UNIVERSITÄT KOBLENZ · LANDAU
Faculty 4: Computer Science

WeST
People and Knowledge Networks
Institute for Web Science and Technologies

Generating a Non-Sexist Corpus Through Gamification for Automatic Sexism Detection

Master's Thesis

in partial fulfillment of the requirements for
the degree of Master of Science (M.Sc.)
in Web Science

submitted by

Ali Aghelmaleki

First Supervisor: JProf. Dr. Claudia Wagner
Second Supervisor: Anupama Aggarwal

Koblenz, February 2019

Statement

I hereby certify that this thesis has been composed by me and is based on my own work, that I did not use any further resources than specified – in particular no references unmentioned in the reference section – and that I did not submit this thesis to another examination before. The paper submission is identical to the submitted electronic version.

	Yes	No
I agree to have this thesis published in the library.	□	□
I agree to have this thesis published on the Web.	□	□
The thesis text is available under a Creative Commons License (CC BY-SA 4.0).	□	□
The source code is available under a GNU General Public License (GPLv3).	□	□
The collected data is available under a Creative Commons License (CC BY-SA 4.0).	□	□

. . . .

(Place, Date) (Signature)

Abstract

Most social media platforms allow users to freely express their opinions, feelings, and beliefs. However, in recent years the growing propagation of hate speech, offensive language, racism and sexism on the social media outlets have drawn attention from individuals, companies, and researchers. Today, sexism both online and offline with different forms, including blatant, covert, and subtle language, is a common phenomenon in society. A notable amount of work has been done over identifying sexist content and computationally detecting sexism which exists online. Although previous efforts have mostly used peoples' activities on social media platforms such as Twitter as a public and helpful source for collecting data, they neglect the fact that the method of gathering sexist tweets could be biased towards the initial search terms. Moreover, some forms of sexism could be missed since some tweets which contain offensive language could be misclassified as hate speech. Further, in existing hate speech corpora, sexist tweets mostly express hostile sexism, and to some degree, the other forms of sexism which also appear online was disregarded. Besides, the creation of labeled datasets with manual exertion, relying on users to report offensive comments with a tremendous effort by human annotators is not only a costly and time-consuming process, but it also raises the risk of involving discrimination under biased judgment.

This thesis generates a novel sexist and non-sexist dataset which is constructed via "*UnSexistifyIt*", an online web-based game that incentivizes the players to make minimal modifications to a sexist statement with the goal of turning it into a non-sexist statement and convincing other players that the modified statement is non-sexist. The game applies the methodology of "Game With A Purpose" to generate data as a side-effect of playing the game and also employs the gamification and crowdsourcing techniques to enhance non-game contexts. When voluntary participants play the game, they help to produce non-sexist statements which can reduce the cost of generating new corpus. This work explores how diverse individual beliefs concerning sexism are. Further, the result of this work highlights the impact of various linguistic features and content attributes regarding sexist language detection. Finally, this thesis could help to expand our understanding regarding the syntactic and semantic structure of sexist and non-sexist content and also provides insights to build a probabilistic classifier for single sentences into sexist or non-sexist classes and lastly find a potential ground truth for such a classifier.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This chapter presents the motivation of this work, then gives a brief overview of "Game With A Purpose" methodology, and lastly describes the thesis problems.

1.1 Motivation

According to the Oxford dictionary, sexism defines as “Prejudice, stereotyping, or discrimination, typically against women, on the basis of sex” [1]. The sexist language exists in everyday conversations between people. We, as human beings are conceived with no earlier information on how sexual orientation shapes our society. As we develop a more established vision, we recognize the presence of gender bias inclination in each part of our regular daily existences. Sex explicit titles and pronouns can impact our thoughts and assumptions regarding our occupations, objectives, and demands. Besides, we know that sexism exists in every part of society, from the workplace to the political ground. However, what is sexist language and how is it recognized? What makes an expression sexist? Would it be sexist if one says “Women are generally not as tall as men”?

An online social environment like Twitter [2] or Facebook [3] allows users to freely express their opinions, feelings, and beliefs. Also, social media platforms can provide a reflection of public sentiment on various events and problems. However, public networks are also an ideal venue for the generation of damaging information like offensive language and hate speech. Previous researches [1, 2, 3] suggest that social media platforms are helpful sources to gather information from the user’s activities and collect data. However, the collected data from social media has some limitations and is not always the best source for further analysis.

The primary motivation of this thesis is to introduce a new plaza rather than other existing means to collect data and consequently generate a novel sexist and non-sexist corpus. To this end, this work designs a "Game With A Purpose" [4] which is called "*UnSexistifyIt*", and employs a combination of gamification and crowdsourcing techniques. As a side-effect of playing the game, a unique sexist and non-sexist dataset would be generated. Further, the generated corpus of this game could be utilized for machine learning purposes.

[1] https://en.oxforddictionaries.com/definition/sexism

[2] http://twitter.com

[3] http://facebook.com

1.2 Game With A Purposes

The idea of a "Game With A Purpose"(GWAP) [4] comes from two fundamental observations. First, specific tasks are trivial for humans to do, but more challenging for computer programs. Second, people generally tend to play games. GWAP is an attempt to integrate computation and playing games. To this end, general design principles would be presented for the development and evaluation of games, in which users as a side-effect of playing, perform some tasks which computers are mostly not able to perform. As an upper-level synopsis, users playing these games perform basic tasks and consequently produce useful data. GWAP with exact mechanism configuration enables us to give the players an intention to take action to contribute to an overall desired outcome of our framework in a fun way. Moreover, players can address issues which are not clear enough for algorithms to spot [5].

However, from the game designer perspective, some questions will be raised such as: How the game should be designed to encourage people to play it? How to design a game to produce high-quality data for our specific purposes? To address these concerns, we need to motivate players to play the game and also we should encourage the players for more contribution. Particularly, we can introduce challenges, competitions, and leaderboards which lead players for more effort and better performance. Also, we can use different techniques such as imposing time limits, earning scores and unlocking new levels. Besides, the other question is how to test the accuracy of the generated data? We are not able to completely force players to stick to the game's rules, and we need to distinguish the irrelevant and bad results and discard those result from our corpus. The output from multiple independent players decreases the probability of having incorrect or meaningless results, and we can use truthful answers to test if the output is accurate. Over time, by playing such a game, we can generate genuine data which can improve the performance of our machine learning algorithms [5].

This work employs the GWAP method in order to collect crowdsourced data for the task of text annotation. The game architecture has been made with challenges and uses scoring, progression, competition and a variety of other game mechanisms to make the activity enjoyable.

1.3 Problem Statement

Social media outlets grow exponentially, and activities of all individual users have generated an enormous amount of data, but it also increasingly exploited the propagation of offensive language and hate-based activities. In fact, hate speech in the form of racist and sexist statements are a common phenomenon on social media. Recently, some efforts address the problem of identifying hate speech and more explicitly detecting sexist language. Recent work mostly utilized publicly available Twitter API in order to gather tweets which exhibit sexism and as a result create their corpus. This approach, however, has a couple of downsides which would be reviewed in this thesis.

The thesis is framed around the following general questions. In the following, these thesis questions would be elaborated to extend their objectives regarding this work.

Why a new sexist and non-sexist dataset is needed?

As mentioned earlier recent efforts used public Twitter API by performing an initial manual search of common sexual and gender terms which are generally used when exhibiting sexism in order to create a dataset of sexist tweets. However, most of these works are restricted to identifying the hostile form of sexism which is characterized by an explicitly negative attitude, while the benevolent sexism which is more subtle was neglected. Further, the method of gathering sexist tweets using Twitter API could be biased towards the initial search terms. Also, Some forms of sexism could be missed since some tweets which contain offensive language could be wrongly misclassified as hate speech and not as sexist language.

The objective of this first thesis question is to create a novel sexist and non-sexist dataset which addresses the issues as mentioned earlier with existing datasets and also introduces another method rather than using public Twitter API for data collection task.

To what extent the GWAP approach can be used to build a unique dataset?

The process of using game design methodologies can help to enhance the active involvement of individuals in the process of finding relevant material which satisfies an information need from within an extensive collection [6]. Nowadays, gamification implemented in various aspects of scientific fields. For instance, gamification technique applied in business [7, 8, 9], medicine [10, 11], and learning activities [12, 13]. Within an interesting game setting and using mechanics and dynamics of the game, we can increase users interaction to obtain positive results.

This work proposes an interactive game which employs the GWAP method to generate useful data as a by-product of play. Also, this work utilizes a combination of gamification and crowdsourcing techniques to improve the performance of users concerning given objectives. Participants of the game perform the task of text annotation by providing the judgments, as well as, checking and validating those judgments.

1.4 Thesis Contribution

The main contributions of this work are twofold.

- **Data collection through gamification**
  First, this work introduces a new approach for gathering data rather than existing methods such as using people's activities in social media platforms to collect data. To this end, this work presents *UnSexistifyIt*, an online game for collecting a corpus of sexist and non-sexist statements. [4]

- **Understanding the syntactic structure of sexist sentences**
  Second, this thesis analyzes the fundamental impact of context and content features of sexist and non-sexist statements mainly from a syntactic perspective.

[4] The code for this game available at: `https://github.com/gesiscss/UnSexistifyIt`

Chapter 2

Related Work

This chapter gives a high-level introductory review on the existing hate-related corpus. Moreover, other works which employ GWAP method and gamification techniques would be reviewed.

2.1 Existing Datasets

While hate speech is not a new problem, detection of hate speech, though, is a recent area. Hate speech detection that includes identification of sexist content has garnered much attention in recent times. A significant amount of effort has been done in social psychology science for identification of sexist language and its impact. However, Labeling a statement as sexist is not a simple task since there is no formal definition for it. Although efforts have been made to address this problem with manual exertion, there has been little progress since relying on users to report offensive comments requires an enormous effort by human annotators and it also increases the risk of applying discrimination under bias judgment.

A broadly used hate speech dataset has been made publicly available by Waseem and Hovy [1]. This dataset contains 16k collected and annotated tweets which are categorized into three classes: sexist, racist or neither. They build the corpus by performing an initial manual search of common terms used concerning religious, sexual, gender, and ethnic minorities. Then, they identified frequently occurring terms in tweets which contain hate speech. Based on samples, they used the public Twitter search API to collect the entire corpus. They proposed a list of criteria based on critical race theory and used Support Vector Machines (SVM) in order to classify tweets. However, one of the main downsides of their dataset and described approach is that collected sexist tweets mostly express hostile sexism. This approach for classification which combining hate speech with offensive language, making it hard to determine the extent to which they are truly identifying hate speech and sexist content.

The work by Jha and Mamidi [2] also collected data using the public Twitter Search API, and investigate the less noticeable form of sexism exhibited online. The authors addressed the tweets classification problem. They classified tweets into ‘Hostile’, ‘Benevolent’ or ‘Other’ while they used the dataset of annotated tweets by Waseem and Hovy [1]. However, they considered the existing ‘sexism’ tweets as being of class ‘Hostile. Further, they collected different forms of tweets separately and categorized tweets as ‘Benevolent’ class. Finally, they applied the FastText classifier by Joulin, et al. [14] as well as SVM classification to build a combined corpus of benevolent and hostile sexist tweets.

However, a limitation of this approach was that the method of gathering benevolently sexist tweets was biased towards the initial search terms and as a result, it is likely many forms of benevolent sexism to be missed.

2.2 Gamification

Gamification is a promising mechanism which is using game design methodologies and game mechanics to improve people's involvement in the process of information retrieval. The GWAP is an interactive and fun game that can be utilized in solving large-scale problems that are difficult for computers to solve. The ESP game which was developed by von Ahn and Dabbish [4] is the first and well-known example of GWAP, where users implicitly collaborate to label images as a by-product while playing the game. Labeling images is not an easy task for computer programs and algorithm, and also it can be tedious and time-consuming for humans. However, the ESP game introduces an interactive system to people with an incentive to perform beneficial work while they are enjoying to play the game. The dataset of the ESP is acquired through the game where two players will be randomly assigned as partners from among all the people playing the game. These two players independently guess what their partner is proposing label for each image. The goal of the game from the player's perspective is matching as many words as possible in a certain time limit. The partners will earn a certain number of points when they agreed on an image. Through playing this game, millions of images are labeled. The ESP game has shown that a large-scale problem can be solved with a GWAP method which uses individuals playing an online web-based game.

The Unfun.me [1] game by West and Horvitz [15] adopts the crowdsourcing approach by designing a game with a purpose. The Unfun.me is an online game that motivates players to make minimal changes to a satirical headline in a way that other players believe the modified version is a serious headline. As a result, a corpus of pairs of satirical headlines aligned with similar-but-serious-looking headlines would be collected. Comparing collected pairs through the game reveals that to exclude the humor from a satirical headline, players tend to replace one phrase with another and rarely do they remove phrases, and seldom propose new phrases. Finally, with initial results of the game and generated data, authors want to make further progress on understanding satire and the role of humor in intelligence, and also answering questions such as "Can computers learn to create jokes that would make us laugh?" [15].

[1] http://unfun.me

Chapter 3

Methodology

3.1 Data Collection Through Gamification

Gathering data in order to understand human behavior is a fundamental and limiting step in the interaction between human and computer [16]. Manual exertion not only requires an enormous effort by human annotators, but it also has a substantial effect on response times. One recent widely used method is crowdsourcing where we can obtain annotations from unknown group labors through an open call [17]. Online labor markets such as Amazon's Mechanical Turk (MTurk) [1] can provide an engaging platform for conducting human problems researches. Crowdsourcing platforms have been proved to successfully recreate experimental results [18, 19] and provide extra advantages such as constant access to a large and diverse participant pool [20].

However, concerns have been raised regarding the quality of the collected data, for instance, the existence of individuals who do not care or do not understand the underlying settings of the task, or potentially biased judgment by users, and also the presence of malicious users could influence the quality of collected data. Moreover, the principles of low pay could be led to a lack of the worker's motivations since they have been encouraged to perform their assignments through micropayments [21]. Further, although the cost of labor is low, it is not free of charge. The other concern arises when developing a crowdsourcing application; it is a fact that we rely on the determinations and judgments taken by the users. Possible biased discrimination and lack of motivation are essential issues with crowdsourcing. Nevertheless, the question is how to engage users using an application and granting observations?

In order to address the crowdsourcing obstacles, this work implemented a game to collect the data in an engaging and entertaining way. Here, gamification techniques can play a significant role since such a system provides a mechanism to motivate users to utilize the application and produce information while they are playing a gamified application [22]. Preliminary examinations confirmed employed gamification mechanism is successful in stimulating a large number of voluntary participants [23] and to increase task entertainment [24]. Gamification method seeks to boost users' experience, and to this end, game mechanics adapted in the application. While some elements of the game are implemented components in the software, others more address users' sentiments. We can say there are three main objectives the

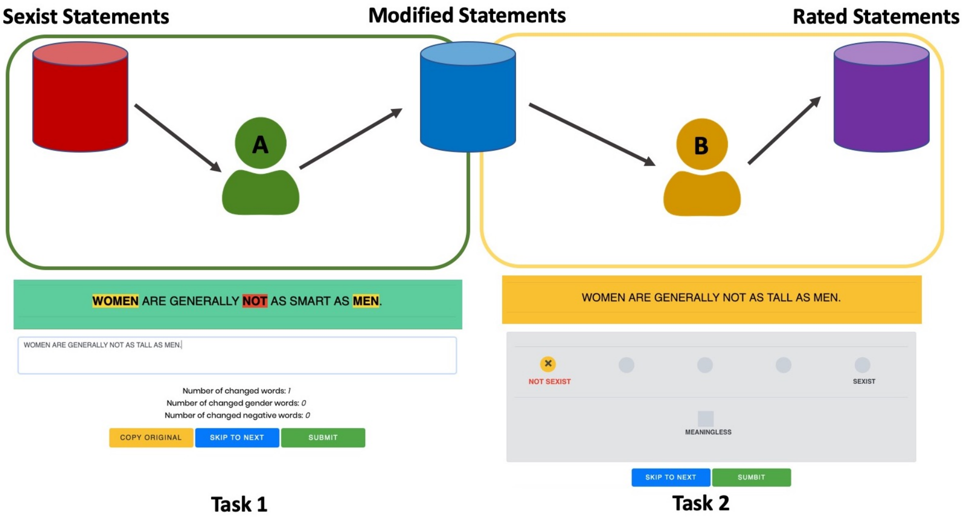
[1] https://www.mturk.com

mechanics of the game has pursued [25]. First, its displays progression second provides feedback, and third engages specific behavior. In order to facilitate the creation of the data, well-known game mechanics and dynamics could be utilized. The principle of employing such techniques is to encourage users to participate in the data collection process by playing a fun game. The result of this public participation is a large amount of the desired dataset that will be used for further analysis.

UnSexistifyIt as a serious GWAP encourages voluntary participants to generate a non-sexist or less sexist content out of the given (original) sexist sentences. The mechanisms of the gamification have been employed in this game. Mechanics of the game includes progression, investment, and cascading information theory [26]. Progression is achieved when users play the game and earn scores, Cascading Information theory is applied when requiring the player to follow the given rules and instructions to tackle the assigned task, and Investment obtained by displaying the leaderboard provides public notice for top players.

3.2 Elements of The Game

This section introduces the fundamental elements of the *UnSexistifyIt* game and the primary game mechanics which have been employed in this game. Figure 3.1 visually depicted two challenges of the game. In order to incentivize players to make high-quality participation, we reward the players. The following illustrates the game's elements in more details.



WOMEN ARE GENERALLY NOT AS SMART AS MEN.

WOMEN ARE GENERALLY NOT AS TALL AS MEN.

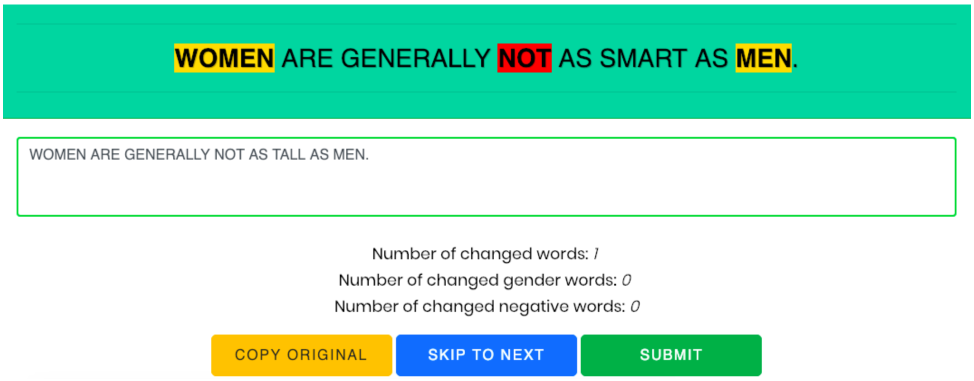
Number of changed words: 1
Number of changed gender words: 0
Number of changed negative words: 0

WOMEN ARE GENERALLY NOT AS TALL AS MEN.

Figure 3.1: The general perspective of *UnSexistifyIt* game

First Task; UnSexistifyIt is the core and primary task of the game where modification of sexist statement into non-sexist statements happens. A player(A) is given a sexist sentence and is asked to turn it into a non-sexist or as less sexist as possible which could conceivably have been published in a non-satirical article or serious news outlet, by changing as few words as possible. The sexist statements corpus collected sexist sentences from various sources [27, 28, 29, 30, 31, 32, 33]. While players were playing this challenge, a dataset of potentially non-sexist statements would be created. Players are given unique sentences for modification which means they do not see a sentence twice. Also, players can skip the current sentence if they do not understand it or find it difficult to modify. Figure 3.2 visualized the first task of the game.

Second Task; Rating is the second task of the game where a sentence which randomly comes from the first task is shown to a player(B). The given sentence was modified by another player(A) in the first task of the game. The player(B) is asked to indicate how sexist the given sentence is by choosing a rating from one to five. Where rating one means the given sentence is not sexist at all, and rating five means the given sentence is utterly sexist. Whether on purpose or not, the other player(A) may have done a lousy job in the first task, and the modified sentence may be awkward, grammatically incorrect, or meaningless. The active player(B) can select the "Meaningless" option if the given sentence makes no sense. The active player(B) will not see her/his own modified sentences from the first task, and also will not ask to rate a modified sentence twice. However, there could be some cases where different players made similar changes in the first challenge so in this scenario the active player could see alike or identical sentences. Figure 3.3 visualized the second task of the game.

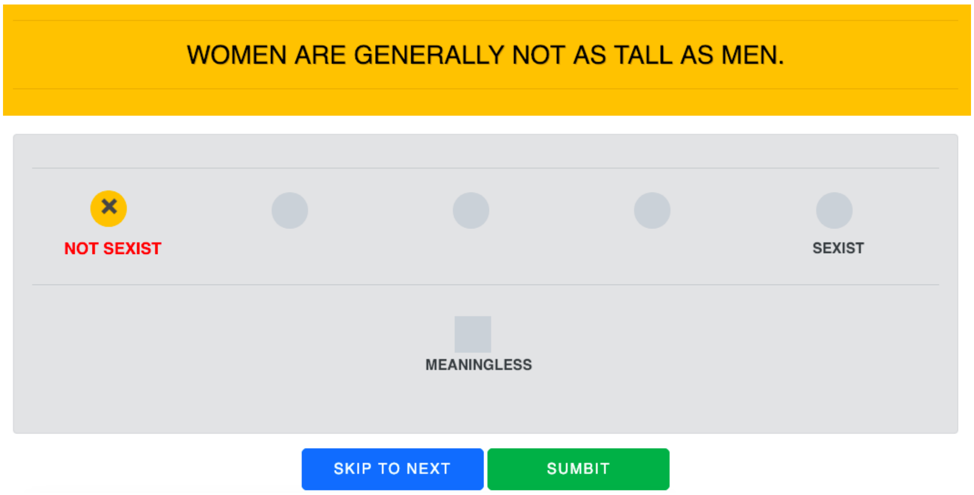


WOMEN ARE GENERALLY NOT AS SMART AS MEN.

WOMEN ARE GENERALLY NOT AS TALL AS MEN.

Number of changed words: *1*
Number of changed gender words: *0*
Number of changed negative words: *0*

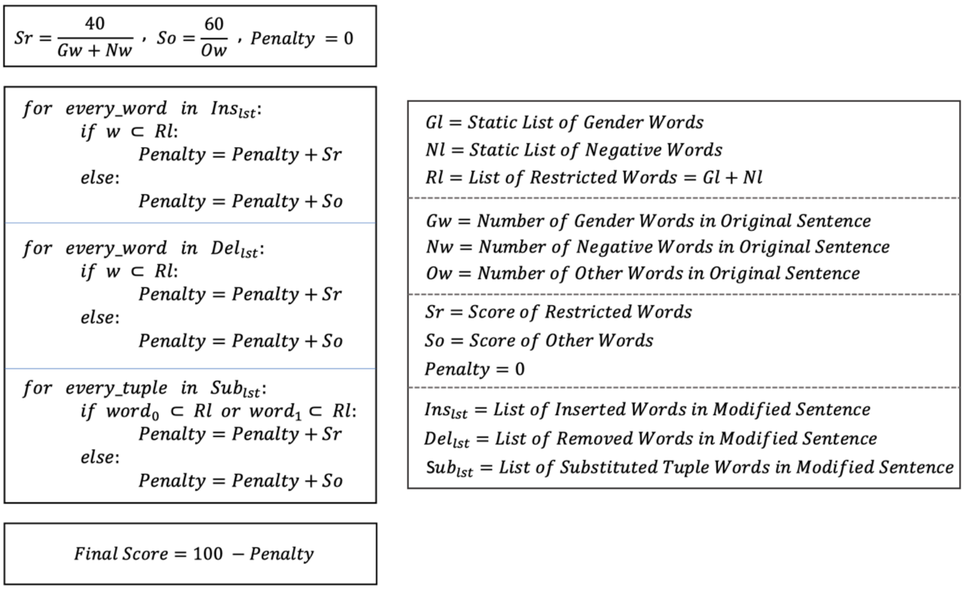
Figure 3.2: The first task of the game; UnSexistifyIt



WOMEN ARE GENERALLY NOT AS TALL AS MEN.

Figure 3.3: The second task of the game; Rating

Reward for Task 1. While players are supposed to turn the sexist sentence into a non-sexist sentence with minimal modification, their reward would be calculated based on the similarity between the original and modified sentences. In order to calculate the difference between the original and modified sentences, the token-based edit distance [34] is used where words are known as units of a string, rather than characters, examined to compute the minimum number of insertions, deletions, and substitutions of the original sentences which can be transformed into the modified one. In other words, the players are penalized depending on the number of changes they made to turn an original sexist sentence into a non-sexist sentence. The idea here is encouraging players to make as few changes as possible to lose fewer points. Besides, making changes to certain kind of words which we call them "Restricted Words" incur more substantial penalties. Restricted words include "Gender Words" such as 'Men', 'Female', 'Mother', and so forth and "Negative Words" such as 'Not', 'Don't' and so forth. In total 314 of both singular and plural gender words, and 15 negative words formed our two static lists of restricted words. Restricted words are highlighted in the original sentence to remind players that changing them would result in a higher penalty. Figure 3.4 illustrates the reward calculation formula for the first task of the game.



$$Sr = \frac{40}{Gw + Nw} \;,\; So = \frac{60}{Ow} \;,\; Penalty\ = 0$$

```
for every_word in $Ins_{lst}$:
    if $w \subset Rl$:
        $Penalty = Penalty + Sr$
    else:
        $Penalty = Penalty + So$

for every_word in $Del_{lst}$:
    if $w \subset Rl$:
        $Penalty = Penalty + Sr$
    else:
        $Penalty = Penalty + So$

for every_tuple in $Sub_{lst}$:
    if $word_0 \subset Rl$ or $word_1 \subset Rl$:
        $Penalty = Penalty + Sr$
    else:
        $Penalty = Penalty + So$
```

$$Final\ Score = 100\ - Penalty$$

Gl = Static List of Gender Words
Nl = Static List of Negative Words
Rl = List of Restricted Words = $Gl + Nl$

Gw = Number of Gender Words in Original Sentence
Nw = Number of Negative Words in Original Sentence
Ow = Number of Other Words in Original Sentence

Sr = Score of Restricted Words
So = Score of Other Words
$Penalty = 0$

Ins_{lst} = List of Inserted Words in Modified Sentence
Del_{lst} = List of Removed Words in Modified Sentence
Sub_{lst} = List of Substituted Tuple Words in Modified Sentence

Figure 3.4: Reward calculation formula for UnSexistifyIt task

Based on the presented formula, first, the score of restricted words and the score of the other words in the original sexist sentence would be calculated separately. Restricted words weighted 60%, and other words weighted 40% of the total score, since as mentioned above modifying restricted words incur the more substantial penalty. Then, for every word in the list of inserted and removed words, and for every pair of words in the list of substituted words, the penalty would be computed accordingly depending on whether the changed word belongs to the restricted words or the other words categories. The total penalty value subtracts from the predefined base score which equals to 100, and the result would be the final score. However, this is an empirical formula in order to compute the reward for playing the first task of the game, and it can be tuned. The following presents an example of a rewarding system for the first task of the game.

- Original Sentence: *'Some jobs are not appropriate for women'*
- Modified Sentence: *'Some jobs are not appropriate for underage.'*
  - Number of Gender words in the original sentence = 1 ("women")
  - Number of Negative words in the original sentence = 1 ("not")
  - Total number of Restricted words in the original sentence = 2
  - Number of Other words in the original sentence = 5 ("some", "jobs", "are", "appropriate", "for")

  - Score of Restricted words = 40 / 2 = 20
  - Score of Other words = 60 / 5 = 12

  - List of Inserted words in the modified sentence: []
  - List of Deleted words in the modified sentence: []
  - List of Substituted pair words in the modified sentence: [("women", "underage")]

  - Penalty = 20
  - Score = 100 - 20 = 80

Reward for Task 2. Since the purpose of the active player in the second task of the game is to determine whether the modified sentence is non-sexist, we do not have a ground truth rating for the modified sentence. The active player can not earn rewards for participating in task 2. Instead, the active player's rating answer uses to calculate how much reward the player gets who created the modified sentence in the first task. The more the active player thinks the given modified sentence is not sexist, the more the author of the modified sentence will earn rewards. Also, the author will lose more points if the active player believes the modified sentence is still sexist or meaningless. As a result, in this task players can either earn or lose scores passively based on their performance in the first task and other players judgment.

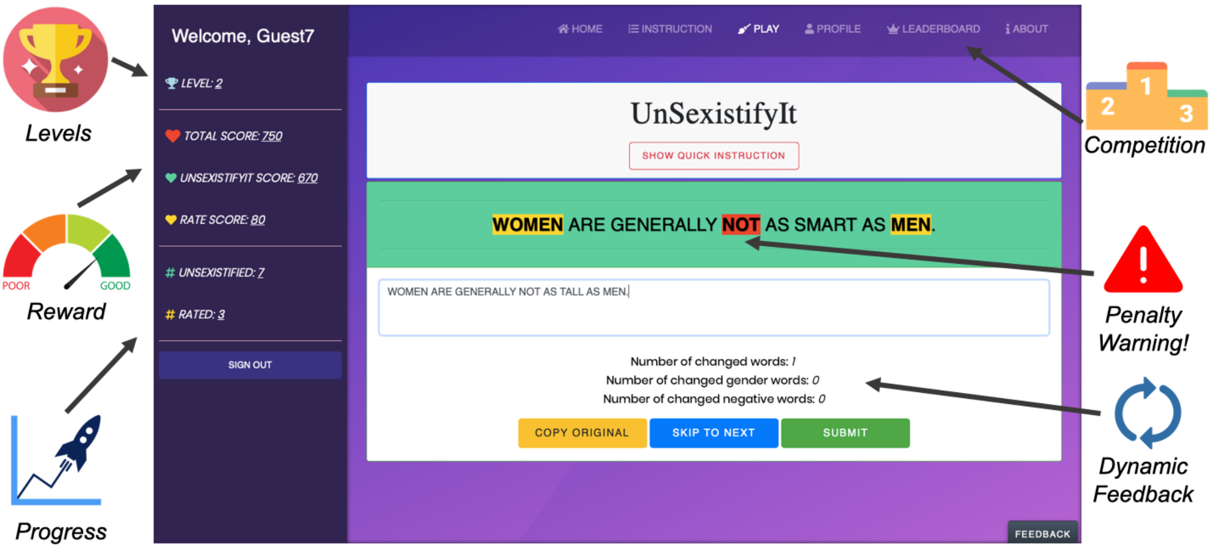
Levels are an indication that players have reached a milestone and defined as point thresholds. Players can automatically progress based on their participation. The game contains two levels, the first level includes only UnSexistifyIt as the primary task, and it is already unlocked for the players when they enter the game either as registered players or as a guest player. The second level is Rating task which can be unlocked as players achieve a certain amount of points.

Status; Players would be informed about their progress in the game with a personalized and informing status panel which contains current level, earned scores in each task separately, and the number of times played for each task. This informative status panel would be updated based on players performance.

Dynamic Feedback is employed in the game to inflate the engagement of players, and also to make players feel more possession and intent when engaging with tasks. Feedback in the game has a direct link to formative assessment. The first task, *UnSexistifyIt*, provides real-time and dynamic feedback which shows the number of changed words, the number of changed gender words and the number of changed negative words based on the modification to the given original statement by players to alert them to perform well.

Leaderboard is implemented in the game as an ordered list of players based on the scores they have obtained within both tasks. The leaderboard uses competition sense to drive valuable behavior of the players, and it is a beneficial tool to track and display desired actions.

Figure 3.5 displays an overview of the game elements.



Welcome, Guest7

LEVEL: 2

TOTAL SCORE: 750

UNSEXISTIFYIT SCORE: 670

RATE SCORE: 80

UNSEXISTIFIED: 7

RATED: 3

UnSexistifyIt

WOMEN ARE GENERALLY NOT AS SMART AS MEN.

WOMEN ARE GENERALLY NOT AS TALL AS MEN.

Number of changed words: 1
Number of changed gender words: 0
Number of changed negative words: 0

Figure 3.5: Overview of the game elements

3.3 Development

UnSexistifyIt is developed using MEAN stack. MEAN is an abbreviation of JavaScript-based technologies and stands for “MongoDB,” “Express.js,” “AngularJS” and “Node.js” which are known to synergize well together, to create websites and mobile applications. The back-end is using MongoDB, Express.js, and Node.js while AngularJS is handling the front-end framework. Since all part of MEAN Stack utilize JavaScript, both server-side and client-side execution environments written in one language, and it empowers us to build a fast, robust and maintainable web application. *UnSexistifyIt* designed as RESTful API to be able to support a variety of end-user devices such as mobile phones and tablets. Therefore, it leverages less bandwidth and would be more suitable for internet usage. Figure 3.6 illustrates a high-level overview of the application. Following are a brief description of each component.

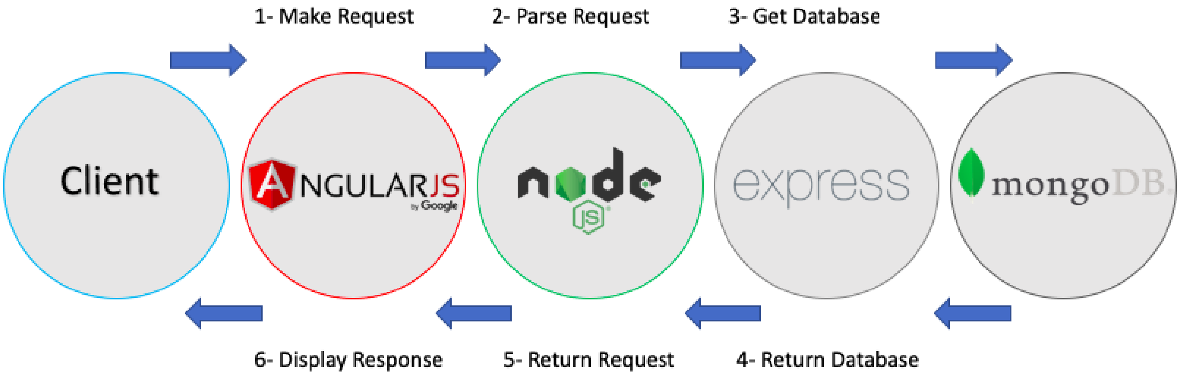


Figure 3.6: MEAN Stack Application overview

MongoDB [2] is the first piece of the frame which is a NoSQL Database program which stores in a JSON-like document, meaning "fields can vary from document to document and data structure can be changed over time. The document model maps to the objects in the application code, making data easy to work with" [35]. Mongoose [3], a "MongoDB object modeling tool designed to work in an asynchronous environment" [36], provides a straight-forward, schema-based solution to model application data. Schemas are used to map to a MongoDB collection and define the shape of the document within that collection.

Express.js [4] "is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications. With a myriad of HTTP utility methods and middleware at your disposal, creating a robust API is quick and easy. Express provides a thin layer of fundamental web application features, without obscuring Node.js features" [37].

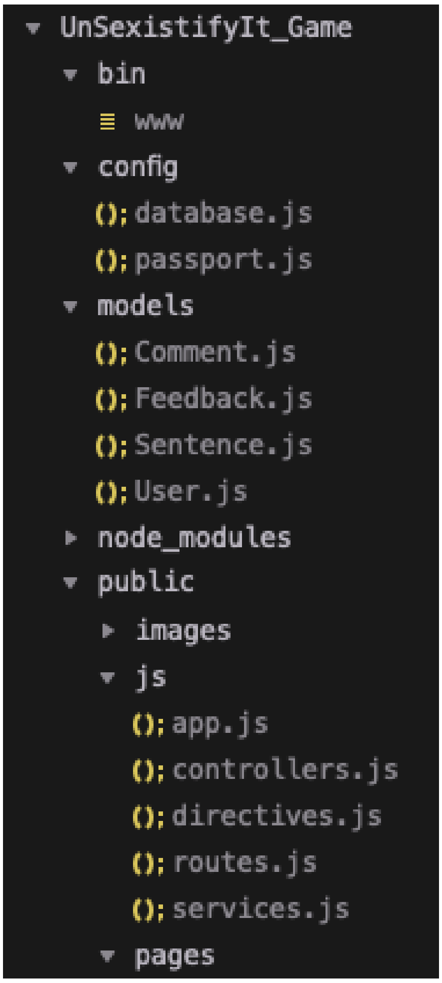
Node.js [5] is a runtime environment to design server-side applications in JavaScript. Node.js uses an event-driven, non-blocking I/O model which makes it lightweight and efficient. It grants a significant boost that comes from using the same language on both the front-end and the back-end. "As an asynchronous event-driven JavaScript runtime, Node is designed to build scalable network applications" [38].

AngularJS [6] is a lightweight MVW(Model-View-Whatever) framework where Whatever stands for "whatever works for you," extends HTML vocabulary for the application [39]. It enables the application to use data-binding which is an automatic way of updating the view when the model changed and the other way around. The data-binding feature and dependency injection eliminate much of the code, it all happens within the browser.

- **Implementation**

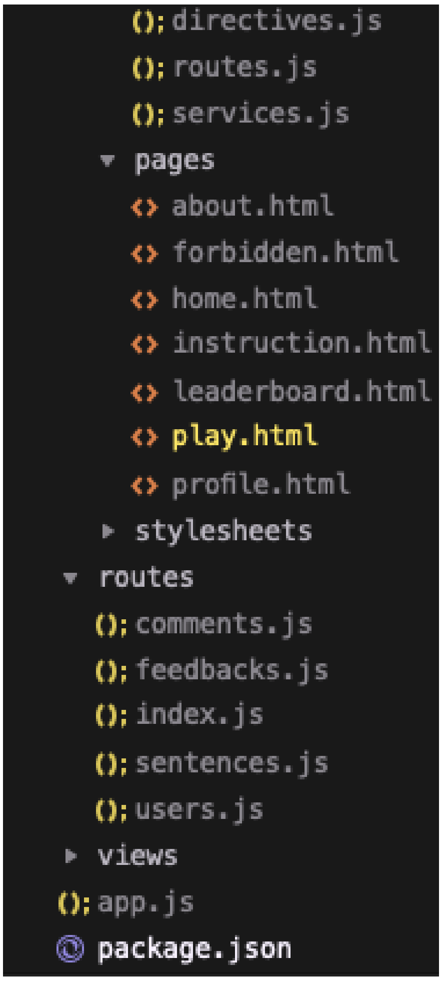
[2] https://www.mongodb.com
[3] https://mongoosejs.com
[4] https://expressjs.com/
[5] https://nodejs.org
[6] https://angularjs.org/

The following shows a general overview of the implemented RESTful API which was developed with MEAN Stack technologies. Figure 3.7 shows the structure of the application, and it follows by concise explanations about essential parts of this application.



```
▾ UnSexistifyIt_Game
    ▾ bin
        www
    ▾ config
        database.js
        passport.js
    ▾ models
        Comment.js
        Feedback.js
        Sentence.js
        User.js
    ▸ node_modules
    ▾ public
        ▸ images
        ▾ js
            app.js
            controllers.js
            directives.js
            routes.js
            services.js
        ▾ pages
```

Figure 3.7: Application Folder Structure(1)



```
            directives.js
            routes.js
            services.js
        ▾ pages
            about.html
            forbidden.html
            home.html
            instruction.html
            leaderboard.html
            play.html
            profile.html
        ▸ stylesheets
    ▾ routes
        comments.js
        feedbacks.js
        index.js
        sentences.js
        users.js
    ▸ views
    app.js
    package.json
```

Figure 3.8: Application Folder Structure(2)

- **bin** includes www file to create the HTTP server.
- **config** directory holds database.js to connect to the database
- **models** folder includes Schemas which mapping to MongoDB collections to hold Mongoose.js model files. The database has three main models including the Sentence model is a collection of original sexist sentences, the User model store all the user information and Comment model is the collection of modified sentences by the users. Handling the subdocuments as documents embedded in other documents was the challenging part.
- **node_modules** created by npm install to bring in required modules.
- **public** directory contains all the front-end including the Angular code for the project.
    - **js** includes AngularJS controllers which control the data of the application AngularJs services as functions or objects for the application.

    - **pages** includes partial HTML pages which using AngularJS to extend its attributes with Directives and binds data to it with Expressions.

- **routes** directory holds router files to handle application routing and to separate the services of the separate parts of the application. Since we created an earlier model, we can generate our Express routes to handle our API calls.
- **views** contains public index view
- **app.js** contains all the server-side code used to implement the REST API which is written in Node.js, using the Express framework and the MongoDB Node.js driver.
- **package.json** is a configuration file that contains the metadata for the application.

Chapter 4

Results and Analysis

Via *UnSexistifyIt*, we have collected 598 modified non-sexist sentences for 92 distinct sexist sentences. The modified sentences came from 122 unique user ids, which means approximately five modified sentences and rating per user. Table 4.1 demonstrates the size of the collected dataset. In order to show the consistency among the collected results provided by MTurk users, we need to evaluate inter-rater reliability(IRR). The assessment of IRR gives a way to measure the degree of agreement between multiple annotators (MTurk workers) who perform the task independently. In our game, we asked three annotators to rate each sentence, and we added only those sentences to our dataset which had an IRR score of two or above. Therefore, after filtering the results, 263 of modified sentences which made semantic sense but had zero inter-annotator agreement for final sexism score were discarded from all modified sentences. In total, 335 modified sentences have been taken into account. In the second task of the game, players attribute sexist ratings to the modified sentences. All the modified sentences have received a rating. Recall that, ratings vary from one to five, where one indicates that the modified sentence is not sexist at all, and rating five means the modified sentence is 100% sexist.

Sentences	**Number**	**Rating**
Original sexist	105	5
All modified	598	1 to 5
Collected modified	335	1 to 5

Table 4.1: Size of collected corpus

4.1 Analysis of Game Dynamics

In this section, we start by analyzing the edit operations players perform in the first task of the game. Then, the sexists rating which players provide in the second task would be investigated. Next, the balance between edit distance and rating would be examined and finally we will discuss the distribution of the edit operations.

4.1.1 Edit Distance

The first question is how much players tend to modify the original sexist sentence in order to turn it into a non-sexist version. We quantify this notion with the token-based edit distance between the sexist sentence and the modified non-sexist version. Figure 4.1, which plots the distribution of edit distance, shows that minimal edits are most common, as incentivized through the reward system of the game for the first task. The smallest possible edit distance equals to one since players had to make at least one change in the original sexist sentence. Particularly, 35% of all modified sentences made the minimum change, and 67% have distance up to two. Dynamic programming is used for computing the edit distance and for measuring the difference between the two sentences. The Levenshtein distance [40] measures the difference between two words and computes the minimum number of single-character edits including insertions, deletions or substitutions in order to turn one word into the other word. Here, the Levenshtein distance is employed in word-level to measure the edit distance between the original and modified sentences.

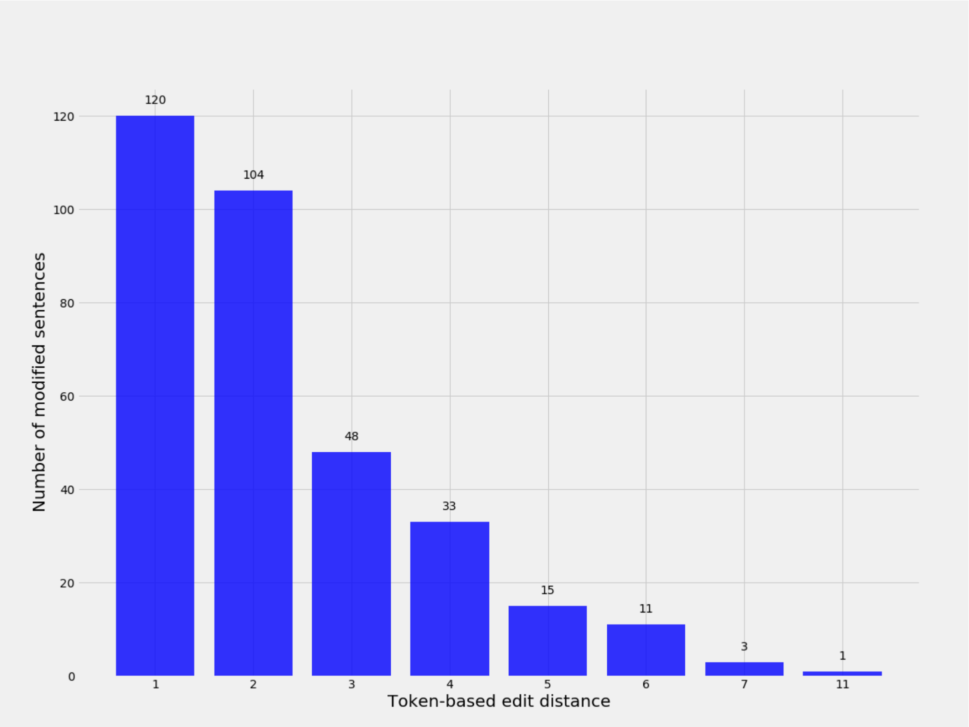


Figure 4.1: Distribution of token-based edit distance between the original and modified sentences

In the following example, the edit distance between the original sexist sentence and the modified version equals five.

- Original Sentence: *'Sons in a family should be given more encouragement to go to college than daughters.'*

- Modified Sentence: *'Sons and daughters in a family should be given equal encouragement to go to college.'*

4.1.2 Rating Distribution

In the second task of the game, players were asked to rate how sexist the given modified sentence is, where the sentence which is rated one is considered as non-sexist, and the sentence which is rated five is considered as sexist, and other rating values between one and five show the sexist degree. Figure 4.2 displays the distribution of rating so that we can understand how players rate the modified sentences. Although most of the players successfully modified the original sexist sentences which were rated ones and twos by 67%, some other players were not able to accomplish the desired result, and about 25% of modified sentences were rated fours and fives.

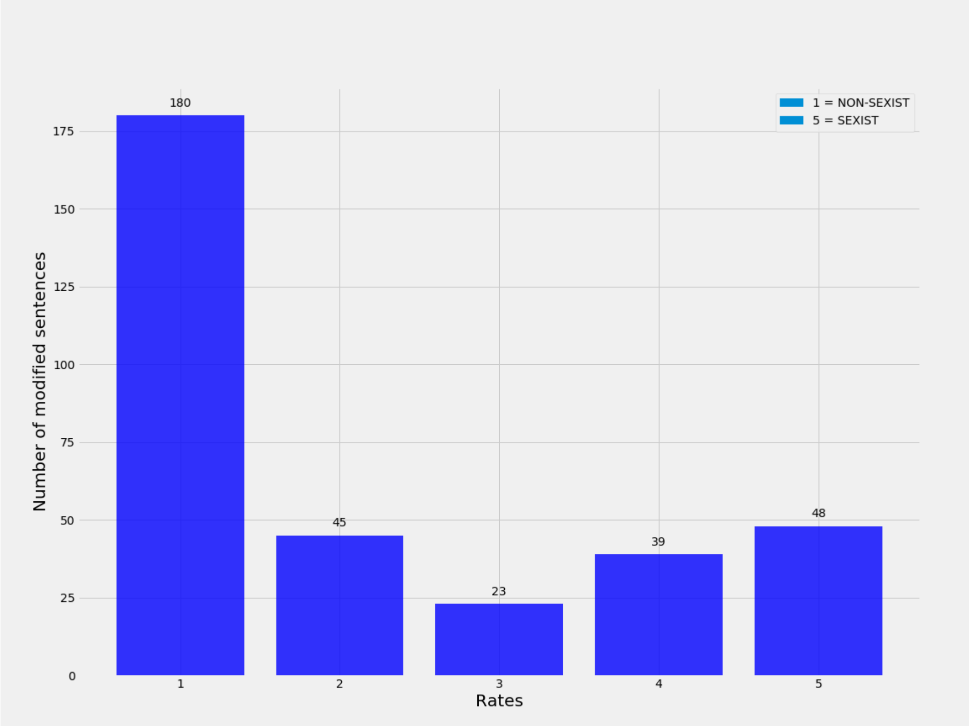


Figure 4.2: Rating distributions for modified sentences

Particularly, those modified sentences which rated five were analyzed. Hence, these failed cases can be categorized into two main groups. The first group is about those sentences which players only swap the gender words in the original sentence in order to remove the sexist content and turn it into the non-sexist version. However, the modified sentences still remained sexist since the sexism against one gender had been changed against the other gender. The following are the example of these cases.

In the first example, the original sentence indicates sexism against females, and it turns to sexism against men in the modified version.

- Original Sentence: *'It is difficult to work for a female boss.'*
- Modified Sentence: *'It is difficult to work for a male boss.'*

This next example is the opposite of the first one where the original sentence is sexism against men/fathers, and the modified version is sexism against women/mothers.

- Original Sentence: *'Mothers should make most of the decisions on how to bring up their children.'*
- Modified Sentence: *'Fathers should make most of the decisions on how to bring up their children.'*

The second group of modified sentences which failed to remove sexist content is about those sentences which players cannot identify the sexism in the original sentence and therefore can not adjust the sexist sentence to eliminate the sexist content. More in-depth studying of these failed cases as future work can yield interesting insights into how people identify sexism. The following shows examples of this group.

In this example, the original sexist sentence shows sexism against women, and although the player did not swap the word 'women' as gender word, she/he was not able to turn the original sexist sentence into a non-sexist one after modification.

- Original Sentence: *'Women exaggerate problems they have at work.'*
- Modified Sentence: *'Women exaggerate problems they have at home.'*

The next example also shows how the player failed to identify sexist content and as a result, the modified sentence rated as five which means it is still a completely sexist sentence.

- Original Sentence: *'Important career-related decisions should be left to the husband.'*
- Modified Sentence: *'Important household-related decisions should be left to the husband.'*

4.1.3 Effect of Edit Distance on Sexist Rating

More substantial edits make it easier to turn a sexist sentence into a non-sexist sentence, while smaller edits increase the risk of not completely removing sexist content. Table 4.2 shows the mean average sexist rating of modified sentences against the edit distance, and it reveals how the tradeoff works practically. It is more difficult to turn the sexist sentence into a non-sexist version by only changing one or two words than by changing three words, while the margin result is insignificant for extensive edits. For edit distance between one and four which is 91% of all modified sentences, the rating has a positive correlation with edit distance.

Token-based Edit Distance	Avg Rating After Modification
1	2.54
2	2.34
3	1.89
4	1.69
5	1.46
6	1.09
7	1.33
11	1

Table 4.2: Tradeoff of edit distance vs. sexist rating

4.1.4 Edit Operations

We can keep track of an optimal sequence of insertions, deletions, and substitutions for transforming the sexist sentence into a non-sexist sentence with using dynamic programming. Figure 4.3 plots the distribution of edit operations, over modified sentences. We can observe that substitutions clearly dominate by 68%, followed by insertions by 17% and deletions by 14%.

In addition to analyzing all the modified sentences, it is interesting to analyze those modified sentences with edit distance one. Because modified sentences with edit distance one not only are the most frequent, but these sentences are the most similar to the original sentences. Figure 4.3 also plots the distribution of edit operations for modified sentences with edit distance one. We can see that substitutions dominate even more by 80%, where insertions by 17% did not change, but deletions are even fewer by just 4%.

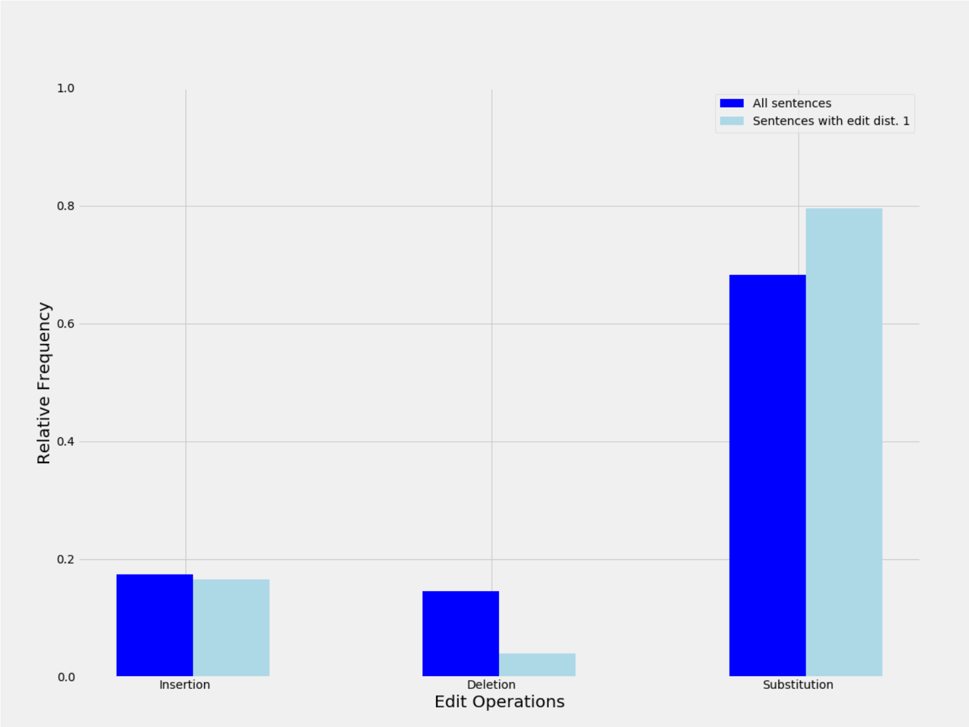


Figure 4.3: Distribution of token-based edit operations

4.2 Syntactic Analysis of Aligned Corpus

In this section, we ask what parts of an original sexist sentence should be modified in order to remove the sexist part from it and turn it into a non-sexist sentence. We tackle this question from a syntactic perspective. Then we go one level deeper and investigate how the sexist rating of the modified sentences would be affected by the insertion and deletion of restricted words such as gender words and negative words.

4.2.1 How Similar Are Two Sentences

In order to analyze the syntactic similarity between an original sentence and the modified sentence, first, we need to use Sequence Alignment method to identify regions of similarity. To this end, we used the Needleman–Wunsch algorithm [41] which uses dynamic programming to achieve global alignment. Now, after both strings get aligned, we want to realize which syntactic Part-Of-Speech (POS) tags are modified in the original sentence. In order to get the POS tag of each sentence, we used tokenizer and POS tagger of Python's NLTK library [1] to output specific tags for words. After all the words are classified into their POS(classes) and labeled accordingly, we can use this collection of tags in the aligned sequence of edit operations including insertions, deletions, and substitutions. As a result, we can see how POS tags of the original sentences have been altered in the modified versions. Moreover, in order to measure the syntactic effects, it is helpful to analyze the rating number for edits operations.

In figure 4.4 we can observe that the words that classified as Adverb have been inserted more than other word classes in the modified sentences by 42%, and it is followed by about 10% for Coordinating Conjunction classes. The Plural and Singular form of nouns are in the next place of being inserted into modified sentences both by about 8%. However, we knew that the word "Not" which classified as the Adverb class had been used more than other words to turn the sexist sentence into non-sexist one. Therefore, the word "Not" excluded from the inserted dataset and considered separately and plotted as a stacked bar over other Adverb words. Even after excluding the word "Not", Adverb class still dominate the inserted words by 21% and the other next two tags increased slightly by about 12%.

[1] https://www.nltk.org/book/ch05.html

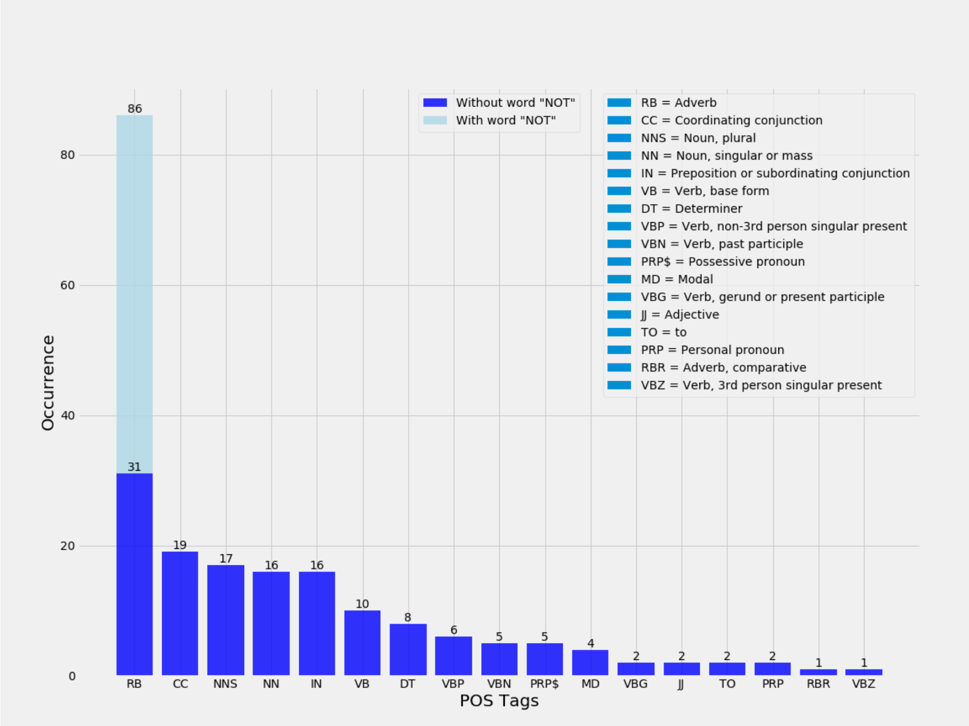


Figure 4.4: POS tags of inserted words in the modified sentences

The following is an example of using sequence alignment mechanism and then getting POS tags of inserted words in the modified sentence:

- Original Sentence: *'The intellectual leadership of a community should be largely in the hands of - men - -.'*

- Modified Sentence: *'The intellectual leadership of a community should be largely in the hands of deserving men and women.'*

After applying sequence alignment, we can see that the words 'deserving', 'and', and 'women' were inserted in the modified sentence. Then we can output their POS tags accordingly as follows: ('deserving', 'VBG'), ('and', 'CC'), ('women', 'NNS'). Table 4.3 shows the top five most frequent inserted words with their POS tags and the mean average rating of the modified sentences.

Rank	**Word**	**POS Tag**	**Number**	**Avg Rating**
1	Not	Adverb	55	1.78
2	And	Coordinating conjunction	19	1.36
3	As	Preposition or subordinating conjunction	11	1.55
4	Women	Noun, plural	8	1.62
5	Only	Adverb	6	1.50

Table 4.3: Top five most frequent inserted words

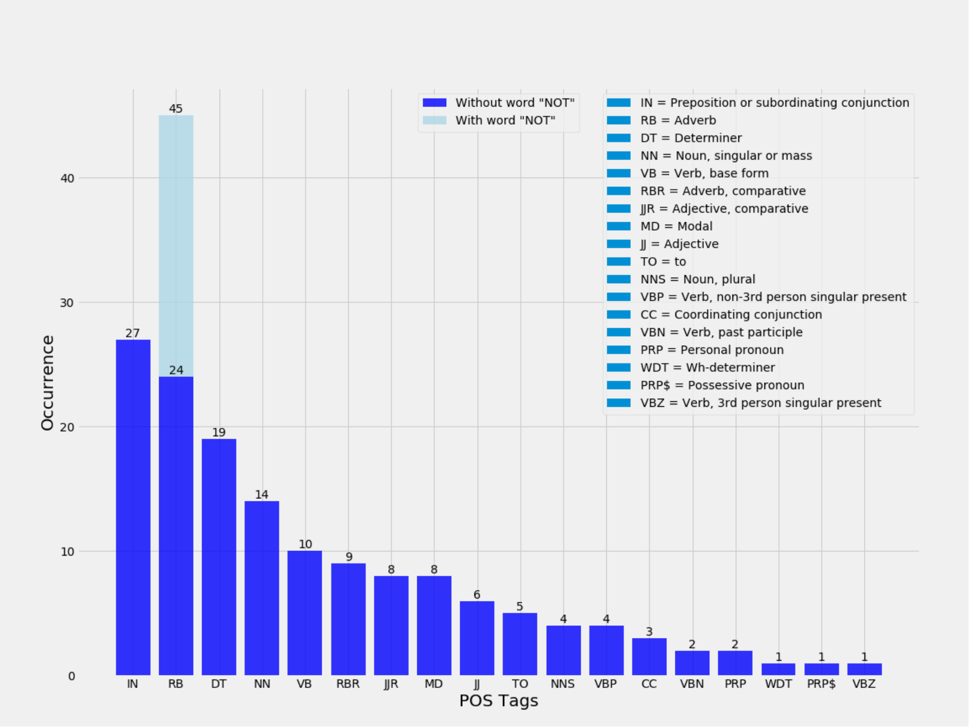


Figure 4.5: POS tags of removed words in the modified sentences

Figure 4.5 plots the distribution of POS tags of removed words. Similar to the inserted words the words with Adverb class have been removed more than other classes by 27%, followed by Preposition or Subordinating Conjunction by 16%, and Determiner by 11%. Here also the word "Not" has been removed to a very great degree in order to modify sexist sentences into non-sexist one. As a result, we excluded the word "Not" and rebuilt the removed words pos tags dataset. However, in contrast to the inserted words, after excluding the word "Not", we can see that Preposition or Subordinating Conjunction class becomes the largest removed words class by 18%, Adverb class has been decreased by 16%, and Determiner by 13%.

The following is an example of POS tags of removed words:

- Original Sentence: *’Women should worry less about their rights and more about becoming good wives and mothers.’*

- Modified Sentence: *’Women should worry - about their rights - - before becoming good wives and mothers.’*

Here also sequence alignment applied for both sentences, and we can see that the words ’less’, ’and’, and ’more’ were removed from the original sentence in the modified version so that we can output the POS tags of these words as follows: (’less’, ’RBR’), (’and’, ’CC’), (’more’, ’JJR’). Table 4.4 displays the top five most frequent removed words with their POS tags and the mean average rating of the modified sentences.

Rank	Word	POS Tag	Number	Avg Rating
1	Not	Adverb	21	1.40
2	Than	Preposition or subordinating conjunction	20	1.30
3	More	Adverb, comparative	14	1.78
4	The	Determiner	12	1.00
5	Rather	Adverb	11	1.72

Table 4.4: Top five most frequent removed words

The same process has been done for words which were substituted in the original and modified sentences. Figure 4.6 plots the distribution of the substituted words, and it shows the dominance of the Noun, in different forms such as Plural and Singular/Mass by 40%, and then Adjective and Preposition or Subordinating Conjunction by about 10% are the most frequent substituted words class. Studying the substituted words yields that the words "Women" and "Men" are the most frequent words, by 7% and 6% respectively, which players are most likely to substitute with each other. Those words are followed by words "than", "as", and "more" each by 4%, which indicates the player's tendency to replace Adjective and Prepositions for turning a sexist sentence into non-sexist one.

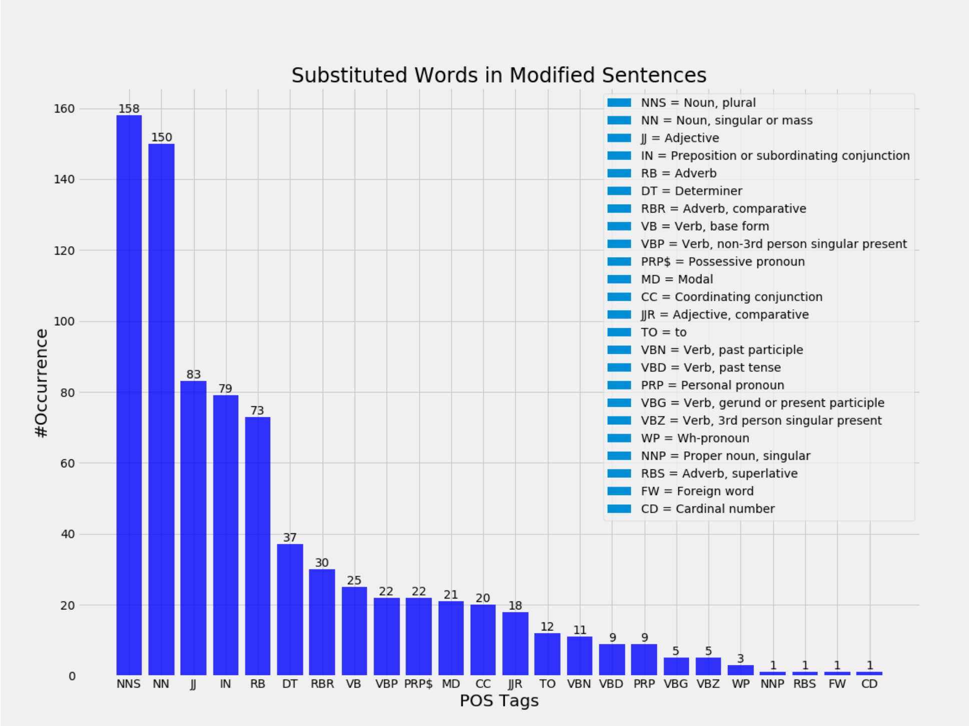


Figure 4.6: POS tags of substituted words in the modified sentences

In the following, we can see two examples of the words substitutions. The first example shows only words substitution without inserting and removing words which seems sequence alignment was not necessary, and the second example shows substitution with inserting and removing words which sequence alignment was applied.

- Original Sentence: *'When a couple is invited to a party, the wife, not the husband, should accept or decline the invitation.'*
- Modified Sentence: *'when a couple is invited to a party, the wife or the husband, can accept or decline the invitation.'*

  POS tags of substituted words are: [('not', 'RB'), ('or', 'CC')] and [('should', 'MD'), ('can', 'MD')]

In the following example first sequence alignment method applied on both sentences, so we can see that the word 'equally' was inserted in the modified sentence and the words 'must', 'obey', and 'him' were removed from the original sentence. Also, the word 'so' was substituted with the word 'as'.

- Original Sentence: *'The husband is - responsible for the family so the wife must obey him'*
- Modified Sentence: *The husband is equally responsible for the family as the wife - - -'*

  POS tags of substituted words are: [('so', 'IN'), ('as', 'IN')]

Table 4.5 displays the top five most frequent substituted words with their POS tags and the mean average rating of the modified sentences.

Rank	Word	POS Tag	Number	Avg Rating
1	Women	Noun, plural	53	3.41
2	Men	Noun, plural	44	4.06
3	Than	Preposition or subordinating conjunction	33	1.57
4	As	Preposition or subordinating conjunction	32	1.28
5	More	Adjective, comparative	29	1.75

Table 4.5: Top five the most frequent substituted words

4.2.2 Which Sequences Were Inserted and Removed

Instead of just inspecting frequencies of POS tags, we can obtain additional information about the syntactic structure of the modified sentences by regarding them as an ordered stream and taking the bigram which is a sequence of two adjacent words, as our basic tokens. In order to see which sequences were inserted and removed in the modified sentences, we get the frequency distribution of every bigram in the original and modified sentences, then we can gain the difference between two sets of bigrams which reveals inserted and removed sequences. To extract a list of bigrams, Python's NLTK library [2] was used which outputs a list of word pairs from a sentence. Moreover, to analyze the syntactic impact of inserted and removed sequences, we get POS tags of each word in a bigram to understand how the syntax of modified sentences was changed in order to remove sexist content from the original sexist sentences.

Figures 4.7 plots the cumulative relative frequency of POS tag bigrams of inserted words in the modified sentences. We can observe that 20% of all POS tag bigrams which include the word "Not"

[2] https://www.nltk.org/book/ch01.html#collocations-and-bigrams

occurred at least ten times.

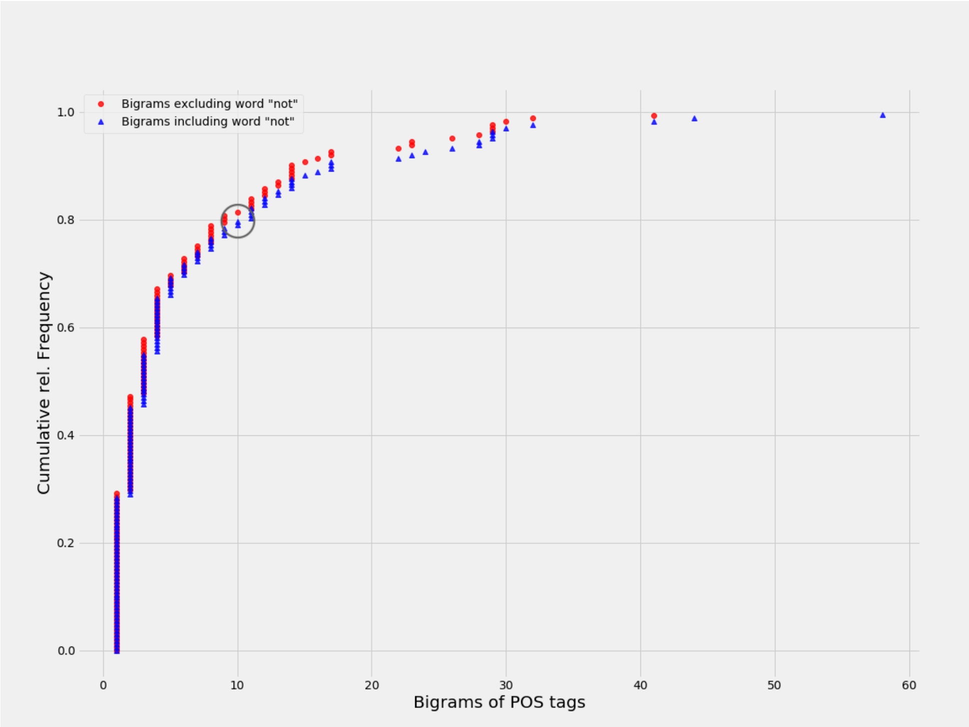


Figure 4.7: Distribution of bigrams of POS tags for inserted words

Table 4.6 splits into two sub-tables which display the first two most frequent POS tags bigrams regarding inserted words with and without the word "Not" respectively. As mentioned earlier, since the word "Not" was used in a very great degree to modify the original sexist sentence, we need to analyze the inserted words POS tag bigrams with and without the word "Not" to have better insight for our sentiment analysis. We make two observations. First, the frequency of bigrams which include the word "Not" is more than the other bigrams without the word "Not", and it tells that players tend to simply insert the word "Not" to remove the sexist content from the original sexist sentence. Second, the average mean rating for the bigrams which the word "Not" were excluded are higher than the bigrams which the word "Not" were included, so it indicates that although players took the risk to lose more scores in the game, they decided to insert the word "Not" to modify the original sexist sentence into a non-sexist sentence. Apart from that, we notice that Nouns played a significant role concerning sequence modification toward words insertion.

	Bigrams including word "Not"			
Rank	**POS Tag Bigram**	**Example**	**Number**	**Avg Rating**
1	(Adverb, Verb - base form)	('not', 'make')	58	1.72
2	(Modal, Adverb)	('should', 'not')	44	1.65

	Bigrams excluding word "Not"			
Rank	**POS Tag Bigram**	**Example**	**Number**	**Avg Rating**
1	(Preposition or subordinating conjunction, Noun - plural)	('as', 'women')	41	2.29
2	(Noun-plural, Verb - non-3rd person singular present)	('men', 'are')	32	3.00

Table 4.6: Bigrams of POS tag of inserted words

Unlike the previous section here we do not need to apply sequence alignment since we want to see which sequences were inserted to the modified sentence. In the following example inserted POS tag bigrams are: [('not', 'RB'), ('make', 'VB')], [('should', 'MD'), ('not', 'RB')]

- Original Sentence: *'Mothers should make most of the decisions on how to bring up their children.'*
- Modified Sentence: *'Mothers should not make most of the decisions on how to bring up their children.'*

Figures 4.8 plots the cumulative relative frequency of POS tag bigram of removed words from the original sentences. We can observe that about 25% of all POS tag bigrams which includes the word "Not" occurred at least ten times.

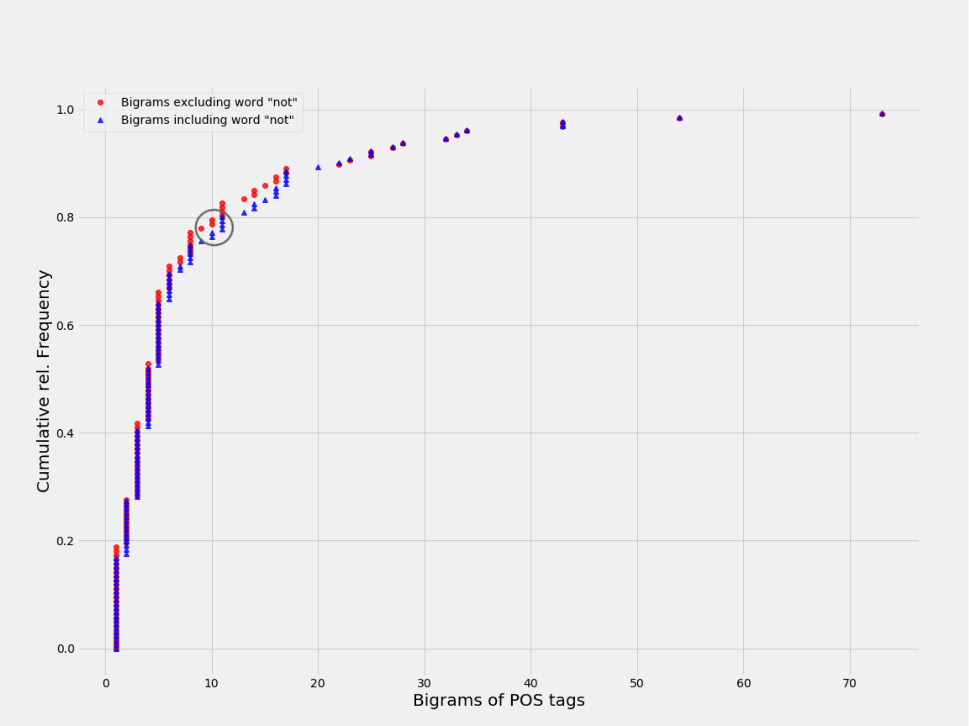


Figure 4.8: Distribution of bigrams of POS tags for removed words

In contrast to the inserted words, the word "Not" did not influence very much in POS tag bigrams of the removed words. Table 4.7 shows the top three most frequent POS tag bigrams of removed words. Here, we can see that players tend to remove those sequences which contain Nouns with both singular and plural form. Also, we can notice that removing bigram of Modal and base form of the Verbs resulted in the best rating number for the top three most frequent POS tag bigrams.

Rank	POS Tag Bigram	Example	Number	Avg Rating
1	(Determiner, Noun - singular or mass)	('the', 'husband')	73	1.82
2	(Modal, Verb - base form)	('should', 'be')	54	1.68
3	(Noun - singular or mass, Preposition or subordinating conjunction)	('mother', 'for')	43	2.25
3	(Preposition or subordinating conjunction, Noun - plural)	('than', 'daughters')	43	2.34

Table 4.7: Bigrams of POS tag of removed words

- Original Sentence: *'Sons in a family should be given more encouragement to go to college than daughters.'*
- Modified Sentence: *'Sons in a family should be given equal encouragement to go to college as daughters.'*

Removed POS tag bigrams of the above example are:
[('college', 'NN'), ('than', 'IN')], [('than', 'IN'), ('daughters', 'NNS')], [('encouragement', 'JJ'), ('to', 'TO')], [('given', 'VBN'), ('more', 'RBR')], [('more', 'RBR'), ('encouragement', 'JJ')]

4.2.3 Which Subsequent Tokens Groups Were Changed

We analyzed the sentences at an intermediate level of abstraction with inspecting the POS tags of words which tells us the words classes such as Nouns, Verbs, Adjectives, etc. However, POS tags did not give us information concerning the structure of the sentences or phrases in the sentences. The chunking, which is also called shallow parsing, divides sentences into syntactically correlated parts of words and enables us to analyze complex parse trees. In order to extract information from the sentences and detecting entities the chunking technique of the NLTK library [3] was used. Chunking technique segments and labels multi-token sequences in the sentences, and chunks(meaningful phrases) abstract away low-level details.

In order to create the chunker, we need to define a chunk grammar which consists of the regular-expression rules that indicate how sentences should be chunked. When all of the chunking rules have been invoked the chunk structure turned out as the result. The chunker takes POS tags as input and provides meaningful phrases. We defined different grammars and regular-expressions to form different chunks such as Noun Phrase (NP), Verb Phrase (VP), and Prepositions Phrase (PP) as output. Figures 4.9 and 4.10 graphically display an example of the chunked original sexist sentence and modified sentence as a tree.

- Original Sentence: *'The husband should be the head of the family.'*
- Modified Sentence: *'The eldest member should be the head of the family.'*

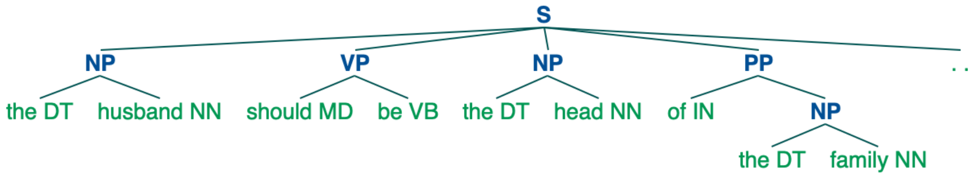


Figure 4.9: Example of an original sentence chunker

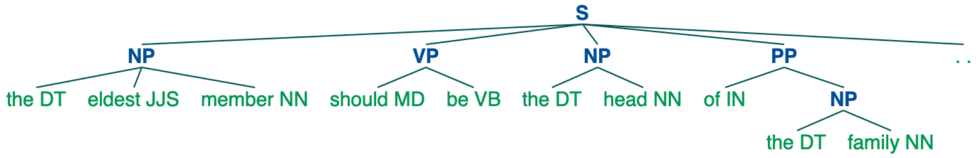


Figure 4.10: Example of a modified sentence chunker

[3] https://www.nltk.org/book/ch07.html#chunking

The chunk grammars are empirically defined. The NP-chunk would be formed when the chunker finds an optional Determiner, followed by any number of Adjectives with different forms, and then a noun also with different forms. Also, the PP-chunk formed when the chunker finds Prepositions followed by NP-chunk, and VB-chunk formed when an optional Modal, followed by an optional Adverb, and then different forms of Verbs. The most frequent chunk pattern among the original sexist sentences is ('NP', 'VP', 'NP', 'PP') by 10%, followed by ('NP', 'VP', 'NP') by 7% and ('NP', 'VP', 'NP', 'NP', 'PP') by 4%. Similar to the original sentences ('NP', 'VP', 'NP', 'PP') by 10% is the most frequent among modified sentences, followed by ('NP', 'VP', 'NP') by 4% and ('NP', 'VP', 'VP', 'NP', 'VP', 'NP', 'PP') by 2%. Moreover, we want to see which syntactic chunk types are modified in order to remove sexist content from the original sexist sentences. Table 4.8 shows the dominance of the noun phrases in both original sentences and the modified versions.

Rank	Label	Chunk Type	Original	Modified(Rating=1)	Modified(Rating=5)
1	NP	Noun Phrase	44%	45%	42%
2	VP	Verb Phrase	37%	36%	39%
3	PP	Preposition Phrase	19%	19%	18%

Table 4.8: Distribution of syntactic chunk types

In order to investigate the syntactic effects, chunk types in the modified sentences with rating one and five are taken into account separately. We can observe that frequency of chunk types in modified sentences with rating one are almost the same as the original sentences. However, verb phrases in the modified sentences with rating five insignificantly increased in comparison to the original sexist sentences. With our empirical chunk grammars, we may conclude that although the distribution of modified chunks did not change significantly, sexist content resides in noun phrases in the modified sentences with rating five despite the frequency of verb phrases raised.

4.2.4 Impact of Restricted Words Modification

One of the advantages of our aligned corpus is that we can generalize the analysis of a particular example to a large set of sentences by identifying the essential words which carry sexist meanings. We check if the sexist content has been successfully removed from an original sexist sentence by modifying certain words, then we realize that these words are crucial to making a sentence as sexist or non-sexist. To this end, we investigate the impact of restricted words insertion and deletion in the modified sentences. Recall that, for the reward mechanism of the first task of the game we introduce two sets of words as restricted words including Gender words and Negative words (in chapter 3.2) which making changes to these words incur more substantial penalties. Now, we want to analyze how changing these restricted words affect the average rating of modified sentences.

Gender Words Modification

Figure 4.11 plots the frequency of inserted and removed gender words in the modified sentence. We can observe that most of the modified sentences have retained the same number of initial gender words, where 85% regarding the insertion and 76% regarding the deletion gender words of modified sentences did not modify the gender words numbers, compared to the original sentence. Moreover, players rarely either insert or remove two or more gender words in their modifications.

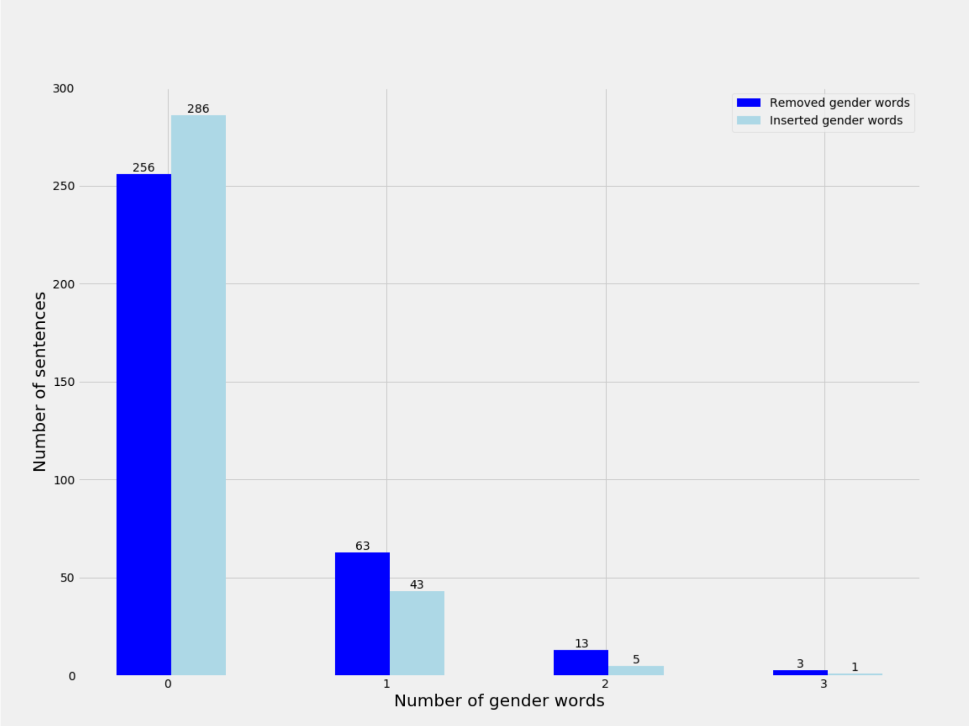


Figure 4.11: Frequency of insertion and deletion gender words

Tables 4.9 shows how changing gender words affect the rating of modified sentences. Generally, gender word deletions compared to insertions lead to better results concerning more desirable rating numbers. Removing two gender words from an original sexist sentence result in the best rating number with 1.30 as average rating, which means these modified sentences are almost non-sexist. Moreover, we can see that removing one gender word from an original sexist sentence has a better average rating score of 2.7, which is an acceptable rating concerning successful removal of sexist content, compared to inserting one gender word with an average rating of 3.27.

Number (inserted/deleted words)	Avg Rating for Insertion	Avg Rating for Deletion
0	2.02	2.10
1	3.27	2.76
2	2.40	1.30
3	4.00	2.00

Table 4.9: Evaluation of inserted and deleted gender words

Negative Words Modification

Similar to the analysis of gender words, we can investigate the effect of negative words concerning the rating of the modified sentences. To this end, first we review the frequency of inserted and removed negative words in the modified sentences, then we evaluate the impact of insertion and deletion on the rating. Figure 4.12 plots the frequency of inserted and removed gender words in the modified sentence. The first observation is that players tend not to change the number of gender words in the original sexist sentences, where 20% of modified sentences had only one negative word inserted, and 8% of modified sentences deleted just one negative word from the original sexist sentences in order to change them into a non-sexist sentence. In fact, the number of insertions and deletions of negative words did not exceed more than one word. Moreover, we can see that players tend to insert rather than deleting a word in their modifications.

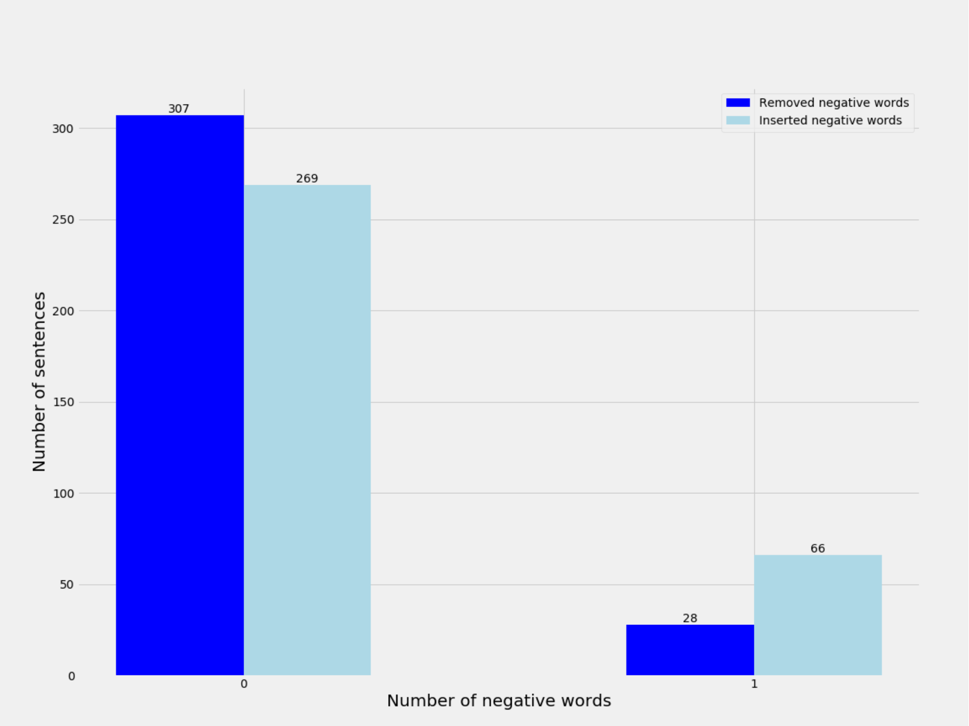


Figure 4.12: Frequency of insertion and deletion negative words

Table 4.10 shows how negative word modifications impact the rating of modified sentences. We can see that the insertion and deletion of one negative word can lead to the proper result, rather than retaining the initial number of negative words in the original sentences.

Number (inserted/deleted words)	**Avg Rating for Insertion**	**Avg Rating for Deletion**
0	2.30	2.26
1	1.72	1.46

Table 4.10: Evaluation of inserted and deleted negative words

4.3 Conclusion

To conclude and summarize this work, the answers to the thesis question as well as the user's feedback about the game are presented.

- **Why a new sexist and non-sexist dataset is needed?**

  Previous research mostly incorporated data by performing an initial manual search to collect tweets which contained sexist content and created a sexist corpus. However, previous research neglected the fact that the method of gathering sexist tweets could be biased towards the initial search terms and also some forms of sexism could be missed since some tweets which contain offensive language would be misclassified as hate speech. Moreover, previous research mostly collected sexist tweets which often expressed only hostile sexism.

  The *UnSexistifyIt* game is an interactive system which allows people to detect sexist content while playing the game. Although the primary purpose of the *UnSexistifyIt* game is to identify sexist sentences and turn them into non-sexist ones, the main contribution of this work is collecting non-sexist data and creating a unique dataset. The generated corpus of this game would be increased over time. Also, it takes various types of sexism into account. Finally, this work successfully generates a new sexist and non-sexist dataset which addresses the limitations among other existing datasets.

- **To what extent the GWAP approach can be used to build a unique dataset?**

  This thesis presents *UnSexistifyIt*, an online game for generating a non-sexist corpus. This work applied the methodology of a "Game-With-A-Purpose" to generate relevant data as a by-product of playing the game. Moreover, this work employed a combination of gamification and crowdsourcing techniques to stimulate a large number of voluntary participants. Further, the findings of this thesis show that the mechanics and dynamics of the game can reveal the sexist content, and also the syntactic analysis explains the structure of a sexist and non-sexist statement.

  The *UnSexistifyIt* game introduces a new corpus which contains 335 non-sexist sentences in total. The dataset initially contains 67 different original sexist sentences, and every sexist sentence has 5 unique modified versions. Modified sentences were rated in a range of one to five, where rating one indicates that the original sexist sentence was successfully turned into a non-sexist sentence, and rating five shows the modified sentence is still sexist.

- **User’s Feedback**

  The players of the games were requested to give their feedback concerning how they observed the game and how they felt about the game. Particularly, three questions were asked.

  1- What was the overall difficulty level of the game?

  The majority of players believe the game was difficult to some degree by 75%, and the other players are divided into two groups with almost equal numbers where 13% thought the game was easy and 12% thought the game was very difficult. In the future, we would try to make the game more involved and easier to play.

  2- Were the game instructions clear?

  Most of the players (73%) believed the game instructions were clear, 18% of players stated that the instructions were moderately clear, and 9% of players thought the game instructions were not clear enough.

  3- How was the game user interface(UI)?

  Favorably, a considerable number of players were satisfied with the user interface design of the game, where 88% of players believe the game user interface was good, and 12% of players thought the UI needs improvement.

Chapter 5

Future Work

A gamified experience is merely a wrapper around the main tasks and cannot turn an unpopular assignment into a favorite one. However, it helps a product to find broader audiences and contributors. Gamification operates well when turning an appealing outcome into a more productive one, with more participants. The following is some ideas and directions for the current status of this work.

- **Develop mechanics of the game**

  We can implement additional incentive mechanisms such as badges, trophies, achievements, etc. The general approach is to configure extra challenges based on players actions which we keep track of, and then reward players with trophies, badges, and achievements when they reach milestones.

  Furthermore, we can improve the current rewarding system of the game for both tasks. Particularly, For the first task of the game, we can introduce an intelligent rewarding system, where the machine can decide how much reward the players can gain rather than a predefined formula. More specifically, we can create a probabilistic classifier for identifying single sentences into sexist or non-sexist classes, and we can use the prediction probability value to compute the reward that the player receives.

- **Employing reinforcement learning**

  The framework of this game allows employing a deep end-to-end reinforcement learning to build intelligent adaptive agents who perform an action which leads to changing the environment state and receiving a reward. Ultimately, the agent maximizes reward value in response to its actions. The general objective of this idea is to introduce a structure that leverages game to learn models of the player's performance in the aspect of sexism detection and reconstruction of the sexist sentence into non-sexist one.

- **Human vs. Machine**

  UnSexistifyIt game currently has two tasks which are performed individually by the players. However, introducing new game plans to the game where players can compete against machines would be an interesting addition to this game. The new game plan is as follows. The player is given a sentence which is not clear whether it is sexist or non-sexist. The player is asked to vote if the given sentence is sexist or non-sexist. If the player votes the given sentence as sexist, the next sentence would be shown to the player and the same process would be repeated. However, if the player votes the given sentence as non-sexist, we ask the player to subtly modify the sentence. Now, if the machine cannot detect sexism in the modified sentence by the player but humans (other players) can, the player gets the maximum number of rewards. If both machine and human can detect sexism in the modified sentence, the players get an average number of rewards. Lastly if neither machine nor human can detect sexism in the modified sentence, the player gets no reward.

Bibliography

[1] Zeerak Waseem and Dirk Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. pages 88–93, 01 2016.

[2] Akshita Jha and Radhika Mamidi. When does a compliment become sexist? analysis and classification of ambivalent sexism using twitter data. 08 2017.

[3] Ziqi Zhang and Lei Luo. Hate speech detection: A solved problem? the challenging case of long tail on twitter. *Semantic Web*, Accepted, 10 2018.

[4] Luis von Ahn and Laura Dabbish. Designing games with a purpose. *Commun. ACM*, 51:58–67, 08 2008.

[5] Luis von Ahn and Laura Dabbish. Designing games with a purpose. 2008.

[6] Luca Galli, Piero Fraternali, and Alessandro Bozzon. On the application of game mechanics in information retrieval. pages 7–11, 04 2014.

[7] Jakub Swacha. Gamification in knowledge management motivating for knowledge sharing. 12:150–160, 01 2015.

[8] Laurentiu Catalin Stanculescu, Alessandro Bozzon, Robert-Jan Sips, and Geert-Jan Houben. Work and play: An experiment in enterprise gamification. pages 345–357, 02 2016.

[9] Jennifer Thom-Santelli, David Millen, and Joan Morris DiMicco. Removing gamification from an enterprise sns. pages 1067–1070, 02 2012.

[10] Yu Chen and Pearl Pu. Healthytogether: exploring social incentives for mobile fitness applications. 04 2014.

[11] Carsten Eickhoff. Crowd-powered experts: helping surgeons interpret breast cancer images. pages 53–56, 04 2014.

[12] Karl Kapp. *The gamification of learning and instruction: Game-based methods and strategies for training and education. San Francisco, CA: Pfeiffer*. 01 2012.

[13] Isabella Kotini and Sofia Tzelepi. *A Gamification-Based Framework for Developing Learning Activities of Computational Thinking*, pages 219–252. 10 2015.

[14] Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. pages 427–431, 01 2017.

[15] Robert West and Eric Horvitz. Reverse-engineering satire, or "paper on computational humor accepted despite making serious advances", 01 2019.

[16] Kristen Dergousoff and Regan L. Mandryk. Mobile gamification for crowdsourcing data collection: Leveraging the freemium model. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, pages 1065–1074, New York, NY, USA, 2015. ACM.

[17] Jeff Howe. *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*. 01 2009.

[18] Jeffrey Heer and Michael Bostock. Crowdsourcing graphical perception: Using mechanical turk to assess visualization design. volume 1, pages 203–212, 01 2010.

[19] Steven Komarov, Katharina Reinecke, and Krzysztof Z. Gajos. Crowdsourcing performance evaluations of user interfaces. pages 207–216, 04 2013.

[20] Winter Mason and Siddharth Suri. A guide to conducting behavioral research on amazon's mechanical turk. *Behavior research methods*, 44:1–23, 06 2011.

[21] Massimo Poesio, Jon Chamberlain, Udo Kruschwitz, Livio Robaldo, and Luca Ducceschi. Phrase detectives: Utilizing collective intelligence for internet-scale language resource creation (extended abstract), 2015.

[22] Irene Garcia-Martí, Luis E Rodríguez, Mauri Benedito-Bordonau, Sergi Trilles Oliver, Arturo Beltran, Laura Díaz, and Joaquín Huerta. Mobile application for noise pollution monitoring through gamification techniques. 09 2012.

[23] Luis von Ahn and Laura Dabbish. Labeling images with a computer game. *In: , Vienna, Austria,*, 319–326, 08 2004.

[24] David R. Flatla, Carl Gutwin, Lennart Nacke, Scott Bateman, and Regan Mandryk. Calibration games: Making calibration tasks enjoyable by adding motivating game elements. pages 403–412, 10 2011.

[25] Maik Schacht and Silvia Schacht. *Start the Game: Increasing User Experience of Enterprise Systems Following a Gamification Mechanism*, pages 181–199. 08 2012.

[26] Jenni Majuri, Jonna Koivisto, and Juho Hamari. Gamification of education and learning: A review of empirical literature. 05 2018.

[27] Janet T. Spence. *The attitudes toward women scale : an objective instrument to measure attitudes toward the rights and roles of women in contemporary society*. not identified, 1972.

[28] Janet Swim, Kathryn Aikin, Wayne S. Hall, and Barbara Hunter. Sexism and racism: Old-fashioned and modern prejudices. *Journal of Personality and Social Psychology*, 68:199–214, 02 1995.

[29] Peter Glick and Susan Fiske. The ambivalent sexism inventory: Differentiating hostile and benevolent sexism. *Journal of Personality and Social Psychology*, 70:491–512, 03 1996.

[30] Francine Tougas, Rupert Brown, Ann M. Beaton, and Stéphane Joly. Neosexism: Plus Ça change, plus c'est pareil. *Personality and Social Psychology Bulletin*, 21:842–849, 08 1995.

[31] Lynda A. King and Daniel King. Sex-role egalitarian ism scale. *Psychology of Women Quarterly*, 21:71 – 87, 07 2006.

[32] Eduardo García-Cueto, Francisco Rodríguez-Díaz, Carolina Bringas Molleda, Javier Borrego, Susana Paíno Quesada, and Luis Rodríguez-Franco. Development of the gender role attitudes scale (gras) amongst young spanish people. *International Journal of Clinical and Health Psychology*, 15, 11 2014.

[33] Shirley Rombough and Joseph C. Ventimiglia. Sexism: A tri-dimensional phenomenon. *Sex Roles*, 7:747–755, 01 1981.

[34] Gonzalo Navarro. A guided tour to approximate string matching. *ACM Computing Surveys*, 33, 04 2000.

[35] What is mongodb? `https://www.mongodb.com/what-is-mongodb`.

[36] What is mongoose? `https://www.npmjs.com/package/mongoose`.

[37] What is express.js? `https://nodejs.org/en/about/`.

[38] What is node.js? `https://nodejs.org/en/about/`.

[39] What is angularjs? `https://angularjs.org/`.

[40] VI Levenshtein. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady*, 10:707, 1966.

[41] Saul B. Needleman and Christian D. Wunsch. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, 48(3):443 – 453, 1970.