occupation, money, and sports. On a linguistic word level, they found women to show greater use of pronouns and adverbs, while men showed greater use of numbers, articles, long words, and swearing. Additionally, women showed a small tendency towards more polite phrases, and negations. However, it is generally important to note that a strong influence of culture on gender behavior exists [BP19].

Research on authorial gender differences can largely be placed in one of two methodological strands: descriptive, where an explanation of observed differences ought to be analyzed, and predictive, where an author's gender ought to be recognized based on certain measures. The latter has recently gained popularity with the increasing popularity of Machine Learning (ML) techniques and resulted in current relatively high gender author discernment accuracy. However, as Koolen and Cranenburgh in their 2017 paper highlighted [KC17], it is difficult to accurately explain what exactly gives rise to these results as often many confounding contextual factors exist, and to not resort to stereotyping and essentialism. Interpretations are often performed with gender stereotypes in mind, resulting in an *emphasis on difference* which enlarges the perceived gap between female and male authors, while an even larger overlap is left unconsidered. They analyzed two theoretical and two practical issues that arise and should be considered when performing NLP research into gender differences. The

[2] LIWC (Linguistic Inquiry and Word Count) is a popular language analysis software tool that analyzes texts and places words in linguistic, psychological, and topical categories.

theoretical issue of *preemptive categorization* captures that categorization can become problematic in cases where differences between the categories are in fact caused by unconsidered factors. Choosing to categorize in such cases and explain the observed outcomes (influenced by unconsidered factors) can reinforce essentialistic ideas and enlarge stereotypes. This issue is enlarged by NLP's *semblance of objectivity*: the automation of gender difference analysis and prediction seems objective by its automatic nature. However, often unconsidered is the human researchers' share to gather the data, select categories, and their desire to build a successful classification model. This results in a model with certain analysis or predictive successes, but in actuality does not capture anything regarding the explanatory value of its output. Linked to these unavoidable theoretical issues are two practical ones: dataset bias and interpretation bias. *Dataset bias* refers to any bias in the output of the model caused by a skewed and inaccurate representation of the population (in this case, the population of males and females). In practice, this usually means for gender prediction models the neglectance of contextual and external author variables, such as time of data collection, physical location of recording, etc. A clear example of this is Baker's research [Bak14] which indicated that gender differences were quite prominent in the British National Corpus. However, upon closer inspection the context turned out to be different: men were mostly recorded at work, while women were mostly recorded at home. Lastly, *interpretation bias* is the phenomenon when researchers too easily attribute differences to gender, when in fact other factors could be at play. Additionally, further supporting research beyond the chosen dataset is often not sought when found results align with "common knowledge", which in actuality, is generally based on stereotypes and thus incorporates the researcher's bias. This becomes additionally problematic when deviant and counterintuitive results are not focussed upon, as this is a form of cherry picking. Even with these potential issues arising from gender analysis (and to a lesser extent gender prediction), Koolen and Cranenburgh highlighted the importance of these NLP tasks, as in fact, female-male differences are existent and worth researching when it comes to cultural production [KC17]. However, it is important to be aware of these theoretical and practical issues when conducting research on authorial gender differences.

Still, the ethical questions arising from automated gender prediction by ML should not be overlooked. As Fosh-Villaronga et al. mentioned [Fos+21], the benefits of this kind of technology might mainly benefit large corporations who seek to more specifically tailor marketing services and increase their users' attention span to increase their marketing profits. This would bypass direct benefit for those whose data is used, while the development of such technologies do have the potential to cause harm to the same group. Increasing explicit treatment differences between classified gender categories opens up the window to discrimination by favoring a certain gender in certain cases while disservicing others. Furthermore, discovering one has been misgendered may impact people's self-esteem, confidence, and authenticity (extra harming people who wish not to be classified as either male or female). These feelings, together with the idea of how members of each gender act, create the potential for this technology to shift from being descriptive to becoming prescriptive [KC17]. However, this does not mean that predictive ML techniques that include gender should be entirely ignored, as they conversely have equal potential to expose and consequently combat existing socially harmful and unconscious biases.

Another often overlooked ethical concern specifically relates to the use of ML and Large Language Models (LLMs): environmental and humanitarian impact. Using ML techniques and training LLMs creates significant environmental and humanitarian costs given our contemporary natural resource management. The large energy demand and need for fresh water to cool, both computational and data centers, results in a large impact on ecosystems and climate by resource depletion and associated carbon emissions. Additionally, the mining of the raw materials required for the creation of these computational and data centers is often done in developing countries, exactly the ones who are already mostly affected by the natural changes caused by the impact on ecosystems and climate [Cra21] [Wei+21]. These concerns might not be specific to gender classification, but concern the use of ML and LLMs as a whole.

3 Background

To conduct our research, we made use of several techniques introduced by other papers which might need further explanation. Below, we briefly discuss the integrated gradient interpretability technique in Section 3.1 and the Rank Biased Overlap (RBO) ranking comparison technique in Section 3.2.

3.1 Integrated gradients

To answer the part of our research question concerning *what part* of the language the BERT model focuses on, we needed to analyze our model more thoroughly. For this, we initially considered the analysis on the model's attention matrices. This seemed intuitive as these values determine exactly what we aim to analyze: the weight the model assigns to each input based on their importance for the task to gain a more optimal output [Vig22], in our case, the predicted gender label. Ultimately, however, we decided not to use this method because of its difficulty for interpretation; every attention head (twelve for BERTje) focuses on different relationships and parts of input, making it hard to create a single accurate importance ranking for words per story. Secondly, we briefly considered the use of Shapley values to learn more about the contribution of each token to the output value. However, as these are more fit for a model that performs on extracted features instead of language sequences [Mol22], this idea was omitted. Lastly, we considered the calculation of attribution scores per token in a story by the use of *integrated gradients*, developed at Google by Sundararajan et al. in 2017 [STY17]. This met our criteria for a metric that produces a clearly interpretable attribution score per token, and is applicable to language sequences, easily implementable[3], and robust.

Integrated gradients is an interpretability technique for classification Neural Network (NN) models that attributes an importance score to all input values given a prediction. The method starts off by first constructing a neutral baseline input that represents an input that lacks any meaningful signal, in our case an input stream of `[PAD]` tokens (internally represented by numerical value 3 in BERTje). This baseline is then linearly interpolated per token in a number of steps from baseline to input, in our case from `[PAD]` token (i.e. numerical value 3) to the desired token (i.e. its numerical encoding of the input token)[4]. At every step, the gradient of each input token is calculated with relation to the model function. This gradient is a mathematical vector, where each of its elements is the derivative of the model function with relation to the input token element. Applied to our

[3] Integrated gradient calculations are largely implemented by Captum, an interpretability package for PyTorch (captum.ai/api/layer.html#layer-integrated-gradients).

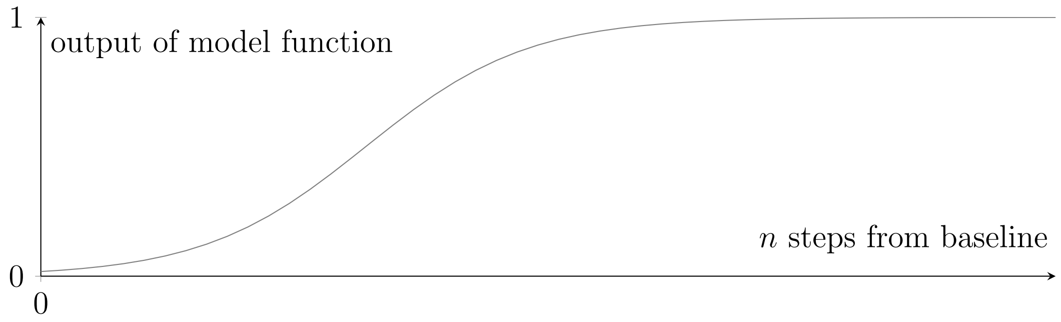


Figure 1: Visualization showing the expected model function output during the integrated gradients technique, as steps from the baseline increase.

tokenized input, it effectively shows in which direction the value of each input token should move to obtain an ascending model function output, with the magnitude of this vector being the rate of value ascension. This means, the greater the (magnitude of a) gradient of a particular input token, the greater its contribution to the output of the model function. At every single interpolation step, the gradients for all tokens are calculated, effectively showing the tokens most contributing to the output value at each interpolation step. For every token at a certain interpolation step, its gradient's magnitude is a measure of importance to the overall prediction output, while the gradient's direction is a measure in what direction it contributed (in our case, all values smaller than zero contribute to a female prediction while all values larger than zero contribute to a male prediction). Additionally, as visualized in Figure 1, the idea is that the most important tokens of a timestep are less interesting at the beginning and ending interpolation steps as the most contributing tokens at these timesteps contribute overall little to the desired output value (hence we use interpolation, and not just solely the original input). In order to account for this, the most attributing tokens at a timestep where the desired output value increases most strongly are more heavily weighted than timesteps where very little difference in output value was observed. This behavior is implicitly implemented by taking an integral of the tokens' attribution scores over all interpolated steps (hence the name integrated gradients), however, as these interpolated steps cannot become infinitely small for an integral, instead an approximation using summation is used. This results in the tokens that attribute most to the model's output when the model's output actually changes the most, to gain the highest (absolute) scores.

3.2 RBO

To objectively compare ranked elements, many different measures can be used. Many of these measures, however, are only applicable to conjoint rankings (i.e. lists that both contain the exact same elements), like Spearman's rho or Kendall's tau, something which is incompatible with our approach as some of our ranking methods produce nonconjoint rankings as they split male and female rankings. Initially, we considered Fagin et al.'s [FKS02] extension to Spearman's rho to account for nonconjointness as it is one of the more intuitive measures. While simpler and most

[4] An often used simpler visual analogy for the neutral baseline input is one for image classification NNs, where it would be a black image. Here linear interpolation would consist of steps of increasing saturation from black to the originally saturated input image.

Age	Girls	Boys
4	4	10
5	14	14
6	6	10
7	21	12
8	12	12
9	10	7
10	2	4
11	1	5
12	1	0
Total part.	71	74
Total stories	117	123

Table 1: Age and gender distribution of our dataset.

preferable on paper, this measure didn't have a clearly defined output range which undermined its interpretability. This led us to choose a different nonconjoint, and relatively novel, similarity measure proposed in 2010 by Webber et al. [WMZ10]: *Rank Biased Overlap* (RBO).

The RBO measure is a nonconjoint top-weighted ranking similarity measure in the range [0, 1], where 0.0 means complete disjointness and 1.0 means identical. Note that an in-between value of, let us say 0.5, might not be as intuitive as one might think at first sight, since what it means to be "50% similar" is rather ambiguous. The RBO measure takes into account both element presence and order, where element absence is weighted most heavily and element order is weighted according to the distance of the difference. Additionally, RBO is a top-weighted measure, meaning that elements higher up in the ranking have more weight than elements further down. Exactly how this weight is distributed is determined by parameter p, which is in range $[0, 1]$. Since p should be determined depending on the size of the rankings, we discuss its exact values in their relevant sections (Section 4.3.1 and Section 4.3.2).

4 Methodology

The dataset that we used to conduct our research is the ChiSCor dataset introduced by Van Dijk et al. (2023) [Dij+23]. The full dataset consisted of 619 fantasy stories in Dutch, told by 442 children at their elementary school who were given the chance to freely come up with a story they wanted to tell. Parents were given the opportunity to voluntarily fill in an additional form with metadata about their child(ren), including their gender. Not all parents filled in this voluntary metadata (for two children the gender section was left blank and for 295 others no form was returned at all), which resulted in only 240 stories by 145 unique children being fit for a large part of our research. Their age and gender distribution can be found in Table 1.

We acknowledge data augmentation techniques could have been used to supply gender data for those without by extracting information from the raw audio files the dataset provides. However, we like to stress our decision not to use these as these techniques give rise to their own inherent problems. Foremost would be what technique to use. Techniques such as automatic classification by pitch threshold [HWN12], classification by another AI system [SA22], or manual listening with *ad hoc* classification all already incorporate their own inherent interpretation biases about gender, eluding our set out purpose of mitigating it. Secondly is the accuracy these techniques provide. While any technique is unlikely to reach an accuracy of 1.0 (however lacking means to verify it), the augmented dataset is likely to cause a false perception of objectivity in our research and possible future research using the augmented dataset. Therefore, all data should be treated as having an accuracy of the portion of originally labeled stories, undermining the goal of the data augmentation. If our goal was to solely predict a storyteller's gender this would be less of a concern as we could assume falsely augmented correctly learned data still as successful, however, our goal is to analyze gender-specific parts of speech where wrong classifications likely only obscure any possible consistencies in language use. Thirdly, this would give rise to an obvious ethical point of contention, whether people who possibly wish not to be binarily classified should be classified in such a way after all for training an AI model. All things considered, we chose not to use and release an augmented ChiSCor dataset.

As a basis for an AI model that can accurately predict a storyteller's gender, we decided to use a popular pre-trained BERT model, developed at Google by Devlin et al. [Dev+19] given its open-source nature, quality documentation, and outstanding performance regarding (con)textual interpretation. More specifically, since our dataset concerned Dutch texts, we used the popular Dutch BERT model "BERTje" (the most popular pre-trained Dutch BERT model as of writing, based on a BERT base model), developed by the University of Groningen [Vri+19].

In this section, we discuss exactly how we conducted our research. More specifically, we go over the steps performed to preprocess the dataset for finetuning the BERT model in Section 4.1. Next, in Section 4.2, we discuss the process of how we finetuned our model and the training parameter to optimize gender predicting performance after which we discuss how we tested this performance with 10-fold cross validation. Finally, in Section 4.3 we discuss our approach to analyzing our model's focus on our initial gender labeled test set as well as an extended gender labelless test set. For the entire process, we used Python 3.7 and PyTorch 1.13.1. An overview of the supplementing code files per section is given in Appendix A.

4.1 Data processing

The dataset consisted of two separate files: one containing the stories and one containing children's metadata received from the voluntarily returned forms. We used the Pandas package to import both files stored locally in CSV format into dataframes. To merge later, we constructed a `child_id` attribute to the stories dataframe by extracting the first three digits from the `id` attribute (the id of every story consisted of five digits: the first three represented the corresponding `child_id`, the latter two their relative story number). Subsequently, we renamed the `id` attribute from the metadata dataframe to `child_id` and merged both dataframes on the same attribute. Finally, we dropped all entries which had an undefined value for the `male` attribute. This attribute was either `0` when "female" or `1` when "male" was reported on the metadata form. This resulted in a dataframe of 240 entries where every entry consisted of a raw story (in `story_raw` and `story_raw_no_newlines`, both attributes equal in content apart from the line separation per sentence in the first; we used

id	story_lemmatized	story_raw_no_newlines	child_id	male	...
10101	er zijn er eens een meisje ze willen heel graag buiten spelen het mogen niet van haar moeder ze gaan toch naar buiten en toen zeggen haar moeder waar gaan je naartoe en het meisje zeggen ik gaan naar buiten einde	er was er eens een meisje. ze wou heel graag buiten spelen. het mocht niet van haar moeder. ze ging toch naar buiten en toen zei haar moeder waar ga je naartoe? en het meisje zei ik ga naar buiten. einde	101	1	
10301	er was een keer een eenhoorn en die was helemaal alleen en er zijn geen eenhoorn er zijn geen eenhoorn en toen zien de prinses dat er een eenhoorn die een babyeenhoorn ze moeder kwijt hebben en dat was het	er was een keer een eenhoorn en die was helemaal alleen. en er waren geen eenhoorns. er waren geen eenhoorns. en toen zag de prinses dat er een eenhoorntje die een babyeenhoorntje ze moeder kwijt had en dat was het.	103	0	
...	...	...	...	...	

Table 2: Example of a dataframe after preparation with Pandas.

the latter), a lemmatized version of the same story, the gender of the author, and several other linguistic and authorial features. An illustrative visualization of this final dataframe is given in Table 2.

After dataframe preparation with Pandas, PyTorch was used to further process the data into a `DataLoader` object for finetuning the pretrained BERT model. For this, we first imported the accompanying tokenizer from the Dutch BERTje model and configured it to lowercase all input to treat all words equally regardless of their capitalization. As BERT models work with uniformly sized input, the tokenizer additionally truncated all stories of over 512 tokens and filled all stories with fewer tokens by adding `[PAD]` tags. The last task of the tokenizer was to wrap the input in `[CLS]` and `[SEP]` tags, and return an attention mask to let the BERT model know where the `[PAD]` tags begin and the tokenized text input ends. After feeding all stories to the tokenizer, the tokenized stories (including their accompanying attention mask and ground truth, i.e. not predicted, gender label) were split in a train, validation, and test set (respectively 70%, 20%, and 10%). As Sklearn's built-in `train_test_split` method only split these entries in two, we sequentially called this method twice to create our desired three-fold split. By argumentation, we ensured the sets were shuffled and stratified on the dataset gender label (i.e. approximately 50% of all three datasets is labeled male while the other 50% female). To solve the problem of a skewed test set proportion (as taking 10% of the 100% dataset is different than 10% of an already 80% train set), we first calculated the adjusted test set size before extracting the test set. Finally, the tokenized words, attention mask, and ground truth labels were combined into a `TensorDataset` object per train, validation, and test set. These three sets were subsequently each put into a `DataLoader` object where they were divided into batches for later training-validation and test iterations. Determining the batch size was done during hyperparameter tuning.

4.2 Model performance

To make the pre-trained BERTje model fit for our gender prediction task, and get an accurate answer on *how successful* a finetuned BERT can be, successful finetuning of the model's parameters and training hyperparameter is key, after which rigorous testing should be performed to test the final model's performance. The finetuning process of the model's parameters entails the tweaking of internal weights and biases to minimize the loss function on input of the train and validation set (i.e. minimize the difference between predicted output and the ground truth), which we discuss in Section 4.2.1. The tuning process of the training hyperparameters is not done automatically (by the optimizer) on the dataset, but is instead done manually as these consider external parameters outside of the model. We discuss this in Section 4.2.2. Lastly, we perform rigorous testing using k-fold cross validation to test the performance of our model after model parameter and training hyperparameter tuning, which we discuss in Section 4.2.3.

4.2.1 Model finetuning

The finetuning process of the model's parameters started with importing the pre-trained model itself. As our prediction output needed to be binary, we defined the model to have just a single output label. We set a threshold to interpret any output value in [0.0, 0.5) as female (as it is closer to its `0` value in the dataset), and any output value in [0.5, 1.0] as male (as it is closer to its `1` value in the dataset). After importing, the model was transferred to the (memory of the) GPU as these are optimized for large parallel computations that take place in the matrix computations inside NNs, in contrast to largely sequential CPUs. Next, we defined the Binary Cross-Entropy (with logits loss) loss function, and the AdamW optimizer with hyperparameter settings we tune later (discussed in Section 4.2.2).

After all data processing and model setup, the model's parameters were ready to be finetuned. This finetuning process consisted of multiple epochs, each epoch consisting of a training phase and a validation phase. During the training phase, the model was set to training mode to enable necessary features such as dropout, batch normalization, and gradient calculation. We then iterated over all batches of the training set and performed for every batch a forward pass, backward pass, and storage of metrics. During the forward pass, the model was fed with the entire batch of encodings and their corresponding attention mask. The model then performed a preliminary prediction and returned a sequence of the outputs (corresponding to each story of the batch) which was then normalized by a sigmoid function. During the backward pass, the loss was calculated using the earlier defined Binary Cross-Entropy loss function, the old gradients were reset, and new gradients were computed. Finally, the parameters (all weights and biases) in the model were updated. When storing the metrics of the model, the accuracy and loss of each batch were stored in a list. After running all training batches, the average train accuracy and average training loss of the epoch were calculated and stored in a list for visualization and analysis.

When training was finished, still within the same epoch, the model was set to evaluation mode to disable necessary features such as dropout, batch normalization, and gradient calculations to create a consistent environment for model inference. During validation, we checked the performance of the model after the finetuning performed in the current epoch with a dataset unused for parameter tuning. For this we iterated over all batches of the validation set and performed again a forward pass, equal to the one during training mode, where the model was fed with the entire batch of encodings and corresponding attention mask, a preliminary prediction was made, which was then normalized by a sigmoid function. In evaluation mode, no backward pass was executed as no parameters should be updated. Again, after running all validation batches, the average validation accuracy and average validation loss of the epoch were calculated and stored in a list for visualization, analysis, and model selection. The program then continued with training the model again etc. Everything discussed within this and previous paragraph resembles one epoch, something which is repeated `nr_epoch` of times (we discuss its value in Section 4.2.2 as this is a hyperparameter). After all epochs, the model with the best performance on the validation set (lowest average loss over all batches) was loaded from memory and considered as the final model. This last step was necessary as it could occur that a model either diverged from a local minimum loss in search for an even better local (or even global) minimum which it never found, or ended up finding a less ideal local minimum, both resulting in the last model not being the best.

After iterating over all epochs, two graphs were plotted: one containing the average training and validation loss, and one containing the average training and validation accuracy. After successful finetuning, the training loss should go down with the validation loss less intensely following, while the training accuracy is expected to go up with the validation accuracy similarly less intensely following. These graphs helped us assess the quality of finetuning and the effect of the current hyperparameters in addition to the test set results. They were obtained in the final stage of our program for model finetuning, where we fed all test set entries to the model and reported the final accuracy, average loss, precision, recall, and F1-score. Additionally, it showed the prediction (in range [0,1]) of each individual test set entry, interpreted prediction (reporting any value in [0.0, 0.5) as 0 and any value in [0.5, 1.0] as 1), and ground truth value of each test set entry. An example of these graphs and test set output used for hyperparameter tuning can be seen in Appendix B.

4.2.2 Hyperparameter tuning

To find the training hyperparameters leading to the best finetuning results, we decided a complete grid search was infeasible due to the limited computational resources and large search space. Instead, we decided to split the hyperparameter search into two stages: in the first stage we considered the model's learning rate, weight decay, and dropout rate, in the second stage the number of epochs, batch size, and effect of lemmatization (indeed, more a parameter of the dataset rather than of the training environment, but by its potential to influence model performance, we discuss it in this section). For measuring the performance impact on the model of these hyperparameters, a variation of metrics could be chosen with none being perfect and all leading to different outcomes. We decided to consider the F1-score metric –where higher is better– over accuracy, average loss, precision, and recall as it is more robust and takes into account both false positives and false negatives[5]. Nonetheless, we stated the accuracy everywhere for its superior interpretability.

The first hyperparameters we considered included: learning rate, weight decay, and dropout rate. For this we performed *ad hoc* manual search due to the very large search space and limited computational

[5] Since females were internally represented as zeros and males as ones, this concretely meant the following. False positives: stories male-predicted when in fact they were female-told. False negative: stories female-predicted when in fact they were male-told.

resources which made a more extensive computationally expensive grid search infeasible. In short, we started our search with standard values obtained by consulting various online fora and articles, observed their effect on the plots and metrics discussed (at the end of Section 4.2.1), and from there made an educated guess on what hyperparameter could lead to an improvement. Examples of this included suspicion of a too low dropout rate and weight decay when training and validation graphs strongly diverged (a sign of overfitting), and suspecting a too low learning rate when train and validation loss rarely changed. Concretely, we ran each configuration once, considered the model's analysis metrics, and made an educated guess on what hyperparameter to tweak. At a point where we could not *ad hoc* improve performance further, we ran two additional runs of the top five performing hyperparameters to obtain a more accurate performance measure of these most promising configurations. As no seed was used during the process, the performance between runs with equal configuration varied automatically, leading to reliable averages. Between every configuration only a single hyperparameter would be changed to isolate the cause of performance differences. We ran this stage twice independently, to find optima for both raw and lemmatized inputs which we would use in our second stage of hyperparameter optimization, twice as both optima might be reached by a different hyperparameter configuration. Other hyperparameters that remained fixed during this first stage included the number of epochs, which we fixed at eight (as most effects of hyperparameters seemed to appear already after three or four epochs), and batch size, which we fixed at four (as batch sizes generally have limited influence on performance, and four was the maximum, thus fastest, for the eleven gigabytes of VRAM available to us). The final performance measure of these top five hyperparameter configurations was the average F1-score of the three runs. Taking the average was preferred over taking the mean to consider all results instead of only a single value, also, the concept of outliers is not a valid one since only three values were considered. Finally, the two best performing hyperparameter configurations –one with raw input, one with lemmatized input– were each selected for stage two of the hyperparameter tuning process. All intermediary results of this first stage of the hyperparameter finding process can be found in Appendix C.

In the second stage, we investigated the effect of the number of epochs, batch size, and lemmatization. As the search space of these variables was more confined, we did perform a grid search in this second stage for a more exhaustive hyperparameter search. We considered lemmatization (yes, no), number of epochs (15, 30), batch size (1, 2, 4), and ran each configuration three times. The danger of overfitting by running too many epochs was mitigated by our design to reload the model of the epoch with the best validation set performance, as discussed in Section 4.2.1. This reasoning led us to decide to not further investigate epochs (8). No model was very stable with a great variation between runs with equal hyperparameters, although some configurations clearly did perform better than others. In the end, similar to stage one, we considered the hyperparameters belonging to the models with the best average F1-score over three runs as the best ones. These were: lemmatization (yes), number of epochs (30), and batch size (2). Combined with our findings in stage one, our final hyperparameters were: learning rate: 0.000010, weight decay: 0.015, dropout rate: 0.25, number of epochs: 30, batch size: 2, and lemmatization: yes[6]. All intermediary results of this second stage of the hyperparameter tuning process can be found in Appendix D.

[6] Even though the tokenizer removed certain prefixes causing implicit lemmatization, the explicitly lemmatized input was more extensive -– especially considering verb conjugations -– enabling the model to better learn associations of verbs regardless of their conjugation and other words regardless of their affixes. This, in combination with the limited size of our training-validation set, we suspect to be the source of the consistent marginal advantage we

4.2.3 *k*-fold cross validation

During hyperparameter tuning, our goal of testing many configurations exceeded the goal of rigorous testing. After deciding upon optimal training hyperparameters, we performed rigorous testing to examine our model's performance[7]. For this, we used k-fold cross validation, with $k = 10$; in short, 10-fold cross validation. This let us test our model and the training hyperparameter on our entire dataset, extra important given its small size, without data leakage. Additionally, to ensure reproducibility, we introduced the use of a seed.

Before 10-fold cross validation could take place, some supplementary code was needed. After performing tokenization, we configured a seed and set the GPU to deterministic mode to ensure reproducibility. This seed was used for the Python hashing method, PyTorch's random number generator, and as an argument for the randomizer in the dataset splitting methods. Instead of the earlier double call of the regular `train_test_split` method for splitting data between a train, validation, and test set, we now used a single call of the `StratifiedKFold` class to split the data in k folds, with $k = 10$. The stratified version of this method again assured an equal gender distribution in every fold, with exactly twelve male-told and twelve female-told stories. Again, we ensured all folds were randomly shuffled by argumentation. Unlike the regular splitting method no encodings were returned, but instead the indices of all folds, which we used to fill lists with the actual encodings. Subsequently, we picked a single fold as our test set[8]. The data in the remaining nine folds was split in an actual training set and validation set by the regular `train_test_split` method (respectively 80% and 20% of the remaining folds). At the end of the program, we added code to save the current model to disk, something we used for saving all ten models. All other code was left unchanged from how it was described earlier in Section 4.1 and Section 4.2.1.

4.3 Model focus

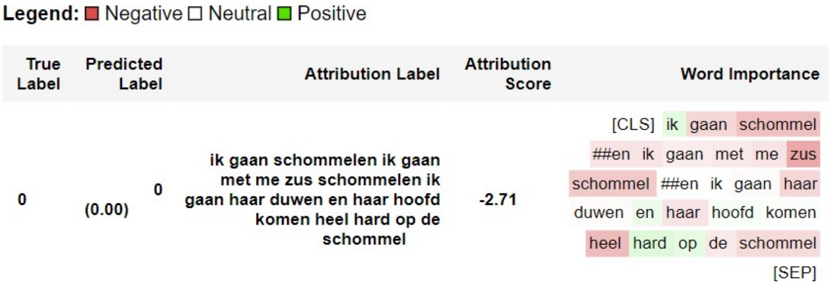
We decided to continue with the best performing 10-fold cross validation model for the remainder of our research. This was based on the reasoning that the model of this iteration must have been most successful in focusing on relevant tokens for gender differentiation, assuming it did not use proxies, such as confounding contextual variables, or plain coincidence. To examine this model's focus we used the integrated gradients technique, as described in Section 3.1.

Practically, this model focus analysis started by loading all data as previously and loading our best performing 10-fold cross validation model. Subsequently, we initialized the `LayerIntegratedGradients` class from the Captum package with our model's embeddings and output function. We fed the `attribute` method of this class the original input values of the model, baseline input values (where all tokens except the `[CLS]` and `[SEP]` tokens were replaced with the `[PAD]` token), and the number

measured over raw input.

[7] k-fold cross validation creates k different models created by k different training-validation sets, something which we come back to later. However, it should be noted that all initial configurations of the same BERTje base model (given the same seed), all hyperparameters, and the majority of the data are equal.

[8] Practically, we chose to manually pick the test fold of an execution, instead of iterating over all folds automatically, as this had two big advantages: testing was more modular making debugging easier, and this left the Python notebook in which we wrote our code better readable, as otherwise, everything would need to be written in a single opaque block of code.



Legend: 🟥 Negative □ Neutral 🟩 Positive

True Label	Predicted Label	Attribution Label	Attribution Score	Word Importance
0	0 (0.00)	ik gaan schommelen ik gaan met me zus schommelen ik gaan haar duwen en haar hoofd komen heel hard op de schommel	-2.71	[CLS] ik gaan schommel ##en ik gaan met me zus schommel ##en ik gaan haar duwen en haar hoofd komen heel hard op de schommel [SEP]

Figure 2: Example of output by Captum's visualization package, visualizing all calculated attribution scores of a sentence.

of interpolation steps which we set to 50 (as suggested by multiple online sources and within the suggested range proposed by the original paper [STY17]). We omitted the tail padding of both the original and baseline input as these tokens did not have influence on the output and this increased calculation efficiency. The `attribute` method produced a matrix with dimensions $[x, y, z]$. Here x reflected the number of model outputs, in our case always 1, y reflected the number of input tokens, and z reflected the total number of hidden nodes, in our case using BERTje always 768 (as Captum additionally captures an attribution score per node). For our purposes, we discarded dimension x, and summed all dimensions of z. We normalized these values in the range $[-1, 1]$, which left us with a single list of attribution scores of length y per story. Finally, and optionally, we visualized the output of these obtained attribution scores with Captum's visualization package. An example of this visualized output can be seen in Figure 2.

4.3.1 Initial test set

Above, we described how the integrated gradients technique for our model focus analysis worked with single input sequences, i.e. stories. This remained the basis of our analysis, but we extended this by making it automatically run over multiple stories and capturing all attribution scores in a Python dictionary[9]. In this dictionary each key was the numerical value of a token, while each corresponding value was a list of attribution scores of each occurrence. We combined all values of a single token to a single value in order to create a ranking of most important tokens the model attended to. Similarly, we created a ranking of most important tokens in the text. We then compared these rankings to analyze whether the model picked up on sensical word classes and domains present in the text.

The two methods we used to analyze the most important words in the text are simple and straightforward; we summarize them below:

- *Count [Text]*: With counting we simply counted all token occurrences per gender.

[9] Practically in code, our decision to consider seven different rankings –as we discuss below– made us introduce several other dictionaries for efficient calculation, although these were theoretically not necessary. Additionally, we saved the average attribution score of a story per token per story, the true total token count per ground truth gender, and both text and attribution Term Frequency (TF) scores per token per story.

- *TF-IDF [Text]*: For TF-IDF scoring we calculated the TF-IDF score for every word. The TF-IDF score is a statistical measure that captures the importance of a word (in our case token) within a document relative to a collection of documents (in our case stories). We grouped all stories of the same gender together by which we made it a measure of how important a token is within the collection of stories told by that gender: this formula is given in Equation 1. It should be noted that by doing this the initial range of values of [0,1] is not valid anymore.

Creating a ranking of the most attended-to words by the model was less straightforward. For this, we had to combine all attribution values of a single token. No single absolute correct way existed for this, hence we considered five, all providing a different perspective on what the model mostly focused on by processing multiple occurrences of the same token differently. We summarize them below:

- *General avg. [Attr.]*: We took the general average attribution score per token. For this, we accumulated all attribution scores per token and divided it by its total number of occurrences, as shown in Equation 2. This method leaves all scores in range [-1, 1], but in no way takes into account how often a term occurred, while an often occurring token with a low attribution score could still have had a large impact in the overall final prediction[10]. This method creates a single ranking where all tokens with a negative value steered the decision towards a female prediction, while all tokens with a positive value steered towards a male prediction.
- *Avg. summation [Attr.]*: We took the summations of a token in a story, of which we took the average over all stories, as shown in Equation 3. The idea is that the summation of an attribution score of a token in a story gives the total score that particular term contributed in the story for the final decision. This could potentially solve the previously stated problem of not taking into account a term that occurred very often in a story and thus greatly contributed towards the decision of either male or female. The downside is that this could give a term a high final score if it occurred very often in a small number of stories. Whether this is desirable depends on the choice of whether we want to consider very important tokens in a small number of stories also as overall important. Here the final score is not in a particular range. Again, a single ranking is created with negative values contributing towards a female prediction while positive values contribute to a male prediction.
- *Summation [Attr.]*: We took a simple summation of the attribution scores of all token occurrences, as shown in Equation 4. The idea is analogous to the previous method, only here we did not punish for occurring in a low number of stories.
- *Summation avg. [Attr.]*: We took the summation of average attribution scores per story, as shown in Equation 5. The idea is to consider per token at most a single attribution score per story. This solves the potential problem simple accumulation might create when a token occurs very often in a small number of stories. This does advantage tokens (with a high attribution score) that occur infrequently in many stories of a gender over tokens occurring very frequently in a few stories. Again, scores are not in a particular range and a single ranking is created.
- *TF-IDF [Attr.]*: We took an adjusted TF-IDF score per ground truth (i.e. not predicted) gender, as shown in Equation 6. The idea is that the adjusted TF-IDF method for texts (which determines how important a token is within the collection of stories told by that gender) might work just as well for attribution scores instead of simple counts (to determine how important a token is for the model within the collection of stories told by that gender). For

this, we first calculate the TF-IDF scores of all tokens using their attribution scores instead of occurrence counts (effectively substituting $occurrences \cdot 1$ by $occurrences \cdot attribution$). Secondly, we sum all TF-IDF values of the same token told by the same gender. Similar to the text TF-IDF score, this causes the output values to not be in a particular range anymore. We believe this method could combine the advantages of all methods above, by considering both tokens occurring in a large number of documents, as tokens that occur very frequently within a story. An additional advantage is the similarity of this measure to the TF-IDF measure we use for the text itself, minimizing the influence by the measure itself on the ranking, that way leading to a more accurate comparison later. This is the only method for attribution score ranking that works with two distinct lists per gender, all other methods use a single ranking where a negative value shows a contribution to a female prediction, and a positive value to a male prediction.

All methods above resulted in a large Python dictionary where every key is the numerical value of a token, while each corresponding value is a single attribution score as a result of applying the above methods. We then sorted this dictionary on value to obtain a ranking.

To objectively compare rankings we used the RBO metric as discussed in Section 3.2. We choose parameter p as 0.991 as we calculated this to give a weight of 97.3% to the first 250 elements, a weight of 99.1% to the first 350 elements, and a weight of 99.9% to the first 566 elements; ideal considering our test set consists of 566 unique tokens.

4.3.2 Extended test set

Only considering this limited size initial test set with known ground truth genders (consisting of only 24 stories) would have had two large advantages. Firstly, when examining the model's focus we would know its accuracy, giving us insight that the model is actually performing its task and does not merely produce (near) random output. Secondly, we would be hinted towards tokens that lead to wrong predictions. This is best explained by an example: if a certain male-told story was predicted as female due to the large number of tokens that the model puts heavy female attribution on, we can see these outliers in the male ranking. These outliers are only likely to be visible in the ranking if the token was involved in multiple wrongful predictions as the ranking methods described above combine multiple attribution scores per token.

While the advantages of having an annotated test set seemed clear and ideal, its limited size was likely to give rise to a major disadvantage, concerning test performance statistics and particularly model focus: it would likely not fully capture the model's focus as our model was finetuned on a way richer training set. Instead, it would likely either underrepresent the model's focus (the model might focus on things the test set did not capture) or overrepresent it (the test set largely contains tokens with high attribution scores). While this challenge always exists for test sets, its extremely limited size could lead to particularly big variations in perceived performance and focus by solely changing a few elements.

[10] Terms do not necessarily have the same attribution score when occurring more than once in a single story, however, our inspection did show their scores are often very similar.

For each token t:

let N be its total number of occurrences over all stories
let $n(i)$ be its total number of occurrences in story i
let S be the total number of stories (S_g of current gender)
let s be the number of stories containing the token
let a_i be its attribution value of occurrence i
let $a_{i,j}$ be its attribution value of occurrence j in story i
let $a_{i,j}$ be its attribution value of occurrence j in story i

Then we combine all attribution scores of each occurrence of t into a single value $V(f)$ with f being the transformation method:

$$V(\textit{TF-IDF [Text]}) = \sum_{k=1}^{S_g} \mathrm{TF}_i \cdot \mathrm{IDF} \tag{1}$$

$$\mathrm{TF}_i = \frac{n(i)}{\text{total number of tokens in story } i}$$

$$\mathrm{IDF} = log_{10}(\frac{S}{s})$$

$$V(\textit{General avg. [Attr.]}) = \frac{\sum_{k=1}^{N} a_k}{N} \tag{2}$$

$$V(\textit{Avg. summation [Attr.]}) = \frac{\sum_{k=1}^{S} \sum_{l=1}^{n(k)} a_{k,l}}{s} = \frac{\sum_{k=1}^{N} a_k}{s} \tag{3}$$

$$V(\textit{Summation [Attr.]}) = \sum_{k=1}^{N} a_k \tag{4}$$

$$V(\textit{Summation avg. [Attr.]}) = \sum_{k=1}^{S} \frac{\sum_{l=1}^{n(k)} a_{k,l}}{n(k)} \tag{5}$$

$$V(\textit{TF-IDF [Attr.]}) = \sum_{k=1}^{S_g} \mathrm{TF}_i \cdot \mathrm{IDF} \tag{6}$$

$$\mathrm{TF}_i = \frac{\sum_{k=1}^{n(i)} |a_{i,k}|}{\text{total of all absolute attribution scores in story } i}$$

$$\mathrm{IDF} = log_{10}(\frac{S}{s})$$

	Girls	Boys	Unknown
Total stories	12	12	379

Table 3: Gender distribution of our extended test set. As metadata was missing, no age information was available.

To shed an additional perspective on our model, and partly mitigate this disadvantage, we decided to append additional data to our initial test set which our model did not train on: the stories earlier discarded by their lack of gender metadata. Including these data had as goal to give us a richer insight into the word types and domains the model focused on beyond the ones present in the limited size initial test set. The gender distribution of this extended dataset can be found in Table 3. Additionally, we changed the p parameter of the RBO metric to 0.998 to give a weight of 67.3% to the first 250 elements, 76.5% to the first 350 elements, and 99.9% to the first 2477 elements to make it better fit for the 2477 unique tokens present in this extended dataset.

Rather ironically, the use of this extended test set caused an exact reversal of the advantages and disadvantages we discussed. While this mitigated –not fully removed– the problem of likely not fully capturing the model's focus, two disadvantages were created. Firstly, this extended test set took away the knowledge about the model's accuracy, hence we potentially end up with a model not being better than random guessing while unaware its interpreted focus is meaningless. Secondly, the attribution ranking methods were configured to act as if the model has an accuracy of 1.0 by accumulating over the model's predicted gender instead of the unknown ground truth gender. This removed the possibility of hinting towards wrong predictions. Furthermore, we acknowledge this lets other problems arise, such as an ethical one, whether people who possibly wish not to be binarily classified should be classified in such a way after all.

Still, although this extended test set is similarly imperfect, we believed it could lead to additional insights into our model's focus.

5 Experiments & results

Above, we described how we conducted our research, in this section we describe the outcome of this methodology. Just as the previous section, this section consists of two parts directly referring to the two parts of our research question. We first discuss in Section 5.1 the performance of our finetuned BERT for predicting a storyteller's gender which we evaluated by means of 10-fold cross validation. We then in Section 5.2 go on to examine what parts of speech this model most strongly focused on to make this distinction, using the integrated gradients technique and the seven ranking methods as described earlier.

Test fold		0	1	2	3	4	5	6	7	8	9	Average
Performance	F1-score	0.38	0.76	0.53	0.70	0.35	0.52	0.44	0.79	0.50	0.58	0.56
	accuracy	0.58	0.71	0.63	0.71	0.54	0.54	0.58	0.79	0.58	0.58	0.63

Table 4: Our finetuned model's performance on the initial test set.

5.1 Model performance

After deciding the most optimal training hyperparameters, we performed 10-fold cross validation to rigorously examine our model's performance[7], and used a seed to ensure reproducibility. The exact details of this implementation were discussed in Section 4.2.3.

The selection of the seed was done using an online random number generator which resulted in the seed of 42 being selected from the interval [0,100]. This led to significant underperformance in the first three folds, by all metrics, compared to our final hyperparameter tuning measure. Therefore we decided to pick the next seed generated by the generator: 9. This resulted in performance more in line with our previous testing during hyperparameter tuning, hence we chose this seed to continue our research. As 10-fold cross validation produced ten separate finetuned models that slightly differ (one for each iteration), we equally obtained ten different performance scores, the results of which can be seen in Table 4. Over all ten iterations, this resulted in a final *average F1-score of 0.56* and an *average accuracy of 0.63*. It should be noted that precision was consistently higher than the recall value, hinting to an overall bias towards a female output (as female was a zero); this further feeds the suspected weakness we describe in Section 7.

However, since our question was how successful a finetuned BERT model *can* be, we decided to regard the top performing iteration (by F1-score). This was iteration seven, which reported an *F1-score of 0.79* and an *accuracy of 0.79*[11], significantly higher than the average. All intermediary results of the 10-fold cross validation testing procedure can be found in Appendix E.

5.2 Model focus

We decided to consider the model of the best performing iteration for the remainder of our research regarding model focus. For this, we assumed the model accomplished this by paying attention to the most relevant parts of speech input for the gender differentiation, and not by proxies, such as confounding contextual variables, or plain coincidence. This would make this model most successful in its focus on relevant parts of speech input, making it similarly most promising to provide insights into differences in language use between (binary) genders.

After we decided upon the specific model and noted its performance, we analyzed *what* parts of language it focussed on, more specifically what word types and domains. For our current research, we decided upon manual analysis ourselves, however, a more professional analysis might be something for further research as we discuss in Section 7. We emphasize our best efforts for an as objective

[11] These performance scores are technically still averages as they are average scores over all test batches. Indeed, the previously mentioned average score over all ten iterations of the 10-fold cross validation are technically averages of averages.

analysis as possible, however, like to stress the impossibility of completely precluding implicit personal biases.

5.2.1 Initial test set

Upon feeding the model the data of our initial test set, the integrated gradients technique produced a unique attribution value per token per story, which we combined in five different ways to produce five different rankings regarding token importance, as discussed in Section 4.3.1. All provided a different perspective on what the model mostly focused on by processing multiple occurrences of the same token differently. Two separate rankings provided an objective analysis of the most important words per gender in the texts.

The rankings to provide an objective analysis of the most important tokens in the texts were not both equally insightful, as we discuss below:

- *Count [Text]*: We noticed that counting was less insightful than imagined. The top of the ranking for both genders was largely dominated by small stopwords. However, removing certain stopwords brings difficulties as well, as that would already incorporate our own biases into the ranking of what words are insignificant. Lower in the ranking most words obtained the same count, yielding a false perception of order; e.g. while the large list of all tokens that occurred three times seems to be ordered, in reality no order exists between them. Considering its drawbacks, we chose to not include this metric in our final discussion of our results.
- *TF-IDF [Text]*: Taking the TF-IDF score for tokens and aggregating identical tokens told by the same gender, seemed to meet our theoretical assumption of being a measure of how important a token is within the collection of stories told by that gender, with stopwords occurring dispersed lower in the ranking. We used this measure later in our final discussion of the results.

As we described previously, no single absolute correct way to combine the attribution scores of a single token, leading us to explore five different ways, all having a different focus. However, not all rankings resulting from these combination methods were equally insightful either, as we discuss below:

- *General avg. [Attr.]*: A general average of all occurrences of a token seemed to show idealistic characteristics with shorter stopwords occurring more in the middle of the ranking, while content-rich words occurred higher to the male and female ends of the spectrum. Additionally, the top male and female words were similar to the ones found in the text by *TF-IDF [Text]*, which is what we would expect for an accuracy of 0.79. For these reasons, we used this measure later in our final discussion of the results.
- *Avg. summation [Attr.]*: The average after summing all attribution scores of a token per story seemed promising. However, upon inspection many very infrequently occurring terms are highly ranked as a result of the punishment of occurring in many stories. While it seemed equally important to focus on very important tokens in a small number of stories, this ranking method seemed to have overshot in this direction with other methods having a seemingly better balance. Therefore, we chose to not consider this measure later in our final discussion of the results.

- *Summation [Attr.]*: Summation most closely resembles the *Count [Text]* method used to analyze the texts, however not counting every token occurrence as 1, but effectively as $attribution \cdot 1$. While mathematically similar to the previous method, the ranking it produced is not at all, showing the great effect normalizing by story occurrences has. After all, this method experienced the same issues as its text counting counterpart, hence we chose to not consider it later in our analysis.
- *Summation avg. [Attr.]*: Taking a summation of averages of a token per story yields an interesting ranking. Both male and female ends of the spectrum contain a lot of stopwords, but to a much lesser extent than previous methods with the same problem. Also surprisingly, the word "en" ["and"] is considered the most expressive male token the model attends to besides its greater use in female stories (see *Count [Text]*). This exposed the problem with this method: a high value in this ranking could point to a token either occurring in many stories, or having a consistent big attention value. The interleaving of these two properties seemed to not result in a meaningful ranking, therefore we chose to not consider it later in our analysis.
- *TF-IDF [Attr.]*: Taking the accumulated adjusted TF-IDF score of a token per gender seemed promising. Our initial intuition of obtaining a measure that determines how important a token was for the model within the collection of stories told by that gender seemed to hold as again most stopwords occur lower in the rankings, while content-rich words occurred higher. Similarly, the ranking occurred similar, but not identical, to *TF-IDF [Text]*, which is what we expected as we used a similar ranking method and got a 0.79 accuracy. Given these promising characteristics, we used this measure later in our final discussion of the results.

Each time, we considered the top 25 ranked tokens per gender. This top 25 of our three chosen rankings is shown in Table 5[12]. All seven rankings in full can be accessed via this paper's Github project as described in Appendix A.

Before shifting our analysis to word types and domain of the rankings, we first briefly examined the similarities of the rankings by their RBO scores shown in Table 6. For both genders, the attention of our model captured by the *TF-IDF [Attr.]* ranking seemed reasonably accurate, as it had a reasonably high RBO similarity score with the one of the text itself. *General avg. [Attr.]* seemed to underrepresent the focus of the model slightly, however, this could also solely be because it did not share the same ranking method, making it more difficult to score high. Between the genders, the total lack of overlap in the *General avg. [Attr.]* can be explained by its construction; the single list the ranking created was simply split in positive and negative values to differentiate between male and female contributing tokens. An additional pleasant observation was the small overlap across genders in the TF-IDF rankings of the text, something our model relatively closely captured.

[12] Words "schommel" ["swing"] and "kikker" ["frog"] being number one and so much ahead of other words in the *TF-IDF [Attr.]* seems peculiar given how uncommon and specific these words are. Their same position in the *TF-IDF [Text]* ranking already gives away part of the cause. Indeed, if we inspect the dataset we see that, coincidentally, the word "schommel" occurs seven times, in three separate stories all told by a girl. Similarly, upon inspection of the word "kikker" we find ten occurrences in five different stories, solely told by boys. While just speculation, it could be that a friend group of girls recently played on the swing, while a friend group of boys recently found a frog, all being inspired by the event for their own story. This is an inherent negative side-effect of the limited size of our dataset.

[13] www.vandale.nl

	TF-IDF [Text]			General avg. [Attr.]			TF-IDF [Attr.]		
Female	schommel	[n]	0.17252641	jullie	[pp*]	-0.45895943	schommel	[n]	0.30477525
	haar	[pp*]	0.11193507	prins	[n]	-0.43907686	amulet	[n]	0.22947860
	hond	[n]	0.08082993	schommel	[n]	-0.29040323	zus	[n]	0.22058301
	trol	[n]	0.07667840	varen	[v]	-0.25994311	vriendin	[n]	0.15582100
	hij	[pp]	0.07652596	zus	[n]	-0.25837650	familie	[n]	0.14495566
	zus	[n]	0.07050864	nul		-0.25330610	meisje	[n]	0.14314929
	ze	[pp]	0.06565141	voetballer	[n]	-0.24954557	haar	[pp*]	0.14301567
	lol	[n]	0.06000918	vriendin	[n]	-0.24778613	##stad	[n]	0.12179743
	familie	[n]	0.05996503	verdrietig	[adj]	-0.24409373	kind	[n]	0.10664627
	meisje	[n]	0.05810404	langzaam	[adj]	-0.23952562	hij	[pp]	0.10197551
	heks	[n]	0.05750880	baby	[n]	-0.23610034	zoeken	[v]	0.09690897
	duwen	[v]	0.05750880	bloemen	[n]	-0.22069535	hond	[n]	0.09246026
	##en		0.05652416	familie	[n]	-0.19054869	bla		0.08985911
	vriendin	[n]	0.05637785	meisje	[n]	-0.17731470	tram	[n]	0.08535734
	maar	[adv]	0.05353466	knuffel	[n]	-0.17083731	##rennen	[v]	0.08350250
	tram	[n]	0.05257948	kind	[n]	-0.16916381	moeder	[n]	0.08348058
	##poli		0.05257948	##dubbel		-0.16749758	##poli		0.07725118
	zombie	[n]	0.04900158	haar	[pp*]	-0.15348738	scoren	[v]	0.07581144
	zeggen	[v]	0.04880114	gat	[n]	-0.14340749	oom	[n]	0.07303520
	me	[pp*]	0.04569205	kapot	[adj]	-0.14172503	verdrietig	[adj]	0.06380856
	hoofd	[n]	0.04496589	blij	[adj]	-0.14122851	heel	[adv**]	0.06299942
	bla		0.04452294	##baar	[adj]	-0.14084593	zombie	[n]	0.06157971
	zoeken	[v]	0.04447706	##meer		-0.13406875	speelgoed	[n]	0.05891076
	spelen	[v]	0.04403981	ring	[n]	-0.13000345	waren	[v]	0.05747571
	kind	[n]	0.04380292	keeper	[n]	-0.12930190	knuffel	[n]	0.05344632
	...			...			...		
Male	kikker	[n]	0.12547375	##kikker	[n]	0.61795437	##kikker	[n]	0.33101872
	##kikker	[n]	0.12547375	amulet	[n]	0.58384508	kikker	[n]	0.18844492
	ik	[pp]	0.11639864	##stad	[n]	0.52224512	baby	[n]	0.15493072
	mama	[n]	0.11221433	stampen	[v]	0.49833589	stampen	[v]	0.15046049
	baby	[n]	0.09810739	overheen	[adv]	0.46434086	overheen	[adv]	0.13623233
	hij	[pp]	0.09218565	regenen	[v]	0.44052370	hij	[pp]	0.13496164
	’		0.07869603	snijden	[v]	0.42574389	jullie	[pp*]	0.13465348
	n		0.07869603	poort	[n]	0.39166624	prins	[n]	0.12497588
	kasteel	[n]	0.07812516	##dwalen	[v]	0.37947539	snijden	[v]	0.12496459
	zien	[v]	0.07691244	solo	[adv]	0.36453815	poort	[n]	0.11491043
	m		0.07460601	kikker	[n]	0.35179389	regenen	[v]	0.10355886
	vis	[n]	0.07460601	uitnodigen	[v]	0.31914275	paard	[n]	0.10102909
	rijden	[v]	0.07038139	eruitzie	[v]	0.31500337	verzinne	[v]	0.09394978
	##d		0.05960283	oom	[n]	0.31316157	eruitzie	[v]	0.09241841
	lopen	[v]	0.05918421	leeuw	[n]	0.25420234	aflopen	[v]	0.08809208
	door	[adv]	0.05693345	##snaam	[n]	0.24936242	kip	[n]	0.08662294
	hem	[pp]	0.05554098	aflopen	[v]	0.23136065	solo	[adv]	0.08569608
	met		0.05524499	##ium		0.22099361	leeuw	[n]	0.07959157
	groot	[adj]	0.05178651	naartoe	[adv]	0.21041547	ik	[pp]	0.07943343
	paard	[n]	0.05115301	nest	[n]	0.20386962	hebben	[v]	0.07705526
	we	[pp]	0.05049455	gluren	[v]	0.20291144	varen	[v]	0.07629865
	schieten	[v]	0.04652397	tovenaar	[n]	0.20139026	deur	[n]	0.07290771
	lat	[n]	0.04652397	##rennen	[v]	0.20115982	familie	[n]	0.07142556
	goal	[n]	0.04652397	verzinne	[v]	0.20004006	langzaam	[adj]	0.06941924
	keer	[n]	0.04621203	##hok	[n]	0.19073038	duister	[adj]	0.06812311
	...			...			...		

Table 5: Top 25 important words per gender in our initial test set by three most promising ranking methods: *TF-IDF [Text]*, *General avg. [Attr.]*, and *TF-IDF [Attr.]*. The female ranking of *General avg. [Attr.]* has negative values as this method constructed a single ranking with negative values pushing towards a female prediction. The middle column of every cell is the word type of that token, either: n (noun), v (verb), adj (adjective), adv (adverb), pp (personal pronoun), or left empty if neither of these collection of five. When a certain word could be of multiple types, the highest occurrence on the online Dutch Van Dale[13] dictionary was chosen (except for ** given the context of the word in the stories). A pp* denotes a personal pronoun which could act as a possessive pronoun according to context.

Female - Male	TF-IDF [Text] - TF-IDF [Text]	0.18242868
	TF-IDF [Attr.] - TF-IDF [Attr.]	0.14744305
	General avg. [Attr.] - General avg. [Attr.]	0.0
Female - Female	TF-IDF [Attr.] - TF-IDF [Text]	0.62934444
	TF-IDF [Attr.] - General avg. [Attr.]	0.34542255
	General avg. [Attr.] - TF-IDF [Text]	0.28372110
Male - Male	TF-IDF [Attr.] - TF-IDF [Text]	0.59165600
	TF-IDF [Attr.] - General avg. [Attr.]	0.47066350
	General avg. [Attr.] - TF-IDF [Text]	0.24156133

Table 6: RBO scores between the three most promising ranking methods in our initial test set: *TF-IDF [Text]*, *General avg. [Attr.]*, and *TF-IDF [Attr.]* (entire ranking). For *General avg. [Attr.]* all positive values were considered male, all negative values female. Parameter $p = 0.991$.

For male-told stories, the model most strongly focussed on (action) verbs and nouns. There was significantly less attention on the remaining three distinguished categories, even though they were largely present in the most distinctive tokens in the text. Solely considering the average attribution score of the tokens, even no adjective, or personal pronoun made the top 25, while the TF-IDF ranking of attribution scores was slightly more diverse. Adventure and fantasy seemed to be the most apparent attended-to domains; domains also present in the TF-IDF ranking of the text, albeit to a lesser extent. Examples of these domains overlap, but included: “kikker” [“frog”], “snijden” [“to cut”], “dwalen” [“to roam”], “leeuw” [“lion”], “tovenaar” [“wizard”], “amulet” [“amulet”], and “prins” [“prince”]. The top 25 of the *TF-IDF [Text]* ranking contained these word types and domains as well, but was significantly more diverse.

For female-told stories, the model had a similarly strong focus on nouns, but also significantly focused on personal pronouns and adjectives. In both model rankings, a clear absence of (action) verbs existed, in stark contrast to the top 25 male-told stories. Here, again fantasy was a present domain, as well as personal relations and feelings; domains that clearly lacked in the male top 25 rankings. Contrarily, adventure was a missing domain clearly present high in the male ranking. Lastly, more explicitly female characters were highly ranked than male characters, something which is not present nor reversely present in male-told stories. These word types and domains seemed accurate as words high in the *TF-IDF [Text]* ranking captured them as well. Clear examples of fantasy included: “prins” [“prince”], “zombie” [“zombie”], and “amulet” [“amulet”]. Clear examples of relations included: “jullie” [“you” [pl] or “your”[pl]], “zus” [“sister”], “familie” [“family”], and “moeder” [“mother”]. Clear examples of feelings included: “verdrietig” [“sad”] and “blij” [“happy”]. Clearly high ranking female characters included: “zus” [“sister”], “vriendin” [“vriendin”], “meisje” [“girl”], and “haar” [“her”].

For both genders we observed a strong focus on their own world of experience, both in our model focus as well as the TF-IDF ranking of the text. This could potentially be related back to Piaget’s stages of development theory which states that children up to age seven (a large portion of our dataset, as can be seen in Table 1) have an inherently egocentric point of view [Mcl24]. However, it should be mentioned that the setup and environment in which the data were collected could have significantly contributed to this. Female examples included: “meisje” [“girl”], “zus” [“sister”], "haar" ["her"], "kind" ["child"], and "hond" ["dog"]. Similarly, male examples included: "ik" ["I"], "mama" ["mom"], "hem" ["him"], and "regenen" ["to rain"]. Moreover, these observed words, and their difference between male and female gender, fit overall nicely in the above-mentioned word types and domains.

5.2.2 Extended test set

A large chance existed that our model learned certain features from the training set that were not included in the initial test set, and thus not visible, or that some features were in fact overrepresented, in the twelve female and twelve male stories our limited size test set contained. Therefore, we ran our model again, but this time with the extended test set as described in Section 4.3.2. Just as in the previous section, we analyze in this section what our model focused on using the three most insightful ranking methods. The results of all seven rankings in full can be accessed via this paper's Github project as described in Appendix A.

Instead of a top 25, we chose to examine the top 50 for our extended test set, given its larger vocabulary and inability to observe certain aspects by solely taking into account the first 25 elements. This top 50 of our three chosen rankings per gender is shown in Table 7. To make comparisons with our initial test set easier, we decided to grayout the last 25 elements.

Again, we first briefly examine the similarities of the rankings by their RBO scores shown in Table 8, before shifting our analysis to word types and domain of the rankings. Compared to our initial test set, we observed a much larger similarity between male(-predicted) and female(-predicted) stories. This made sense as the chance of having tokens only appear in stories of a single gender significantly decreased. Similarly, we observed a significant increase in the similarity of the *TF-IDF [Attr.]* ranking and the *TF-IDF [Text]* ranking in both female(-predicted) and male(-predicted) stories. This showed that, indeed, our initial test set was too limited to show a representative view of our model's attention. Meanwhile, the similarity of the *General avg. [Attr.]* ranking decreased for both TF-IDF rankings. We find it unlikely for this to actually directly hint at an underperformance of our model compared to what we first expected, as this is likely due to the increase in ranking size (due to the increase in unique tokens) and the increase in ranking differences introduced by using a different ranking method. Furthermore, it should be noted again that both ways (*General avg. [Attr.]* and *TF-IDF [Attr.]*) seemed to be good ways of reviewing model focus in their own regard, merely projecting a different perspective as described in Section 4.3.1.

For male-predicted stories (again, we cannot be certain they were actually told by male identifying individuals), we observed the model to focus on the same word types as in the initial test set: a strong focus on verbs and nouns, however, it should be noted that verbs accounted for a much smaller share than previously. Adjectives and adverbs this time had a slight presence, however, a clear absence of personal pronouns remained. The most prominent attended-to domains still included adventure and fantasy. In the top of the *General avg. [Attr.]* ranking, technology related tokens were now noticeable, however, absent in the *TF-IDF [Attr.]* ranking, again, possibly explainable by the aforementioned explanation. Interestingly, attended-to word types and domains both significantly diverged from the ones present in the *TF-IDF [Text]* ranking where in fact many more stopwords were present. Clear examples of adventure and fantasy included: "bruut" ["brute"], "berove" ["to rob"],

	TF-IDF [Text]			General avg. [Attr.]			TF-IDF [Attr.]		
Female	ze	[pp]	1.68224686	meisjes	[n]	-0.67096633	prinses	[n]	5.75576584
	ik	[pp]	1.58729745	tweeling	[n]	-0.49824447	meisje	[n]	2.14687458
	prinses	[n]	1.28401886	stellen	[v]	-0.39640318	moeder	[n]	1.74123065
	mijn	**	1.07525772	prinses	[n]	-0.39158631	heel	[adv**]	1.48277946
	we	[pp]	1.02253068	vanmiddag	[adv]	-0.34612846	mama	[n]	1.35519511
	hij	[pp]	0.96899896	turn	[v]	-0.32432915	ze	[pp]	1.35515830
	dan	[adv]	0.88507214	jurk	[n]	-0.29155831	vulkaan	[n]	1.34874363
	ridder	[n]	0.87520160	astronaut	[n]	-0.28949348	hij	[pp]	1.29266991
	zeggen	[v]	0.85991282	schommel	[n]	-0.27844376	haar	[pp*]	1.25369247
	toen	[adv]	0.85664769	dierentuin	[n]	-0.27523189	hond	[n]	1.23672607
	de		0.83401364	juffrouw	[n]	-0.27462770	ridder	[n]	1.17749621
	hond	[n]	0.80124379	atletiek	[n]	-0.25530997	zus	[n]	1.16282063
	naar	**	0.79146410	stap	[n]	-0.25468876	er	[adv**]	1.15463296
	haar	[pp*]	0.77342037	splitsing	[n]	-0.25077564	was	[v]	0.99584999
	heel	[adv**]	0.75553502	hondje	[n]	-0.24850380	draak	[n]	0.99038232
	hebben	[v]	0.71104281	repareren	[v]	-0.23813254	mijn	**	0.98988052
	jarig	[adj]	0.70186560	bloemen	[n]	-0.22069535	ik	[pp]	0.97771690
	het		0.69697643	zus	[n]	-0.21041863	is	[v]	0.93998505
	mama	[n]	0.69547068	meisje	[n]	-0.20871836	prins	[n]	0.93649408
	einde	[n]	0.68944985	vrienden	[n]	-0.20711009	kind	[n]	0.93248410
	dat		0.67651878	##groep	[n]	-0.20651282	kabouter	[n]	0.93235645
	die		0.66877743	musical	[n]	-0.20188073	we	[pp]	0.86954044
	is	[v]	0.66226125	diploma	[n]	-0.20147535	de		0.84298439
	maar	[adv]	0.65832899	poppen	[n]	-0.19394922	juffrouw	[n]	0.82704904
	gaan	[v]	0.64361193	pizza	[n]	-0.18901573	schommel	[n]	0.78793377
	in	[adv]	0.63964251	vriendin	[n]	-0.18541298	vriendin	[n]	0.78728524
	##en		0.63493847	sprookje	[n]	-0.18517846	het		0.77977045
	was	[v**]	0.63460220	fout	[n]	-0.18307599	jarig	[adj]	0.77300068
	paard	[n]	0.61579098	achtbaan	[n]	-0.18138246	hebben	[v]	0.76453765
	zijn	[pp**]	0.61576752	##hoek	[n]	-0.17746313	einde	[n]	0.74046264
	bos	[n]	0.60808336	verdrinken	[v]	-0.17487174	school	[n]	0.73386183
	draak	[n]	0.60713495	moeder	[n]	-0.17122078	jurk	[n]	0.71978732
	meisje	[n]	0.60471843	klas	[n]	-0.16864280	wolf	[n]	0.70831873
	er	[adv]	0.60298903	digi		-0.16846426	zijn	[pp**]	0.70736853
	op	[adv]	0.60055250	##dubbel		-0.16749758	eten	[v]	0.70153005
	nog	[adv]	0.59827023	grootmoeder	[n]	-0.16668293	eens	**	0.70122666
	wolf	[n]	0.59475544	nul		-0.16538969	waren	[v]	0.69940897
	ook	[adv]	0.59350625	onhandig	[adj]	-0.16402403	zeggen	[v]	0.67677687
	van		0.57521335	onbekend	[adj]	-0.16364890	rijk	[adj]	0.67131798
	kauwgom	[n]	0.57469051	verdrietig	[adj]	-0.16193848	kauwgom	[n]	0.66574609
	met		0.57463695	prins	[n]	-0.16186837	paard	[n]	0.65556225
	##a		0.57087165	bruiloft	[n]	-0.16011644	dat		0.65526951
	niet	**	0.56940392	thee	[n]	-0.15927821	broer	[n]	0.64588767
	willen	[v]	0.56461819	familie	[n]	-0.15918455	heb	[v]	0.64359101
	eten	[v]	0.56186590	set	[n]	-0.15895059	juf	[n]	0.62919646
	huis	[n]	0.55732480	jonkvrouw	[n]	-0.15538579	toen	[adv]	0.60692068
	komen	[v]	0.54475205	speelgoed	[n]	-0.15028885	kopen	[v]	0.59348711
	eens	**	0.54439902	##bank	[n]	-0.14926012	voetballen	[v]	0.59311579
	moeder	[n]	0.54361878	##pri		-0.14905454	dan	[adv]	0.56913928
	je	[pp]	0.54292842	##hulp	[n]	-0.14889461	verhaal	[n]	0.56465383
	...			...			...		
Male	hij	[pp]	1.40450057	bruut	[adj]	0.76319720	hij	[pp]	1.46242540
	ik	[pp]	0.83593589	gezelligheid	[n]	0.74947593	ridder	[n]	1.06148327
	toen	[adv]	0.56323167	glibberig	[adj]	0.72409643	draak	[n]	0.99033221
	ze	[pp]	0.53839865	##blaadje	[n]	0.66991679	zwaard	[n]	0.78820071
	dus	[adv]	0.53120433	##kikker	[n]	0.61795437	bruut	[adj]	0.66151861
	die		0.46315253	amulet	[n]	0.58384508	##kikker	[n]	0.62483532
	zien	[v]	0.46199757	berove	[v]	0.55846868	hebben	[v]	0.56397175
	dat		0.44099834	##robot	[n]	0.54814092	ik	[pp]	0.56307039
	hebben	[v]	0.43322095	schurk	[n]	0.54327435	aflopen	[v]	0.56064174
	maar	[adv]	0.42487051	##stad	[n]	0.52224512	toen	[adv]	0.52578597
	dan	[adv]	0.42310351	##mak		0.51765296	prins	[n]	0.52147030
	##e		0.42279641	begroeten	[v]	0.50811120	die		0.52093955
	komen	[v]	0.41128121	##nslotte		0.44239352	opeens	[adv]	0.51645541
	op	[adv]	0.38950614	##druk	[n]	0.44003354	rijk	[adj]	0.51264001
	in	[adv]	0.38872840	##sporen		0.41303809	maar	[adv]	0.45796420
	hem	[pp]	0.38664172	##hill		0.41222235	##mak		0.44699220
	je	[pp]	0.38525553	waarmee	[adv]	0.40018622	eten	[v]	0.43425279
	de		0.38524072	linkerzij	[n]	0.39921658	amulet	[n]	0.43316685
	zeggen	[v]	0.37217190	##motor	[n]	0.39921434	er	[adv]	0.43070756
	niet	**	0.36959968	vervloekt	[v]	0.38696767	dus	[adv]	0.42436265
	was	[v**]	0.36720728	##geduwd	[v]	0.37165958	willen	[v]	0.40480747
	##n		0.36672626	bas		0.36672019	mama	[n]	0.40005198
	eten	[v]	0.36011801	vijand	[n]	0.36552738	denken	[v]	0.39175106
	met		0.35696469	solo		0.36453815	eens	**	0.38628282
	naar	**	0.34503452	##ater		0.35698606	schat	[n]	0.38369437
	daar	[adv]	0.34176200	##barsten		0.35692185	berove	[v]	0.38031571
	lopen	[v]	0.33635978	sappig	[adj]	0.35352390	ezel	[n]	0.37395896
	te	**	0.33280782	##mink		0.35228781	heb	[v]	0.36856348
	van		0.33195116	kikker	[n]	0.35179389	heel	[adv**]	0.36552238
	politie	[n]	0.32590942	begroe	[v]	0.34976071	kikker	[n]	0.35571113
	ook	[adv]	0.32496779	donder	[n]	0.34419624	meneer	[n]	0.34789544
	willen	[v]	0.32407733	zwaard	[n]	0.33969121	zeggen	[v]	0.34342076
	nog	[adv]	0.32397185	opendoe	[v]	0.33846304	schatkist	[n]	0.33671387
	heel	[adv**]	0.31823361	miljard		0.32674219	verhaal	[n]	0.33209791
	doen	[v]	0.31775714	verrader	[n]	0.32575691	jongen	[n]	0.32290356
	kunnen	[v]	0.31744745	politieauto	[n]	0.32128891	baby	[n]	0.32005101
	we	[pp]	0.31490466	eruitzie	[v]	0.31500337	broek	[n]	0.31943848
	het		0.30805489	oom	[n]	0.31316157	hem	[pp]	0.31930422
	zijn	[pp**]	0.30792521	snijden	[v]	0.31288006	duivel	[n]	0.31922370
	##en		0.30692235	##iem		0.30789082	##hill		0.31843107
	weer	[n]	0.30681950	waarvoor	[adv	0.30371015	kopen	[v]	0.31809874
	vallen	[v]	0.30565307	overheen	[adv]	0.30230482	ze	[pp]	0.31642883
	ridder	[n]	0.30524729	uitgeven	[v]	0.30042289	honderdduizend		0.31153629
	bij	**	0.29889699	broek	[n]	0.29864445	was	[v**]	0.31086306
	al	[adv]	0.29713716	schoppen	[v]	0.29569662	ineens	[adv]	0.30867264
	wat	**	0.29684045	profvoetballer	[n]	0.29404883	vader	[n]	0.30702861
	aan	[adv]	0.29673562	recept	[n]	0.29141720	dan	[adv]	0.30503729
	vliegen	[v]	0.29468225	bevriezen	[v]	0.28974125	dat		0.30402397
	draak	[n]	0.29037451	##saus	[n]	0.28912242	lopen	[v]	0.29805088
	peper	[n]	0.28947834	fort	[n]	0.28867568	duiken	[v]	0.29796523
	...			...			...		

Table 7: Top 50 important words per gender in our extended test set by three most promising ranking methods: *TF-IDF [Text]*, *General avg. [Attr.]*, and *TF-IDF [Attr.]*. Ranks 26-50 are given in gray. Note that the female ranking of *General avg. [Attr.]* has negative values as this method constructed a single ranking with negative values hinting towards an average push towards a female prediction. The middle column of every cell is the word type of that token, either: n (noun), v (verb), adj (adjective), adv (adverb), pp (personal pronoun), or left empty if neither of these. When a certain word could be of multiple types, the highest occurrence on the online Dutch Van Dale[13] dictionary was chosen (except for ** given the context of the word in the stories). A pp* denotes a personal pronoun which could act as a possessive pronoun according to context.

Female - Male	TF-IDF [Text] - TF-IDF [Text]	0.59244143
	TF-IDF [Attr.] - TF-IDF [Attr.]	0.45124364
	General avg. [Attr.] - General avg. [Attr.]	0.0
Female - Female	TF-IDF [Attr.] - TF-IDF [Text]	0.73989223
	TF-IDF [Attr.] - General avg. [Attr.]	0.26745244
	General avg. [Attr.] - TF-IDF [Text]	0.21520131
Male - Male	TF-IDF [Attr.] - TF-IDF [Text]	0.66882254
	TF-IDF [Attr.] - General avg. [Attr.]	0.32286696
	General avg. [Attr.] - TF-IDF [Text]	0.16693228

Table 8: RBO scores between the three most promising ranking methods in our extended test set: *TF-IDF [Text]*, *General avg. [Attr.]*, and *TF-IDF [Attr.]* (entire ranking). For *General avg. [Attr.]* all positive values were considered male, all negative values female. Parameter $p = 0.998$.

"vervloekt" ["cursed"], "ridder" ["knight"], and "draak" ["dragon"]. Clear examples of technology related tokens included: "#robot" ["#robot"], "#motor" ["#motor"], and "politieauto" ["police car"].

A larger difference in word type and domain existed for female-predicted stories. Still, a strong focus was present on nouns, some focus on personal pronouns, and explicitly close to no focus on verbs. Unlike our initial test set, very little explicit focus on adjectives seemed to exist, all while the *TF-IDF [Text]* ranking was significantly more diverse than before. Fantasy and personal relations were again highly attended-to domains, equally present in the top-ranked TF-IDF tokens of the text. Additionally, explicit female characters were again more represented in the top 25 and top 50 than male characters, something only existent to a much lesser extent in the *TF-IDF [Text]* ranking. This attention to relational domains and female characters was strongly absent in the top 25 for male-predicted stories. Surprising was the total lack of tokens directly relating to feelings in the top 25 and even the entire top 50, something which was uniquely present in our original test set, but seemingly not representative for larger datasets. Examples of fantasy related tokens included: "prinses" ["princess"], "ridder" ["knight"], "draak" ["dragon"], "prins" ["prince"], and "kabouter" ["gnome"]. Examples of tokens in the relational domain included: "juffrouw" ["teacher" [fem.]], "vrienden" ["friends"], "vriendin" ["girlfriend"], "moeder" ["mother"]. Examples of tokens explicitly mentioning female characters, additionally to the ones mentioned in previous examples, included: "mama" ["mom"], "grootmoeder" ["grandmother"], and "jonkvrouw" [female rank of nobility, often used in fairy tales].

Again, for both genders we observed a strong focus on their own world of experience, similar to the initial test set.

6 Discussion

Having considered our experiments and results above, we further analyze the outcomes of these results in this section with the aim of finding an answer to our research question: "How successful can a finetuned BERT model distinguish informal Dutch speech between boys and girls, what parts of language does it focus on, and can this provide insight into differences between (binary) genders?". We discuss an answer to the first part of this question in Section 6.1 where we zoom in on the model's performance (i.e. an answer to "How successful can a finetuned BERT model distinguish informal Dutch speech between boys and girls [...] ?"). The second part of the question is discussed in Section 6.2 where we zoom in on the model's focus (i.e. an answer to "[...] What parts of language does [the model] focus on [...]?"). Lastly, the last part of the question is answered in Section 6.3 where we discuss its generalizability and if our findings can provide any further insight into differences in language use between (binary) genders (i.e. an answer to "[...] Can this provide insight into differences in language use between (binary) genders?").

6.1 Model performance

Our model performed over the entire test set of 24 stories (12 male, 12 female) using 10-fold cross validation on average with an F1-score of 0.56 and an accuracy of 0.63. This average F1-score and accuracy is significantly better than random guessing, but is not outstandingly impressive either. However, we have reason to believe this modest result might not be limited by the capabilities of the model itself, but instead by the limited size dataset.

Comparable research with a similar goal to binarily identify author gender from text using ML was conducted by Cheng et al. [CCS11]. They showed performance with accuracies of 76.75% and 82.23% for the two datasets they used using Support Vector Machines. Notable is that their datasets respectively contained 810,000 and just over 500,000 samples. On the contrary, other research by Khan et al. [Kha+23] showed even worse performance than us with an accuracy between 48% and 63% depending on the ML technique they used. Notable here is their use of only 1,000 data samples. While other factors could be additionally at play, a possible explanation for this difference between the two researches similar to ours would be one in line with general knowledge regarding ML: dataset size plays a significant role in model performance. This contributes to our earlier suspicion of dataset size being a limiting factor in performance. This would also explain the big performance difference between the individual seeds and folds; with the limited size of the training set, the split of the training-validation and test set can lead to a major (dis)advantage in actual and perceived performance. A larger dataset could lead to a more consistent and robust training set, consequently leading to a more robust model. We discuss this possible future improvement further in Section 7.

To answer the first part of the research question we like to conclude: our model performed on average with an F1-score of 0.56 and an accuracy of 0.63, and a top performance of an F1-score of 0.79 and an accuracy of 0.79. While this is proof a finetuned BERT model *can* be this successful with the aforementioned results on a test set, given prior research and the limited size of our dataset we have reason to believe these numbers have the potential to be higher.

6.2 Model focus

The observations of our model's focus are summarized in Table 9.

Using our initial test set we performed a manual analysis on the model's focus regarding highly ranked word types and domains. For male-told stories, we concluded that the model has a particularly strong focus towards (action) verbs and nouns; an observation significantly less present in the

	Word types	Domain
Male		
Initial	– (Action) verbs – Nouns	– Adventure – Fantasy
Extended	– (Action) verbs (much less) – Nouns – Adjectives (much less) – Adverbs (much less)	– Adventure – Fantasy – Technology
Female		
Initial	– Nouns – Personal pronouns – Adjectives	– Fantasy – Personal relations – Feelings – Explicit female characters
Extended	– Nouns – Personal pronouns (much less)	– Fantasy – Personal relations – Explicit female characters

Table 9: Summarized observations of manual analysis on attended-to word types and domains using the finetuned model's attribution scores.

TF-IDF ranking of solely the text (without attribution weights). The same goes for the observed domains: adventure and fantasy. After extending the test set with the remaining unlabeled stories, our observations stayed similar, however slightly changed. This time mainly nouns were highly ranked, with (action) verbs, adjectives, and adverbs to a lesser extent. Just as in the initial test set, the domains of adventure and fantasy were most strongly attended to. However this time, technology was a highly ranked domain not strongly attended to previously. The fantasy domain being highly present in both is not surprising due to the nature of the dataset.

For female-told stories in the initial test set, we concluded that the model most strongly focused on nouns, with an additional significant focus on personal pronouns and adjectives. Verbs were nearly absent (in stark contrast to the male stories). The discussed domains that mostly ranked high were fantasy, personal relations, and feelings. Generally, explicit female characters ranked significantly higher than explicit male characters, something not present nor reversely present in male stories. After the extension of the test set by unlabeled stories, we still observed near equal attention on nouns and to a lesser extent on personal pronouns. This time, the attention on adjectives was nearly absent. The highly ranked domains stayed equal to the ones found in our initial test set, with the exception of feelings; surprisingly no tokens relating to feelings were present. Again, the high ranking of the fantasy domain can be explained by the nature of the dataset, hence we do not interpret this as a gender-specific domain.

Clearly observable in the extended test set (see Table 7) for both genders is that the *General avg.*

[Attr.] ranking is much more expressive in showing the preference for nouns and verbs than the *TF-IDF [Attr.]* ranking which contains more diverse word types and significantly more stopwords. This difference in ranking shows that the model overall puts most attention to single nouns and verbs, however, since most do not occur in many stories, more often occurring tokens like adverbs, personal pronouns, and adjectives can still be found important specifically for the collection of male predicted documents (as this is what the TF-IDF value shows) as they often occur in many documents. While we cannot prove this specific focus helped the model to obtain better performance we expect it to, as nouns and verbs can be seen as more content bearing than stopwords. An additional observation that might be able to be explained by the same reasoning is the larger variability on top of the *TF-IDF [Attr.]* ranking, while the tokens in place 26 till 50 are much more monotonously nouns and verbs.

An interesting observation is even broader than plainly considering word types. In the initial test set, most word types and domains had a similar presence both in the top of the attribution rankings and the *TF-IDF [Text]* ranking (albeit often to a lesser extent in the text ranking). This is different in the extended test set: the word types of the *TF-IDF [Text]* ranking are highly diverse, mostly consisting of stopwords instead of content rich words. While this might be largely caused by our choosing of this ranking method, it is interesting to observe that the attribution rankings contain much more focus on nouns for both genders, affirming that our finetuned model indeed extra focuses on what an average human might consider to be relevant and content rich words. Notable is that all non-stopwords in the *TF-IDF [Text]* ranking of the extended test set follow the same trend as the initial test set; the domains high in the attribution rankings are equally present in the *TF-IDF [Text]* ranking, albeit to a lesser extend.

To answer the second part of the research question we like to conclude: our model largely considers the presence of verbs, nouns, and adverbs in domains of adventure and technology as indicators of a male-told story, while it considers the presence of nouns and personal pronouns in the domain of personal relations as well as tokens relating to explicit female characters as indicators of a female-told story. These word types and domains occurred similarly high in the TF-IDF ranking (aggregated per gender) of the texts themselves, however, this contained much more variability and diversity. This affirms our belief that our model truly more strongly focuses on these indicators.

6.3 Generalizability

In our experimentation, we proved that a BERT model can be successfully finetuned –with relatively high accuracy– to predict a storyteller's gender. Therefore, the clearly differentiating word types and domains the model focused on seem promising gender-specific features, assuming it accomplished this by paying attention to relevant parts of speech for gender and not by proxies, such as confounding contextual variables, or plain coincidence. Whether this method successfully provides any insight into truly existing language differences between (binary) genders can be determined by consulting relevant research and comparing findings. As our research context is unique in focusing on natural Dutch children's speech, a direct comparison is not possible hence we must make our comparison based on more general research. We acknowledge this is a significant limitation as gender behavior is highly culture and context dependent [BP19] [KC17].

Mulac and Lundell [ML86] analyzed oral descriptions of landscapes for linguistic variables between genders. They found that indeed objectively coded language features can be used to accurately differentiate a storyteller's gender. Similar to us, they found the extended use of personal pronouns an indicator for female language use. However, contrary to us, they found (intensive) adverbs to be indicative for female language use, something which we contributed to males. All their other female features were ones we did not consider classifying, nor were the ones for male features (such as the use of impersonals and elliptical sentences). Bischoping [Bis93] did specific research into gender differences of conversation topics in freely recorded conversations by undergraduate students. While it must be stated that the topic categories they investigated seemed to have their pre-existing biases incorporated into them ("people & relationships", "work & money", "leisure activity", "appearances", and "issues" which encaptured serious current events and politics), they had overwhelmingly clear observations in these categories. They found a consistent preference for females to discuss people, relationships, and appearances, while males preferred topics about work, money, leisure, and serious current events and politics. These topics are in line with the findings of our model.

Newman et al. [New+08] performed one of the most rigorous studies to-date comparing over 14,000 text files including transcribed conversations. In speech, they found a similar difference in (personal) pronoun use with higher recordings in female authors, however, their higher recording of adverbs for female participants is an aspect absent in our research. Additionally, they found a focus in female language discussing other people, thoughts, and emotions, something which could fall in our domain of personal relations and feelings, although these domains were not explicitly used in this research. Furthermore, they found (motion) verbs to be more popular under male authors, similar to our research. Our male prominent domains of adventure and technology were not specifically mentioned, and therefore hard to affirm. However, speech relating to sports, external processes, and objects were more prominently found in male speech, categories one might place within adventure and technology.

These general pre-existing researches, along with others ([Coo+85] [Haa79]), seem promising and in line with the results of our BERT model. However, while these seem to prove the potential of our approach, it would be unfair not to mention a critical side note to this. As Koolen and Cranenburgh [KC17] mention, results in gender related research are easily overgeneralized due to interpretation using existing gender stereotypes, and many researchers are not always aware of possible confounding variables related to gender. Similarly, Leaper [Lea14] argues for the usually negligible or small magnitude of differences when contextual variables are taken into account. Even more extreme is the research of Brouwer et. al [BGH79]. They criticize that the greater part of research on differences in language use by gender is conducted with data collected in unreliable ways, and without attention to confounding variables. In their own research they did not find any statistically significant results between language used by male and female participants. However, they did find a statistically significant difference in language use depending on sex of the addressee, something they point out as a potentially confounding variable rarely taken into consideration in prior research, again reinforcing their prior claim of lack of attention to confounding variables.

Ultimately, we are positive about the generalizability of our approach as our results are in line with earlier research that found differences between male and female language use. This gives strong reason to believe that one can indeed successfully finetune a BERT model to predict a storyteller's gender with relatively high accuracy and then examine what parts of language it focuses on as a way to gain insight into differences in language use between the (binary) genders, affirming our research question. However, we do acknowledge the possibility of unaccounted confounding and contextual variables that influenced the data collection between genders as part of a wider criticism in research on gender differences.

7 Future work

During the trajectory of our research, we came across certain limitations and shortcomings which should be addressed. The most notable potential points of improvement and opportunities for future research are discussed below.

Improved model selection In hindsight and upon closer evaluation of our research, we must acknowledge a slight flaw in our model selection method. During hyperparameter selection, we consistently evaluated the model's performance upon its test set F1-score after which we chose the hyperparameters corresponding to our best performing model to be the most optimal. After hyperparameter selection, we went on to more rigorously test models with these hyperparameters using 10-fold cross validation where we decided upon considering the model corresponding to the best performing iteration –according to the test set's F1-score– for our further model focus analysis. Unconsciously this led us to implicitly use the test set for model optimization and selection leading to a form of data leakage. In further research, extra attention to this common mistake [KN23] of neglectance should be given.

Consult linguist professionals We believe the consultation of professional linguists could reap great benefits for the analysis on the rankings of the model. As we are no professional linguists ourselves, we did our best to extract conclusive and complete insights, however, given our mere modest linguistic expertise, we are aware we might have overlooked insightful nuances obvious to linguistic professionals. This might lead to a more sound answer to the second part of our research question (the parts of language our model focuses on) which in turn can lead to a better explanation of what the differences in (binary) gender language use are.

More robust hyperparameter tuning One way to obtain better model performance is to tune the training hyperparameters; we divided this process into two parts. In the first part, we performed manual *ad hoc* search as the potentially optimal hyperparameter values spanned a very large search space, the search space of the second part was smaller, which opened the opportunity to perform a more extensive grid search. By either allocating more time or resources for this process, a more robust grid search method for all hyperparameters could potentially yield more optimal hyperparameters for our model. Alternatively, newly emerging hyperparameter finding techniques such as Bayesian Search might be worth investigating given their promising results with limited computational costs, with some even able to search continuous search spaces eliminating the need for discrete deltas that could lead to overshooting an optimum [Sol23].

Larger (labeled) dataset The rather limited average model performance and large variability in performance between iterations with 10-fold cross validation hints at a large variability between folds and choice of seed. This attributes to our suspicion of the dataset size being a limiting factor.

This reasoning can be justified by the fact that a more limited dataset significantly increases the (dis)advantage of choosing a particular fold as test fold (and the others for training-validation) as the interfold differences are likely to be larger. A larger test is likely to reduce these interfold differences both in content and, consequently, model performance. Additionally to gaining more consistent model performance measures, a larger dataset would evidently equally increase model performance and robustness by capturing a more complete picture of language differences. Apparent is the importance of the presence of gender labels in this dataset to eliminate the disadvantages introduced by our labelless extended set which could solely be used for testing, discussed in Section 4.3.2. As an additional note, we like to stress the importance of confounding and contextual variables in such an enlarged labeled dataset as discussed in Section 6.3. While it is impossible to include a comprehensive set of (possible confounding and contextual) variables, and like to remark the inclusion of many in the ChiSCor dataset, possible additional ones such as those mentioned in previous papers like time of day and gender of addressee might be worth considering.

Influence of age differences Unconscious biased conclusions can be introduced when not accounting for certain confounding and contextual variables, as pointed out by Koolen and Cranenburgh [KC17]. In our research, we assume to have finetuned a model predicting a storyteller's gender with relatively high accuracy, however, we should beware of other variables that highly influence language use which the model might have focused on. One of such variables is the storyteller's age. Upon examining the age and gender distribution of our initial dataset (see Figure 1), we observe a skew in age with participating boys being generally younger than participating girls: the average age of participating girls is 7.01 years, the average age of participating boys is 6.86 years. This could lead our model to, in fact, not predict a storyteller's gender but rather age as a proxy for gender. While we have to admit this average age difference of only 2 months likely has limited influence, this question remains one for potential further research.

Double label use Our model's task was one of binary classification; something which is theoretically doable by a single output label and introduction of a threshold value. As the labels in the original dataset were zeros and ones for female-told and male-told stories respectively, we decided upon choosing this threshold at 0.5 with interpreting all outputs below as female and above as male. However, as could be seen from the visualization of the integrated gradients technique in Figure 1, this exposes a problem most clearly revealed when providing a sequence of `[PAD]` tokens to the model: the output is always near 0.0 (with the exact value dependent on sequence length). In other words, by designing the model the way we did, using only a single label and setting a threshold after which output is interpreted as male, we created a model that effectively determines its output by the sufficient presence of male-related tokens, otherwise female is predicted. This observation falls in line with the observation in our experiments where consistently *precision* > *recall* (meaning the model indeed more often falsely predicted zeros [female] than ones [male], see Appendices C, D, and E). Using two labels instead might mitigate this issue and lead to a more robust model, where every label represents the probability of being told by that gender. This, however, remains a question for further research.

8 Conclusion

In this paper, we proposed the idea of analyzing differences in language use between (binary) genders in a novel way using AI. We decided not to analyze the text data itself directly, but instead finetune a BERT model to predict a storyteller's gender, after which we examined what this model focused on. The reasoning was that if an AI model can determine a storyteller's gender with sufficient accuracy, it must focus on significant parts of language more than insignificant ones to accomplish this accuracy. Examining this model focus led to specific word types and domains that differed per gender, effectively providing insight into differences in language use between (binary) genders.

For this research, we used the ChiSCor dataset, consisting of informal fantasy stories freely told by children at their elementary school. As not all stories of the dataset contained information about the storyteller's gender, 240 stories by 145 unique children remained fit to train and initially test our model.

We started by focusing on the first part of our research question: "How successful can a finetuned BERT model distinguish informal Dutch speech between boys and girls[. . .]?". For this, we repeatedly finetuned the Dutch BERTje model's parameters on the acquired dataset and further investigated optimal training hyperparameters. After deciding upon this configuration, we used 10-fold cross validation to more rigorously test the performance of this configuration while avoiding data leakage. We made use of stratified folds in order to get a representable result each iteration. Running the 10-fold cross validation gave us the average performance results of an average F1-score of 0.56 and an average accuracy of 0.63. While the model's average performance scores were significantly better than random guessing, they were not outstandingly impressive either. However, since we asked how successful a finetuned BERT model *can* be, we also considered the top performing iteration with an F1-score of 0.79 and an accuracy of 0.79. This large performance variation was likely caused by the limited size dataset as this creates relatively large variations per fold which can lead to major (dis)advantages regarding train and test sets. We decided upon continuing the remainder of our research with this best performing model (all k folds of k-fold cross validation produce a slightly different model); it was the most successful in making the desired gender prediction of the storyteller in the test set (assuming it accomplished this by paying attention to relevant parts of its speech input for gender and not by proxies, such as confounding contextual variables, or plain coincidence).

For the second part of our research question, "[...] What parts of language does [the model] focus on [...]?", we made use of the integrated gradients technique which created a unique attribution score per token occurrence. Since we wanted a single value per token to create a ranking for further analysis, we constructed five different ways of combining all attribution scores of a single token of which we found two insightful ones. We then manually analyzed and compared these rankings, focusing on highly ranked word types and domains per gender. Only considering the limited size initial test set with known ground truth genders (consisting of only 24 stories) had the two large advantages of knowing the model's accuracy and hinting towards tokens that the model attributed to the wrong gender, however, gave rise to the problem of likely not fully capturing the model's focus. Therefore, we introduced an extended test set that included the earlier discarded stories without gender data, which had its own inherent problems, but in combination did provide additional insights into our model's focus. Manual visual analysis on the top 25 and top 50 most important tokens for the initial test set and extended test set respectively led to some clear observations. Differences between ranking methods exposed some insights into the model's focus that were present for both genders. Firstly, in the extended test set, we observed that nouns and verbs were consistently ranked much higher by our model by average attribution score compared to the adjusted TF-IDF score by attribution. This showed that on a per token level, the model put the most attribution weight on these word types, however, since most of these only occurred in few stories, more often occurring tokens were still found more important in the collection of a certain gender (being the definition of the *TF-IDF [Attr.]* ranking). The fact that slightly lower ranked tokens in the *TF-IDF [Attr.]* similarly were mostly nouns and verbs reaffirms this belief; the nouns and verbs were individually the most attributed tokens, however, they simply did not occur enough to be found most important. Secondly, in the initial test set, we observed a great similarity between both attribution rankings and the *TF-IDF [Text]* ranking regarding both word types and domains. Contrarily, in the extended test set, we observed the attribution rankings to contain many more nouns and verbs than the more diverse *TF-IDF [Text]* ranking which was significantly richer in word types and contained more stopwords. While we could not prove this specific focus of nouns and verbs over other word types and stopwords helped the model to obtain better performance, we expected it to, as nouns and verbs could be seen as more content bearing than stopwords. Differences between the genders were furthermore clearly present. For male stories, we observed the model to extra focus on tokens that were verbs and nouns in the domains of adventure and technology. Differences between test sets included a stronger focus on verbs in the initial test set than the extended test set, while the slight focus on adjectives and the domain of technology present in the extended test set was absent in the initial one. For female stories, we observed the model to extra focus on tokens that were nouns and personal pronouns in the domain of personal relations as well as extra focus on explicit female characters. Here, differences between test sets included a focus on adjectives and the domain of feelings in the initial test set, both absent in the extended test set.

We compared all our findings to relevant research on differences in language use between (binary) genders to examine whether our method successfully provided any insight into truly existing language differences. As no other research known to us was performed on natural Dutch children's speech, we diverted to more general research, which is a limitation as gender behavior is highly culture and context dependent. Still, the results our approach produced gave reason to be positive about the generalizability as, in fact, they were in line with earlier conducted research. This gives strong reason to believe one can indeed successfully finetune a BERT model to predict a storyteller's gender with relatively high accuracy and then examine what parts of language it focuses on as a way to gain insight into differences in language use between the (binary) genders, affirming our research question.

It should be noted that our research was still imperfect and open for future improvements and research. The most notable points of further research we recognized were: improved model selection, consulting linguist professionals, performing more robust hyperparameter tuning, using a larger (labeled) dataset with sufficient attention to confounding contextual variables, investigating the influence of age differences between genders on our research, and using double labels for the model output.

In summary, to answer our research question "How successful can a finetuned BERT model distinguish informal Dutch speech between boys and girls, what parts of language does it focus on, and can this provide insight into differences in language use between (binary) genders?". The finetuned BERT model we constructed in this paper performed with an average F1-score of 0.56 and an average accuracy of 0.63, however, recorded a top performance of an F1-score of 0.79 and an accuracy of 0.79. Given our own research and relevant prior research, we have reason to believe that the dataset size was a limiting factor and these numbers have the potential to be higher. To examine the model's focus we could extend the test set by reincluding earlier discarded stories without gender data. For male stories, the model extra strongly focussed on tokens representing verbs, nouns, and adverbs in the domains of adventure and technology. For female stories, the model extra strongly focussed on tokens representing nouns, personal pronouns, and adjectives in the domains of personal relations, feelings, and explicit female characters. Similar pre-existing relevant research into differences in language use between (binary) genders show results in line with our findings. This gives strong reason to believe that one can indeed successfully finetune a BERT model to predict a storyteller's gender with relatively high accuracy and then examine what parts of language it focuses on as a way to gain insight into differences in language use between the (binary) genders, affirming our research question.

References

[Bak14] Paul Baker. "Using corpora to analyze gender". In: London; New York: Bloomsbury, 2014, p. 30. ISBN: 9781441110589.

[BGH79] Dédé Brouwer, Marinel Gerritsen, and Dorian De Haan. "Speech Differences between Women and Men: On the Wrong Track?" In: *Language in Society* 8.1 (1979), pp. 33–50. ISSN: 00474045, 14698013. URL: http://www.jstor.org/stable/4167038.

[Bis93] Katherine Bischoping. "Gender differences in conversation topics, 1922–1990". In: *Sex Roles* 28.1 (Jan. 1993), pp. 1–18. ISSN: 1573-2762. DOI: 10.1007/BF00289744. URL: https://doi.org/10.1007/BF00289744.

[BP19] Deborah L. Best and Angelica R. Puzio. "Gender and Culture". In: *The Handbook of Culture and Psychology*. Oxford University Press, July 2019. Chap. 9, pp. 235–291. ISBN: 9780190679743. DOI: 10.1093/oso/9780190679743.003.0009.

[But90] Judith Butler. *Gender trouble: feminism and the subversion of identity*. Thinking gender. New York: Routledge, 1990. ISBN: 9780415900423.

[CCS11] Na Cheng, Rajarathnam Chandramouli, and K. Subbalakshmi. "Author gender identification from text". In: *Digital Investigation* 8.1 (July 2011), pp. 78–88. DOI: 10.1016/j.diin.2011.04.002.

[Coo+85] Alicia Skinner Cook et al. "Early gender differences in the functional usage of language". en. In: *Sex Roles* 12.9 (May 1985), pp. 909–915. ISSN: 1573-2762. DOI: 10.1007/BF00288093.

[Cra21] Kate Crawford. *Atlas of AI: Power, Politics, and the Planetary Costs of Artificial Intelligence*. Yale University Press, Apr. 2021. Chap. 1. ISBN: 9780300252392. DOI: 10.12987/9780300252392.

[Dev+19] Jacob Devlin et al. *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*. 2019. DOI: 10.48550/arXiv.1810.04805.

[Dij+23] Bram M. A. van Dijk et al. *ChiSCor: A Corpus of Freely Told Fantasy Stories by Dutch Children for Computational Linguistics and Cognitive Science*. 2023. DOI: 10.48550/arXiv.2310.20328.

[FKS02] Ronald Fagin, Ravi Kumar, and D. Sivakumar. "Comparing Top k Lists". In: *SIAM Journal on Discrete Mathematics* 17.1 (Oct. 2002), pp. 134–160. DOI: 10.1137/S0895480102412856.

[Fos+21] Eduard Fosch-Villaronga et al. "A little bird told me your gender: Gender inferences in social media". en. In: *Information Processing Management* 58.3 (May 2021), p. 102541. ISSN: 03064573. DOI: 10.1016/j.ipm.2021.102541.

[Haa79] Adelaide Haas. "Male and female spoken language differences: Stereotypes and evidence". In: *Psychological Bulletin* 86.3 (1979), pp. 616–626. ISSN: 1939-1455. DOI: 10.1037/0033-2909.86.3.616.

[HWN12] Yakun Hu, Dapeng Wu, and Antonio Nucci. "Pitch-based gender identification with two-stage classification". In: *Security and Communication Networks* 5.2 (Feb. 2012), pp. 211–225. DOI: 10.1002/sec.308.

[Jon94] Karen Sparck Jones. “Natural Language Processing: A Historical Review”. In: *Current Issues in Computational Linguistics: In Honour of Don Walker*. Ed. by Antonio Zampolli, Nicoletta Calzolari, and Martha Palmer. Dordrecht: Springer Netherlands, 1994, pp. 3–16. ISBN: 978-0-585-35958-8. DOI: 10.1007/978-0-585-35958-8_1.

[KC17] Corina Koolen and Andreas van Cranenburgh. “These are not the Stereotypes You are Looking For: Bias and Fairness in Authorial Gender Attribution”. In: *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*. Ed. by Dirk Hovy et al. Valencia, Spain: Association for Computational Linguistics, Apr. 2017, pp. 12–22. DOI: 10.18653/v1/W17-1602.

[Kha+23] Muhammad Hood Khan et al. “Author’s Age and Gender Prediction on Hotel Review Using Machine Learning Techniques”. In: *Journal on Big Data* 5.1 (July 2023), pp. 41–56. ISSN: 2579-0056. DOI: 10.32604/jbd.2022.044060.

[KN23] Sayash Kapoor and Arvind Narayanan. “Leakage and the reproducibility crisis in machine-learning-based science”. In: *Patterns* 4.9 (Aug. 2023). ISSN: 2666-3899. DOI: 10.1016/j.patter.2023.100804.

[Lea14] Campbell Leaper. “Gender similarities and differences in language”. In: *The Oxford handbook of language and social psychology.* Oxford library of psychology. New York, NY, US: Oxford University Press, 2014, pp. 62–81. ISBN: 9780199838639. DOI: 10.1093/oxfordhb/9780199838639.013.002.

[Mcl24] Saul Mcleod. *Piaget’s Preoperational Stage (Ages 2-7)*. Jan. 2024. URL: https://www.simplypsychology.org/preoperational.html.

[ML86] Anthony Mulac and Torborg Louisa Lundell. “Linguistic Contributors to the Gender-Linked Language Effect”. In: *Journal of Language and Social Psychology* 5.2 (1986), pp. 81–101. DOI: 10.1177/0261927X8652001.

[Mol22] Christoph Molnar. “Shapley Values”. In: *Interpretable Machine Learning: A Guide For Making Black Box Models Explainable*. 2022. Chap. 9. URL: https://christophm.github.io/interpretable-ml-book/shapley.html.

[New+08] Matthew Newman et al. “Gender Differences in Language Use: An Analysis of 14,000 Text Samples”. In: *Discourse Processes* 45 (May 2008), pp. 211–236. DOI: 10.1080/01638530802073712.

[SA22] G.U. Shagi and S. Aji. “A machine learning approach for gender identification using statistical features of pitch in speeches”. In: *Applied Acoustics* 185 (Jan. 2022), p. 108392. DOI: 10.1016/j.apacoust.2021.108392.

[Sch20] Lenhart Schubert. “Computational Linguistics”. In: *The Stanford Encyclopedia of Philosophy*. Ed. by Edward N. Zalta. Spring 2020. Metaphysics Research Lab, Stanford University, 2020.

[Sol23] Ali Soleymani. *Grid search and random search are outdated. This approach outperforms both.* Feb. 2023. URL: https://medium.com/@ali.soleymani.co/stop-using-grid-search-or-random-search-for-hyperparameter-tuning-c2468a2ff887.

[STY17] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. “Axiomatic Attribution for Deep Networks”. In: arXiv:1703.01365 (June 2017). DOI: 10.48550/arXiv.1703.01365.

[Vig22] Jesse Vig. *Deconstructing BERT, Part 2: Visualizing the Inner Workings of Attention*. en. Apr. 2022. URL: https://towardsdatascience.com/deconstructing-bert-part-2-visualizing-the-inner-workings-of-attention-60a16d86b5c1.

[Vri+19] Wietse de Vries et al. *BERTje: A Dutch BERT Model*. Dec. 2019. DOI: 10.48550/arXiv.1912.09582.

[Wei+21] Laura Weidinger et al. *Ethical and social risks of harm from Language Models*. Dec. 2021. arXiv: 2112.04359 [cs.CL].

[Who97] Benjamin Lee Whorf. "The Relation of Habitual Thought and Behavior to Language". In: *Sociolinguistics: A Reader*. Ed. by Nikolas Coupland and Adam Jaworski. London: Macmillan Education UK, 1997, pp. 443–463. ISBN: 978-1-349-25582-5. DOI: 10.1007/978-1-349-25582-5_35.

[WMZ10] William Webber, Alistair Moffat, and Justin Zobel. "A similarity measure for indefinite rankings". In: *ACM Trans. Inf. Syst.* 28.4 (Nov. 2010). ISSN: 1046–8188. DOI: 10.1145/1852102.1852106.

Appendices

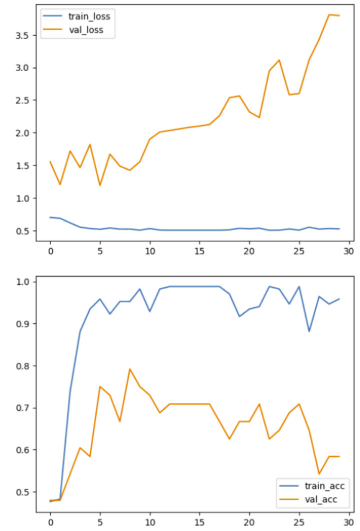
A Relation paper sections & code files

Section in paper	Codefile in Github
Data processing (Section 4.1) Model finetuning (Section 4.2.1) Hyperparameter tuning (Section 4.2.2)	101, 102
K-fold cross validation (Section 4.2.3)	103
Model focus (Section 4.3)	104
Initial test set (Section 4.3.1)	105
Extended test set (Section 4.3.2)	106

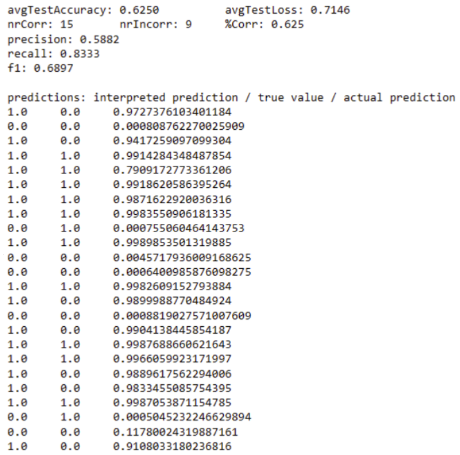
Table 10: Overview of discussed topics in the Methodology with their relating Github files.

Codefiles can be accessed via this paper's Github project, accessible at: github.com/sanderhonig/scriptie

B Example of graph and test set output during hyperparameter tuning



(a) Two plotted graphs: upper containing the average training and validation loss, lower containing the average training and validation accuracy. Each data point represents one epoch.



```
avgTestAccuracy: 0.6250          avgTestLoss: 0.7146
nrCorr: 15        nrIncorr: 9    %Corr: 0.625
precision: 0.5882
recall: 0.8333
f1: 0.6897

predictions: interpreted prediction / true value / actual prediction
1.0     0.0     0.9727376103401184
0.0     0.0     0.000808762270025909
1.0     0.0     0.9417259097099304
1.0     1.0     0.9914284348487854
1.0     1.0     0.7909172773361206
1.0     1.0     0.9918620586395264
1.0     1.0     0.9871622920036316
1.0     1.0     0.9983550906181335
0.0     1.0     0.000755060464143753
1.0     1.0     0.9989853501319885
0.0     0.0     0.0045717936009168625
0.0     0.0     0.0006400985876098275
1.0     1.0     0.9982609152793884
1.0     0.0     0.9899988770484924
0.0     0.0     0.0008819027571007609
1.0     0.0     0.9904138445854187
1.0     1.0     0.9987688660621643
1.0     1.0     0.9966059923171997
1.0     0.0     0.9889617562294006
1.0     0.0     0.9833455085754395
1.0     1.0     0.9987053871154785
0.0     1.0     0.0005045232246629894
0.0     0.0     0.11780024319887161
1.0     0.0     0.9108033180236816
```

(b) Test set performance output.

Figure 3: Example of graph and test set output during hyperparameter tuning to assess the quality of finetuning and effect of the current hyperparameters[14].

[14] Taken from second stage of hyperparameter tuning, settings: lemmatization (no), number of epochs (30), and batch size (2).

C Results of first stage of hyperparameter tuning

Lem.	Learn. rate	W. decay	Dropout	Accuracy*	Loss*	Precision*	Recall*	F1-score*	Avg. f1-score
no	0.000010	0.01	0.35	0.5000	0.7024	0.0000	0.0000	0.0000	
no	0.000010	0.01	0.25	0.5000	?[15]	0.0000	0.0000	0.0000	
no	0.000020	0.01	0.25	0.6667	0.6740	0.7000	0.5833	0.6364	
no	0.000015	0.01	0.25	0.5000	0.6926	0.0000	0.0000	0.0000	
no	0.000025	0.01	0.25	0.7083	0.6426	0.7273	0.6667	0.6957	
no	0.000025	0.006	0.25	0.7083	0.6741	0.6471	0.9167	0.7586	0.3418
				0.5417	0.6776	0.6667	0.1667	0.2667	
				0.5000	0.6942	0.0000	0.0000	0.0000	
no	0.000025	0.01	0.20	0.5417	0.6605	1.0000	0.0833	0.1538	
no	0.000025	0.01	0.30	0.5000	0.6919	0.0000	0.0000	0.0000	
no	0.000025	0.01	0.15	0.7083	0.7087	0.6316	1.0000	0.7742	0.2581
				0.5000	0.6909	0.0000	0.0000	0.0000	
				0.5000	0.6956	0.0000	0.0000	0.0000	
no	0.000030	0.01	0.15	0.5833	0.7160	0.5714	0.6667	0.6154	**0.6157**
				0.7083	0.6465	0.8571	0.5000	0.6316	
				0.5000	0.7677	0.5000	0.7500	0.6000	
no	0.000030	0.01	0.22	0.4583	0.7638	0.4706	0.6667	0.5517	
no	0.000030	0.01	0.25	0.5000	0.7012	0.0000	0.0000	0.0000	
no	0.000025	0.01	0.20	0.5000	0.7107	0.5000	0.3333	0.4000	0.1778
				0.4583	0.6981	0.3333	0.0833	0.1333	
				0.3750	0.7157	0.0000	0.0000	0.0000	
no	0.000025	0.009	0.25	0.5417	0.7156	0.5333	0.6667	0.5925	
no	0.000025	0.02	0.20	0.5833	0.6998	0.6250	0.4167	0.5000	
no	0.000025	0.015	0.15	0.7083	0.6510	0.6471	0.9167	0.7586	0.4838
				0.6250	0.6576	0.8000	0.3333	0.4706	
				0.4167	0.7261	0.3333	0.1667	0.2222	

Table 11: Intermediary test set results of *ad hoc* hyperparameter tuning process using the non-lemmatized dataset. All ran using 8 epochs and a batch size of 4. *average over all batches of the test set.

Lem.	Learn. rate	W. decay	Dropout	Accuracy*	Loss*	Precision*	Recall*	F1-score*	Avg. f1-score
yes	0.000010	0.015	0.35	0.5000	0.6984	0.0000	0.0000	0.0000	
yes	0.000010	0.015	0.25	0.7083	0.6465	0.8571	0.5000	0.6316	**0.6457**
				0.6667	0.6306	0.8333	0.4167	0.5556	
				0.7500	0.6294	0.7500	0.7500	0.7500	
yes	0.000020	0.015	0.25	0.5833	0.6833	0.6667	0.3333	0.4444	
yes	0.000015	0.015	0.25	0.5833	0.6941	0.5714	0.6667	0.6154	
yes	0.000025	0.015	0.25	0.6667	0.6712	0.6667	0.6667	0.6667	0.6013
				0.6250	0.6531	0.8000	0.3333	0.4706	
				0.6250	0.7033	0.6000	0.7500	0.6667	
yes	0.000025	0.006	0.25	0.5000	0.7063	0.5000	0.1667	0.2500	
yes	0.000025	0.01	0.25	0.6667	0.6607	0.6667	0.6667	0.6667	0.5295
				0.5417	0.7088	0.5455	0.5000	0.5217	
				0.6250	0.6492	1.0000	0.2500	0.4000	
yes	0.000025	0.01	0.30	0.5417	0.6985	0.5455	0.5000	0.5217	
yes	0.000010	0.015	0.20	0.4583	0.6992	0.3333	0.0833	0.1333	0.4059
				0.5833	0.6685	0.6667	0.3333	0.4444	
				0.6250	0.7068	0.6154	0.6667	0.6400	
yes	0.000015	0.02	0.20	0.5833	0.6568	0.7500	0.2500	0.3750	
yes	0.000010	0.015	0.15	0.7883	0.6589	0.8571	0.5000	0.6316	0.4234
				0.5417	0.6860	0.6000	0.2500	0.3529	
				0.5833	0.6681	1.0000	0.1667	0.2857	

Table 12: Intermediary test set results of *ad hoc* hyperparameter tuning process using the lemmatized dataset. All ran using 8 epochs and a batch size of 4. *average over all batches of the test set.

[15] Unrecorded due to manual error. This caused no major issue due to the presence of the other metrics.

D Results of second stage of hyperparameter tuning

Lem.	Nr. epochs	Batch size	Accuracy*	Loss*	Precision*	Recall*	F1-score*	Avg. f1-score
no	30	4	0.5000	0.6940	0.0000	0.0000	0.0000	0.4058
			0.5417	0.7207	0.5455	0.5000	0.5217	
			0.7083	0.6687	0.7273	0.6667	0.6957	
no	30	2	0.6250	0.7146	0.5882	0.8333	0.6897	0.5818
			0.6667	0.6396	0.8333	0.4167	0.5556	
			0.5000	0.7550	0.5000	0.5000	0.5000	
no	30	1	0.5000	0.6916	0.0000	0.0000	0.0000	0.0000
			0.5000	0.6911	0.0000	0.0000	0.0000	
			0.5000	0.6926	0.0000	0.0000	0.0000	
no	15	4	0.5833	0.6922	0.5714	0.6667	0.6154	0.5184
			0.6250	0.6774	0.5882	0.8333	0.6897	
			0.5000	0.7116	0.5000	0.1667	0.2500	
no	15	2	0.5833	0.6864	0.6250	0.4167	0.5000	0.3810
			0.5833	0.7317	0.5625	0.7500	0.6429	
			0.5000	0.6865	0.0000	0.0000	0.0000	
no	15	1	0.7083	0.6349	0.7778	0.5000	0.6667	0.4183
			0.7083	0.6088	1.0000	0.4167	0.5882	
			0.5000	0.6922	0.0000	0.0000	0.0000	
yes	30	4	0.5417	0.6976	0.6000	0.2500	0.3529	0.4960
			0.6250	0.6648	0.7143	0.4167	0.5263	
			0.6250	0.6837	0.6364	0.5833	0.6087	
yes	30	2	0.7500	0.6074	0.8750	0.5833	0.7000	**0.6854**
			0.8750	0.5794	0.8462	0.9167	0.8800	
			0.5417	0.7026	0.5556	0.4167	0.4762	
yes	30	1	0.5417	0.7211	0.5556	0.4167	0.4762	0.4762
			0.5833	0.6630	1.0000	0.1667	0.2857	
			0.7083	0.6348	0.7778	0.5833	0.6667	
yes	15	4	0.5417	0.6995	0.5714	0.3333	0.4211	0.3626
			0.5000	0.6962	0.5000	0.2500	0.3333	
			0.5000	0.7245	0.5000	0.2500	0.3333	
yes	15	2	0.5833	0.6749	0.7500	0.2500	0.3750	0.4300
			0.6250	0.6564	0.8000	0.3333	0.4706	
			0.5833	0.6869	0.6667	0.3333	0.4444	
yes	15	1	0.5417	0.7447	0.5333	0.6667	0.5926	0.6227
			0.6667	0.6640	0.5429	0.7500	0.6923	
			0.5833	0.6864	0.5833	0.5833	0.5833	

Table 13: Intermediary test set results of grid search hyperparameter tuning process using the two best models found in Appendix C. *average over all batches of the test set.

E Results of k-fold cross validation

Seed	Test fold	Accuracy*	Loss*	Precision*	Recall*	F1-score*	Avg. f1-score
42	0	0.5000	0.7349	0.5294	0.6923	0.6000	
	1	0.5833	0.6425	0.7143	0.3846	0.5000	
	2	0.5417	0.6873	0.6000	0.4615	0.5217	
9	0	0.5833	0.6483	1.000	0.2308	0.3750	0.5551
	1	0.7083	0.6453	0.6875	0.8462	0.7586	
	2	0.6250	0.6330	0.8333	0.3846	0.5263	
	3	0.7083	0.6479	0.7273	0.6667	0.6957	
	4	0.5417	0.6842	0.6000	0.2500	0.3529	
	5	0.5417	0.7274	0.5455	0.5000	0.5217	
	6	0.5833	0.6769	0.6667	0.3333	0.4444	
	7	0.7917	0.6120	0.8182	0.7500	0.7926	
	8	0.5833	0.6800	0.6250	0.4167	0.5000	
	9	0.5833	0.6831	0.5833	0.5833	0.5833	

Table 14: Intermediary test set results of k-fold cross validation, with $k = 10$. *average over all batches of the test set.