On the fairness of crowdsourced training data and Machine Learning models for the prediction of subjective properties. *The case of sentence toxicity.*

To be or not to be #$&%! toxic? To be or not to be fair?*

Agathe Balayn

TU Delft

On the fairness of crowdsourced training data and Machine Learning models for the prediction of subjective properties. *The case of sentence toxicity.*

To be or not to be #$&%! toxic? To be or not to be fair?*

by

Agathe Balayn

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday September 27, 2018 at 11:00 AM.

Student number:	4620844
Project duration:	November 1, 2017 – September 27, 2018

Thesis committee:

Chair	Prof. dr. G.-J. Houben, Faculty EEMCS, TU Delft
University supervisor	Dr. A. Bozzon, Faculty EEMCS, TU Delft
Committee member	Dr. G. Gousios , Faculty EEMCS, TU Delft
Company supervisor	B. Timmermans, Center for Advanced Studies IBM Benelux

TU Delft

IBM Centre for Advanced Studies

Preface

Training machine learning (ML) models for natural language processing usually requires lots of data that is often acquired through crowdsourcing. In crowdsourcing, crowd workers annotate data samples according to one or more properties, such as the sentiment of a sentence, the violence of a video segment, the aesthetics of an image, ... To ensure quality of the annotations, several workers annotate the same sample, and their annotations are combined into one unique label using aggregation techniques such as majority voting.

When the property to be annotated by the workers is subjective, the workers' annotations for one same sample might differ, but all be valid. The way the annotations are aggregated can have an effect on the fairness of the outputs of the trained model. For example only accounting for the majority vote leads to ignoring the workers' opinions which differ from the majority and consequently being discriminative towards certain workers. Also, ML models are not always designed to account for individual opinions, for simplicity's or performance's sake. Finally, to the best of our knowledge, no method exists to assess the fairness of a ML algorithm predicting a subjective property. In this thesis we address such limitations by seeking an answer to the following research question: how can targeted crowdsourcing be used to increase the fairness of ML algorithms trained for subjective properties' prediction?

We investigate how annotation aggregation via majority voting creates a dataset bias towards the majority opinion, and how this dataset bias in combination with the current limits of ML models lead to an algorithmic bias of the ML models trained with this dataset and unfairness in the model's outputs. We assume that an ML model able to return each annotation of each user is a fair model. We propose a new evaluation method of the ML models' fairness, and a methodology to highlight and mitigate potential unfairness based on the creation of adapted training datasets and ML models. Although our work is applicable to any kind of label aggregation for any data subject to multiple interpretations, we focus on the effects of the bias introduced by majority voting for the task of predicting sentence toxicity.

Our results show that the fairness evaluation method that we create enables to identify unfair algorithms and compare algorithmic fairness, and the final fairness metric is usable in the training process of ML models. The experiments on the models point out that we can mitigate the biases resulting from majority voting and increase the fairness towards the minority opinions. This is provided that the workers' individual information and each of their annotations are taken into account when training adapted models, rather than only relying on the aggregated annotations, and that the dataset is resampled on criteria according to the favoured aspect of fairness. We also highlight that more work needs to be done to develop crowdsourcing methods to collect high-quality annotations of subjective properties, possibly at low-cost.

Agathe Balayn
Delft, September 2018

Acknowledgement

I would like to acknowledge every person and institution with who I spent time during this year. Without the support of many people, and without the Delft University of Technology and the Center for Advanced Studied of the IBM Benelux, the completion of this thesis would have not been possible.

Particularly, I would like to thank my thesis supervisors from the university Alessandro Bozzon and from the company Benjamin Timmermans for their frequent feedbacks. They always listened to my explanations about my work and answered questions about my research. They also read my long thesis report and gave valuable comments about its structure and the scientific writing.

I would also like to thank the other people with who I had detailed discussions about my work, mainly Zoltán Szlávik who took the time to listen to my questionings about the fairness metric, Panagiotis Mavridis who brought a lot of insights into the crowdsourcing work and for the writing of the CrowdBias paper, Lora Aroyo who shared her point of view on the use of her CrowdTruth framework, and Pavel Kucherbaev for the initial supervision of the project. I also wish to acknowledge the other members of the committee, Geert-Jan Houben and Georgios Gousios without who I would not be able to defend my thesis.

Furthermore, working at the CAS of the IBM Benelux was undoubtedly a great chance for me to interact with more people during the long time of the thesis project. On this, I wish to acknowledge the other interns and my friends from university who made the thesis time very fun and more motivating all along the way.

Finally, I must express my profound gratitude to my parents and siblings for always providing me with support and continuous encouragements throughout the thesis year at the TU Delft. This accomplishment would not have been possible without them.

Thank you.

Agathe Balayn
Delft, the Netherlands,
September 2018

Contents

List of Figures

List of Tables

1 Introduction

Machine Learning (ML) aims at predicting properties of new data by learning correlations between available data samples and their known properties. For example, certain ML algorithms are built to classify radiology images depending on whether they show a cancerous tumour, by learning the existing correlations from available images and their labels (existence or not of a cancerous tumour). This is a traditional classification task for Machine Learning, whose objective is well defined as it is clear that each image belongs to only one possible class (with or without tumour). More and more ML algorithms are now used to address tasks whose objectives are highly disputable because the predictions can not be verified by a human [80], [96]. For example a ML algorithm made to predict whether an individual convicted of a crime might be a repeat offender is disputable because the future prediction is not verifiable by a human since the future is not known. The purpose of these predictions is to classify human beings in order to decide how these persons will be treated, what potentially has a negative impact on humans' lives [5]. Thus the accuracy of the predictions is important not to harm someone wrongly.

Most research papers are considered as progress in their field when they report high accuracies of their ML algorithms, whereas their outputs might be biased, unfair or discriminative towards certain categories of population [52]. For example, the COMPAS system (Correctional Offender Management Profiling for Alternative Sanctions) made to predict a defendant's risk of re-offending was proved to be discriminative because it labels Black people twice more often than White people as potential reoffenders whereas it is not the case in reality [1] [114]. It is claimed that it is unethical not to analyse the outputs' errors and that systematic error analysis would improve the understanding, the transparency and the accountability of the algorithms, instead of simply reporting accuracy performance [52]. Therefore, investigating the outputs of Machine Learning algorithms made for predicting disputable properties of samples and their potential unfairness power is becoming an increasingly important task.

In this thesis project, we identified a subset of the disputable tasks on which to focus on. This subset is the group of tasks interested in classifying subjective properties of samples: properties for which there is no consensus between the judgements of different people. We made this choice because we consider it is important to tackle these tasks properly since we assume that low performance on these tasks translate into high risk of harmful consequences.

ML models usually require a lot of data to be trained on and the methods to get training data often involve crowdsourcing. "Crowdsourcing is the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call" [61]. For example, crowdsourcing is used to answer queries unanswerable by computers (database systems or search engines) answered by the crowd (CrowdDB framework [48]), to assess designs of visualization [56], to create labels for natural language tasks [97], to perform user studies [68].

One of the main research questions for Crowdsourcing is: how to ensure that the annotations provided by the annotators are correct? [8, 42] The quality of the annotations (often measured with the agreement rate) depends on three factors: the expertise of the annotators, the quality of the samples to annotate, and the way the task is presented. This may be because the annotators might not provide accurate annotations due

[1] `https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing`

to a lack of expertise or a lack of consideration for the task (spammers or annotators who make mistakes from inattention). The task might not be described precisely enough and the samples to annotate might be ambiguous or the property to annotate subjective, what leads the annotators to give different answers that they all consider valid.

The annotation quality problem is resolved using methods directly implemented on the crowdsourcing platform and post-processing methods of crowdsourcing treatment. For example, it is shown that in order to create datasets for Machine Learning tasks using crowdsourcing, a large crowd of non-expert annotators enables the training of classifiers with a higher accuracy than using unique annotations from experts. This is provided that specific metrics are used to aggregate the annotations of multiple annotators into a unique label [41], and when accounting for the annotators' quality [97]. Research on crowdsourcing for quality assessment experiments also shows similar results to laboratory experiments [67]. Several frameworks were created to realize subjective multimedia evaluations and Quality of Experience evaluations [26, 60, 83, 84, 88] with promising results in many domains such as video quality evaluation [47].

These methods all return one unique label per sample, which might not be representative of the annotated property in cases where several interpretations of the sample are valid (our selected case of the classification of subjective properties); this might lead to unfairness since certain annotators' judgements are ignored. The annotators might express different but all valid judgements about the property for several reasons: the task or the samples to annotate might be ambiguous leading to different interpretations from different annotators; or the property to annotate the sample on might be subjective (for example the annotators might be asked to express their feelings, perception, opinion, aesthetic judgement about a sample), and because the annotators have different backgrounds, they have diverging points of view on the property of the sample. Although the first two causes of diversity in judgements for a same sample might be eliminated by refining the crowdsourcing task, the third cause (subjectivity of the property to annotate and related people's subjectivity) cannot, because it is intrinsic to the property of the data sample and to workers. Consequently it is important to investigate the field of crowdsourcing for the annotation of subjective properties.

In this chapter, we identify the problem that we tackle in the thesis project, and its attached challenges. We turn them into a set of research questions, and expose the contributions that we choose to make while answering these questions.

1.1. Problem studied in the thesis: Machine Learning, Crowdsourcing, need of data, biases and fairness

In this section, we explicit the three main limitations (summed up in Fig. 1.1) related to the fairness of ML systems which aim at predicting subjective properties of samples.

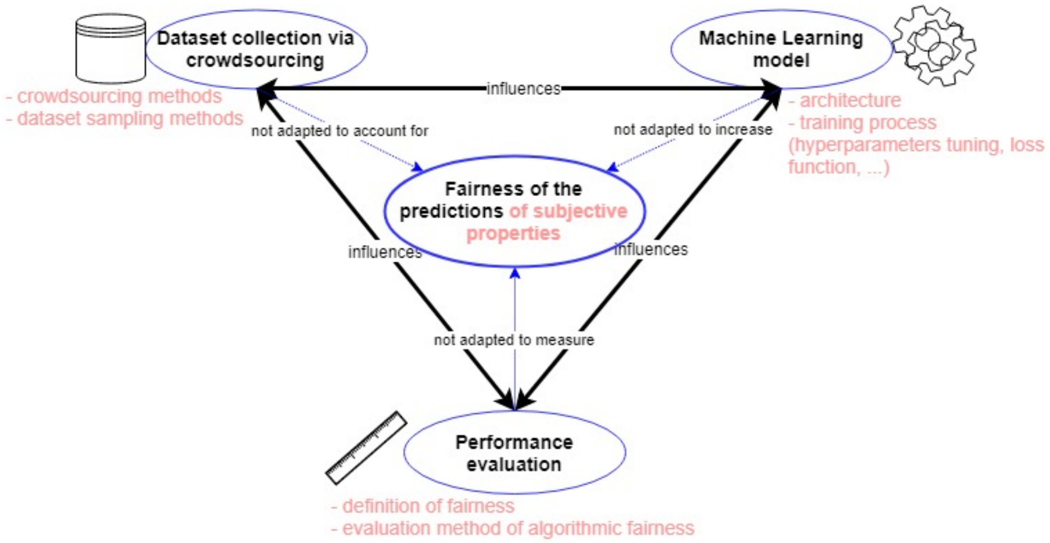

Dataset collection via crowdsourcing
- crowdsourcing methods
- dataset sampling methods
influences
Machine Learning model
- architecture
- training process (hyperparameters tuning, loss function, ...)
not adapted to account for
not adapted to increase
Fairness of the predictions of subjective properties
influences
influences
not adapted to measure
Performance evaluation
- definition of fairness
- evaluation method of algorithmic fairness

Figure 1.1: The three limitations related to the fairness of current Machine Learning systems for the prediction of subjective properties.

1.1.1. Limitations of current systems for the prediction of subjective properties

The prediction of subjective properties of samples is one of the disputable tasks that ML algorithms are applied to. In these applications, it is not always possible to consider that a unique label is enough to describe a sample correctly because different people might express different but all valid judgements about the sample.

ML algorithms are not adapted to the task at stake because they are made to be trained on unique labels and consequently can not deal with the subjectivity of the property, this possibly making them unfair (limitation 1). These algorithms would merit being adapted to the use of multiple labels. For example certain systems are interested in classifying video segments into categories such as violent, non-violent [93], and violence is perceived differently depending on the age, gender, etc. of the person watching a video, thus the labels might differ depending on the judge of the video. Similarly sentiment of sentences is subjective, sentences are interpreted differently depending on who reads them and consequently sentiment labels might show diversity. Certain researchers are interested in evaluating the aesthetics of images [11] while humans are shown to feel differently about a same image, and thus the collected judgements about a same image diverge. For all these tasks, using one unique label (violent/non-violent, positive/negative, aesthetic/not aesthetic) to describe the samples (video segment, sentence, image) at stake would not take into account the subjectivity of the property to classify on (violence, sentiment, aesthetic) and the subjectivity of people.

In all these example applications, it is implicitly assumed that current ML models ignore certain judgements because they only output one unique judgement, and thus they are considered unfair towards the people who did not emit these judgements. However, this specific notion of algorithmic unfairness is not studied in the literature: there is neither a definition of fairness for these models, nor an adapted evaluation method of algorithmic fairness. ***This lack of research and understanding about algorithmic fairness do not enable the systematic investigation of the fairness of ML systems (limitation 2).***

In order to train these algorithms, crowdsourcing tasks are set-up, that collect annotations considered as labels for the samples in the dataset, so that the complete dataset is used in the training process. ***The crowdsourcing techniques which aggregate the annotations into unique labels to increase their quality are not valid for our specific domain of application, because all the valid perceptions available about each sample should be taken into account but this information is lost during the aggregation, what leads to unfairness towards the ignored perceptions (limitation 3).*** That is why it is important to investigate whether it is possible to collect high-quality annotations via crowdsourcing without using traditional aggregation methods to filter out low-quality annotations. For example, toxicity is a subjective property of sentences and consequently if several annotators are asked to annotate the toxicity of sentences, they might have different perceptions of the toxicity of some or all of the sentences. Examples of sentences and judgements about their toxicity are given in Table 1.1. Aggregating the annotations into unique labels would conduct to consider only one perception as valid, what is unfair to the other perceptions.

sample	annotations	label
Is there perhaps enough newsworthy information to make an article about the Bundy family as a whole, that the various family members can be redirected to? Or does that violate a guideline I'm not aware of?	non-toxic (100%)	non-toxic
What shit u talk to me, communist rat?	toxic (100%)	toxic
"Please relate the ozone hole to increases in cancer, and provide figures. Otherwise, this article will be biased toward the environmentalist anti-CFC point of view instead of being neutral. Ed Poor"	toxic (20%), non-toxic (80%)	non-toxic?
The article is true, the Israeli policies are killing Arab children.	toxic (50%), non-toxic (50%)	?

Table 1.1: Example samples of a Machine Learning dataset for the task of predicting sentence toxicity. When a sentence has multiple valid annotations about its toxicity, aggregating them into one unique label results in ignoring certain opinions.

1.1.2. Consequences from the combination of Machine Learning and Crowdsourcing

According to Dr. Yoshua Bengio, "Machine learning research is part of research on artificial intelligence, seeking to provide knowledge to computers through data, observations and interacting with the world. That acquired knowledge allows computers to correctly generalize to new settings."[2]. This definition highlights

[2] `https://www.techemergence.com/what-is-machine-learning/`

the need for data -data samples and their corresponding labels- in the field of ML, need which is even more important for Deep Learning algorithms. Consequently, the training of most Machine and Deep Learning algorithms starts with a crowdsourcing phase to constitute large datasets. The pipeline is depicted in Fig. 1.2.

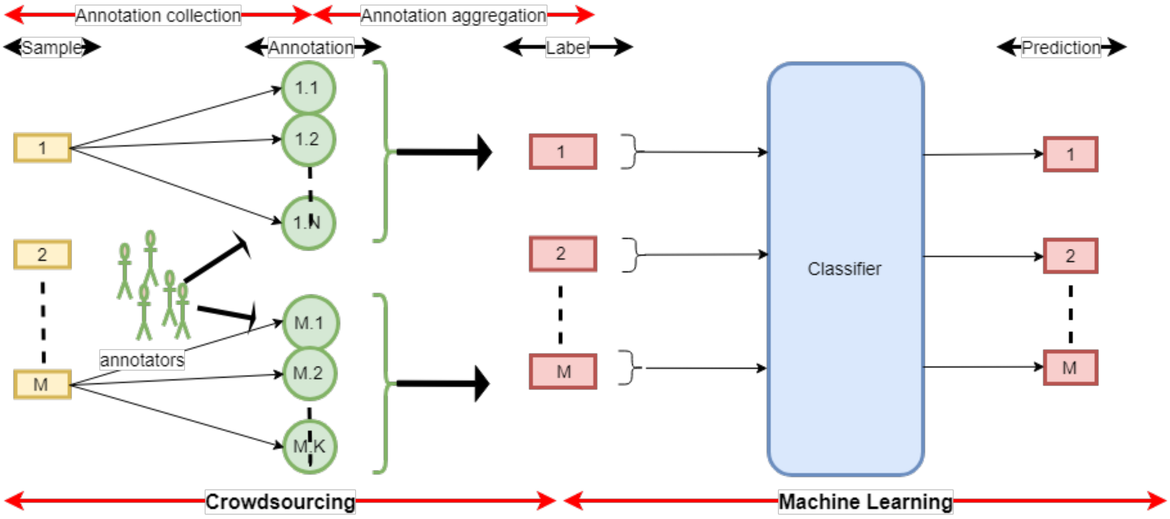


Figure 1.2: Usual Machine Learning pipeline combined with crowdsourcing. The training dataset is collected via crowdsourcing by aggregating the annotations of multiple annotators into labels. The Machine Learning model is then trained on this dataset.

The three above-cited limitations of ML systems for the prediction of subjective properties, put together, ca make certain systems highly unfair. The annotation aggregation creates a ***dataset bias*** towards one of the perceptions of the property. The algorithms, not adapted to the task, and trained on these inappropriate labels consequently exhibit an ***algorithmic bias*** usually towards the majority perception, resulting into ***algorithmic unfairness*** towards the other perceptions, usually the perceptions of the minorities. For example, in the third example of Table 1.1, 50% of the annotations have the label "toxic" while the other 50% are labelled "non-toxic". This might be because a part of the population agrees with the statement of the sentence while the other part disagrees. If one unique label such as "toxic" was selected, the judgements of 50% of the population would be ignored (dataset bias). The Machine Learning model which would be trained on this sample would then be biased towards the first type of judgement (algorithmic bias) and its outputs would ignore the judgements of 50% of the population (unfairness towards a subset of the population). Finally the lack of proper definition and evaluation method of this algorithmic unfairness does not help researchers to identify and tackle the issues related to unfairness when predicting subjective properties.

In order to make predictions of algorithms closer to reality and consequently more fair, we stress it is necessary to investigate how to integrate the subjectivity of the property to annotate and the subjectivity of the crowd in Crowdsourcing tasks, in further ML algorithms, and in the evaluation methods of the performance of the systems. According to the above pipeline, Machine Learning and Crowdsourcing are currently two separate fields of research that are used sequentially. However, we consider that for algorithms fulfilling the disputable task of predicting subjective properties of samples, the two fields have to be brought together in order to increase the fairness of the predictions. Since the labels in the training set can not be defined by a simple aggregation of annotations anymore, we hypothesize that the multiple annotations should be directly used by adapted ML algorithms. We argue for models' architectures enable to use multiple labels per sample: a Machine Learning model able to return different annotations for a same sample depending on the judgement of the person currently using the model would be fair toward each user.

1.2. Main research question, challenges, hypothesis and use-case

Several **main challenges** to investigate are brought in three main fields when making algorithms which classify subjective properties fairer:

- ***CH1:*** Dataset collection by crowdsourcing: How to ***ensure crowdsourced labels' quality*** without aggregating the annotations into unique labels?

- ***CH2:*** Machine Learning model:
  - ***CH2.1:*** How to ***adapt Machine Learning algorithms' architecture*** to return multiple labels depending on the user of the algorithms?
  - ***CH2.2:*** How to ***build datasets*** to train these algorithms?
- ***CH3:*** Performance evaluation: How to ***measure the fairness*** of the algorithms quantitatively?

"Targeted crowdsourcing" is introduced by Ipeirotis et al. [63] as crowdsourcing where the crowd workers with the needed expertise are identified during the task. Specifically, we call in the rest of the thesis "targeted crowdsourcing" crowdsourcing tasks which take into account the available properties of the annotators' background during annotation collection. We believe that these properties are important to make algorithms fairer at classifying subjective properties since subjectivity is intrinsic to each annotator.

The **main research question** (RQ) that follows is:

> RQ: ***How can targeted crowdsourcing be used to increase the fairness of Machine Learning algorithms trained for subjective properties' prediction?***

The **main hypothesis** we test is the following:

> H: ***Even if annotation aggregation enables to eliminate annotation mistakes and spammers, when the annotations of subjective properties differ, but are all considered valid, there is also a loss of information that leads to decrease of fairness in ML results. Therefore using disaggregated labels to train adapted algorithms on adapted datasets should increase their fairness.***

We could study different prediction tasks, as long as they involve subjective properties with several valid interpretations over a unique sample. We chose the **use-case** of ***predicting toxicity of sentences*** for the following reasons.

- Toxicity is a subjective property which depends on people's perception of a sentence characteristics and context, as well as on people's own subjectivity (mainly influenced by their background).
- Prediction of sentence toxicity is useful for several purposes such as to build well-behaved chatbots or to filter offensive Web content since the use of hate speech over the Internet has increased with the growth of the Internet. A short reflection about the ethical issues related to the automatic prediction of sentence toxicity is proposed in Appendix D.
- No study has previously been performed to study the collection of toxicity annotations using crowdsourcing, nor to automatically predict toxicity depending on people's subjectivity.
- On the contrary to other tasks like sentiment analysis or subjectivity annotation (determining whether a snippet is subjective or not) [62], it is hard to define a category of people who are experts at toxicity judgement, thus crowdsourcing appears as a suited way of collecting a toxicity dataset.

Therefore, detecting and understanding toxicity and its subjectivities is an interesting use case to apply our experiments on the prediction of subjective properties using crowdsourcing and Machine Learning.

1.3. Research questions of the research project

In order to answer the main RQ, we divide it into the following research questions and their research subquestions, each one corresponding to a specific challenge cited previously. We explicit the methods employed to answer each of the questions and the results we hope to find.

- ***RQ1: How can a dataset be built to train algorithms for the prediction of subjective properties?***
  This question aims at answering the first challenge.
  First it is required to choose an existing dataset which contains subjectivity. The analysis of the psychology literature about sentence toxicity shows that toxicity is a subjective property in theory and enables to identify the major variables which influence the perception of sentence toxicity. This leads to the

creation of a list of requirement to select the Computer Science dataset for the study of toxicity in practice. Experiments on the selected dataset (Jigsaw dataset) proves that it is adapted to study subjective properties prediction.
Second, we study how to collect datasets containing subjectivities using crowdsourcing while ensuring high-quality data and low-cost. A literature review of existing crowdsourcing methods compared to the requirements of the selected task and dataset, and experimentations on the retained method which does not aggregate annotations into unique label, show that applying different crowdsourcing methods enables to filter out part of the invalid collected annotations, but that more work is needed to identify the rest of the annotators' mistakes. Experiments on automatic clustering of the data samples open a possibility to refine for the grouping and evaluation of a reduced number of samples by selected annotators to decrease the cost of the crowdsourcing task.

- ***RQ2: How to evaluate algorithmic fairness when predicting subjective properties?***
  This question aims at answering the third challenge.
  A literature review focused on algorithmic fairness in ML shows that current definitions and evaluation methods are not adapted to algorithms made to classify subjective properties. This conducts to propose a new definition which generalize the concepts mentioned in the literature, and to investigate new evaluation methods. For that, we set-up a list of algorithms with different expected fairness-related behaviours and propose different ways to characterize and visualize their potential unfairness by clustering the dataset according to multiple criteria and evaluating the algorithms' performance on each cluster separately. We select the characterizations which enable to observe the expected fairness-related behaviours and make several hypotheses to summarize these characterizations into unique fairness measurements. We again select the ones which highlight the expected behaviours.

- ***RQ3: How to build and train algorithms whose outputs are fair when predicting subjective properties of samples?***
  This questions aims at answering the second challenge.
  A completely fair algorithm is assumed to be an algorithm which outputs accurately the perceptions of the current judge of the samples. An extensive review of the literature which aims at classifying toxic (or related) speeches is conducted to identify potential algorithms to adapt, and a search of ML algorithms made to predict different outputs about one same sample depending on certain criteria is done to find potential ways to adapt the outputs to the different crowd workers. Hypotheses are formulated concerning the training processes and architectures of the models, as well as on resamplings of the training dataset, in order to make their predictions fairer. They are evaluated by applying the hypotheses to a default ML model and comparing the performance of the default model to this new model using the previously defined fairness evaluation method. It is concluded that augmenting the algorithms' inputs using variables describing the users' background information and resammpling the training dataset according to the fairness criteria which we want to optimize enables to make the algorithms fairer. However, the performance are not totally accurate and additional modifications of the models should be made to increase the fairness even more.

1.4. Thesis contributions

In this thesis we set-up to create a methodology aimed at evaluating the fairness of the outputs of classifiers for subjective properties, and make them fairer. The methodology tackles the metrics to evaluate fairness, as well as fairness related to crowdsourced data and ML algorithms. It consists in three main steps: the dataset creation phase, the algorithm design and training phase, the evaluation of the fairness of the predictions phase. We bring five main contributions that we list here.

- ***CO1:*** The first contribution of the thesis is an ***extensive literature review*** to study the fairness of ML algorithms trained for subjective properties prediction, with the use-case of sentence toxicity. We investigate existing literature on each of these sub-topics which enable to highlight current limitations, as well as possible directions to improve the fairness of subjective properties classification algorithms as well as their training data (study of crowdsourcing for subjective property annotations). It enables to answer the first sub-questions of each research question (RQ).

- ***CO2:*** The second contribution answers the first research question (RQ1). It consists in a ***list of recommendations on the collection and cleaning of a toxicity dataset*** for further training of ML algorithms.

- ***CO3:*** The third contribution is an ***evaluation method to measure the fairness*** of ML algorithms made to predict subjective properties on samples. This contribution is the answer to the second research question (RQ2).
- ***CO4:*** The fourth contribution is a ***modification of current ML and Deep Learning algorithms' architectures*** so that their fairness as defined in the second contribution is improved. This is the answer to the third research question (RQ3).
- ***CO5:*** Finally, the last contribution which answers part of the third research question (RQ3) is a ***set of dataset resampling methods*** to modify training sets and increase the fairness of the ML algorithms which are trained on these sets.

The first contribution leads the reflection on the different aspects of the main RQ and enables to formulate hypotheses to answer it. Contributions 2) and 5) tackle the targeted crowdsourcing aspect of our main RQ, they enable to identify how to use crowdsourcing to create datasets to train fair Machine Learning algorithms made to predict subjective properties of samples. Contribution 4) tackles the ML aspect of the RQ. Combining contributions 2), 4) and 5) forms the main elements of the methodology to make the algorithms' outputs fairer, while contribution 3) enables to evaluate quantitatively algorithmic fairness so that future research to improve the current methodology could be objectively compared to our proposition.

1.5. Thesis outline

The thesis is organized as follows. We first proceed to a literature review of the different fields concerned with our research questions (psychology, Machine Learning, crowdsourcing and algorithms' fairness mainly) in order to answer the first sub-questions of each research question (RQ) (Chapter 2). Then we tackle the first research question (RQ1): we show the validity of studying sentence toxicity to work on the automatic prediction of subjective properties, and investigate the crowdsourcing processing steps to create datasets adapted to train algorithms for subjective property prediction (Chapter 3). Afterwards, we focus on the problem of creating an algorithmic fairness' evaluation method (Chapter 4). In the next chapter, we work on the creation of Machine Learning algorithms to realize the automatic classification of subjective properties and we investigate dataset resampling to improve the performance of these algorithms (Chapter 5). The fairness metric and Machine Learning steps are entangled chronologically since we do not have a baseline for our task and thus creating an adapted evaluation metric and testing it on adapted algorithms is an iterative process. Finally, we discuss the overall results and give suggestions for future work (chapter 6).

Fig. 1.3 presents an overview of the thesis work and organization.

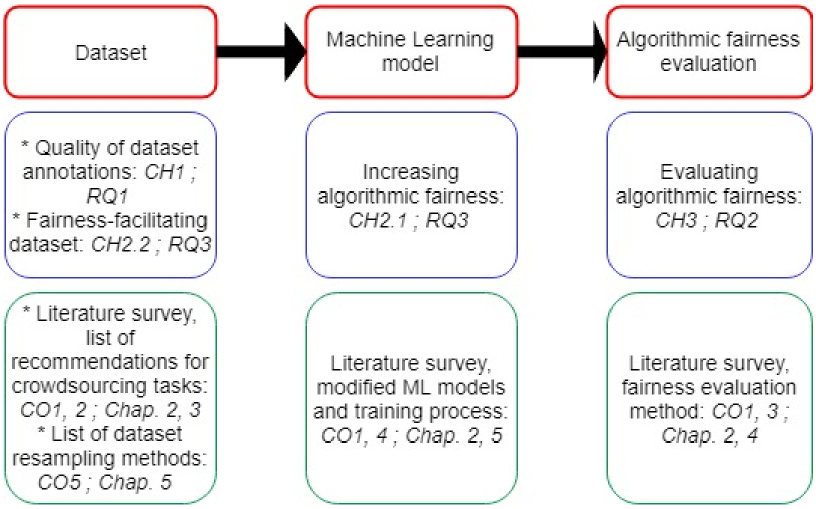

Dataset
Machine Learning model
Algorithmic fairness evaluation
* Quality of dataset annotations: *CH1 ; RQ1*
* Fairness-facilitating dataset: *CH2.2 ; RQ3*
Increasing algorithmic fairness: *CH2.1 ; RQ3*
Evaluating algorithmic fairness: *CH3 ; RQ2*
* Literature survey, list of recommendations for crowdsourcing tasks: *CO1, 2 ; Chap. 2, 3*
* List of dataset resampling methods: *CO5 ; Chap. 5*
Literature survey, modified ML models and training process: *CO1, 4 ; Chap. 2, 5*
Literature survey, fairness evaluation method: *CO1, 3 ; Chap. 2, 4*

Figure 1.3: Overview of the thesis project.

2 Literature review

To tackle our main research question, we start by conducting a literature survey in order to investigate the current state-of-the-art in the area of toxicity detection both in the psychology and Computer Science fields, as well as in the areas of fairness of Machine and Deep Learning algorithms and crowdsourcing for subjective tasks. It enables to answer the first sub-sub-question of each question, and it corresponds to contribution 1.

We begin by looking at how ***toxicity*** (RQ1) and its related topics are studied in the field of psychology. The aim of this first part is double, it is 1) to verify that sentence toxicity is a subjective property and identify which variables influence its perception, and 2) to understand what are the necessary elements a training dataset for toxicity prediction should contain, and how to build such a dataset.

Next we are interested in the ***Machine Learning*** (RQ3) part of the work. On the one hand, we review the existing Machine Learning approaches for toxicity prediction and the Machine Learning algorithms which target prediction of subjective tasks in order to find ways to perform subjective toxicity prediction ; on the other hand we study the methods used to create the datasets employed in these research in order to reuse or create our own dataset (RQ1). Additionally, we investigate what are the evaluation metrics used to measure the performances of these algorithms and the Machine Learning metrics aiming at evaluating algorithmic fairness and algorithms' discrimination power, with the purpose of defining an evaluation method adapted to the prediction of the fairness of toxicity classification algorithms (RQ2).

Finally, we study how ***crowdsourcing*** is currently used to collect data related to subjective tasks, in order to devise a methodology to collect annotations for our task, and evaluate the quality of our dataset (RQ1).

2.1. Definitions of toxicity-related speeches in the psychology literature

Several studies show the necessity of detecting hate speech on the Internet [105]. With the increasing use of the Internet and websites where user comments are enabled, the quantity of user posted messages is too large for human moderators alone to filter all of them. Thus, it becomes more and more important to be able to detect hateful comments in an automatic way. However, each study do not tackle exactly the same problem, or at least do not use the same words to describe it. This is why we first proceed to define the different expressions found in the literature to qualify the different types of undesirable speeches found on the Internet. Afterwards, we give a description of the variables which influence these speeches perception. Finally, we examine how the studies were realized to point out these variables because it might give insights to create a dataset for our experiments.

2.1.1. Term definition

Looking at the psychology and Computer Science literatures, we noticed that several words or expressions are considered equivalent or are used without a precise definition, what results in a blurry distinction between them. These words are "offensiveness", "aggression", "toxicity", "hateful" or "harmful" speeches and more rarely "flaming". To refer to all of them, we name them undesirable speeches in the following sections.

From psychology, Archard [6] explains that ***offensive speech*** "offends the other in as much as it is directed at some property of the other (a personal characteristic, belief, relationship, membership of a group, etc.) and causes offence by the manner in which the other is represented. Some offensive speech can be hateful but need not be." He investigates the wrongfulness of such speeches. The main key point he highlights is the wrong aim of these speeches which "attempt to denigrate, humiliate, diminish, dishonor, or disrespect the other". He also makes a clear distinction between ***harmful and hurtful speeches***, and consequently offensiveness: "Harmful acts need not be hurtful. I can damage your interests without you being aware of the fact. Equally but conversely hurtful acts need not be harmful. I can occasion you intense but short-lived mental distress without setting back your interests. Offense is thus hurtful but need not be harmful". In a similar way, the difference between offensive speech and hate speech is explained in some Computer Science papers [36] by the fact that ***hate speeches*** use "language that is used to express hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group.", while offensive language is not always aiming at hurting people.

Finally, ***toxicity*** is differentiating from these previous concepts in the sense that it does not deal with the ideas expressed in the speech, but only with how these ideas are communicated [1]: the ideas could be hateful but expressed in a clearly argumented way without using abusive language, and so would not be toxic.

Flaming is a concept directly related to the Web since it is defined only for online language. It refers to "the use of offensive language such as swearing, insulting and providing hateful comments through an online medium" [71]. When studying YouTube comments, flaming was found related to "political attack and racial attack" with the use of "stereotypes, speculation, comparison, degrading comments, slander/defame, sedition, sarcasm, threaten, challenge, criticism, name-calling, and sexual harassments". Thus flaming seems to be a larger group of speeches containing both offensive and hateful speeches.

In the Computer Science literature, these different concepts are sometimes blurry, that is why in the following section we proceed by looking at the psychology literature related to each of these concepts. Toxicity and flaming perception study were not found, whereas studies related to offensiveness, harmful and hate speech perception exist. Thus, we focus on these speeches.

2.1.2. Methodology to search for the papers

To research the psychology literature concerning the perception of toxicity-related speeches, we performed a search on Google Scholar using the combination of the following keywords: "psychology", "speech", and ("offensiveness" or "hatefulness" or "hateful" or "harmful" or "harmfulness" or "aggressivity" or "toxicity" or "cyberbullying"). We specifically selected the papers focusing on the study of the variables influencing the perception of these speeches, and on the different characteristics of the speeches (mainly the different targets). Then, we added the two keywords "perception" and "variable" which enabled to find more specific papers. Finally, we searched whether there is existing literature which deals with these speeches on the Web by adding "Web" and "Internet" to the initial query, but no results were found.

2.1.3. The different targets of hate speech

Several papers identify different classes of hate speech depending on the target of the speech.

In an analysis of hate speeches on the Internet [95], several different "hate" topics are identified: race, religion, disability, sexual orientation, ethnicity, or gender, as well as behavioural and physical aspects that are not necessarily crimes. Hate categories are defined: Race, Behaviour, Physical, Sexual orientation, Class, Gender, Ethnicity, Disability, and Religion, (and other). In the paper analyzing hate speech on Instagram in Indonesia [78], 6 classes are identified: race/ethnics, religion, ability, social status, moral, and appearance look. Henry et al. [57] distinguish between 5 target groups: "groups representing different races (Black/African-American, Latino(a), White/European-American, Asian/Asian-American, Arab/Arab-American), genders (male, female), sexual orientations (gay people, straight people), mental status (highly intelligent people, mentally ill people, mentally disabled people), religious affiliation (Jewish people), age (elderly people), and physical status (obese people)".

2.1.4. The different variables influencing how toxicity-related speeches are judged

Theoretical papers, not specific to hate speech on the Internet, study what are the variables that influence how hate and offensive speeches are received by different people. A summary of these different factors is written in Table 2.1. The different groups of variables which influence sentence offensiveness perception are summarized in Fig. 2.1.

[1] https://www.forbes.com/sites/kalevleetaru/2017/02/23/fighting-words-not-ideas-googles-new-ai-powered-toxic-speech-filter-is-the-right-approach

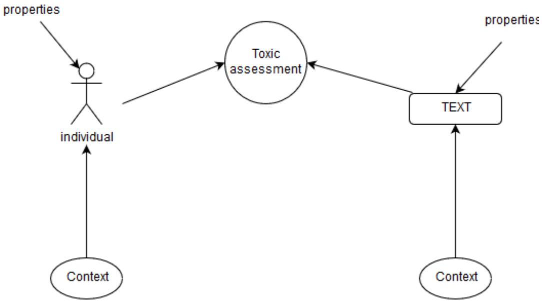


Figure 2.1: Summary of the variables influencing the assessment of sentence toxicity.

People's internal characteristics

Guberman et al. [54] underline the "variation in individual perceptions of malicious content". In their study attempting to rate *aggressiveness* of tweets, they show a difference depending on *gender*: women rate tweets more often as aggressive than men. They additionally mention other possible factors influencing how aggressive a sentence is perceived. They mention the tendency that some people have "to interpret ambiguous stimuli as being intentionally aggressive" (named *"individuals' attributions of intent"*), as well as the dispositions of people to become angry and anxious (named *individuals' angry and anxious dispositions*) because these dispositions make people more prone to judge sentences as aggressive. Downs et al. [40] find that two main factors influence how harmful a hate speech is perceived: *gender and liberalism inclination.*

Cowan and al. [32, 33] made a detailed study on the different variables influencing the perceived offensiveness of hate speech. Their results are correlated with the other papers. They show that "the nature of the observer - *his or her ethnicity, gender, education, and age* - plays a significant role". In [34], they make a distinction between the perceived offensiveness of a speech and the perceived harmfulness. For example, they show that ethnicity is a main factor in the perceived harmfulness but not offensiveness.

Several works specifically study racial hate speech. O'Dean et al. [79] find two different factors influencing how hate speeches are perceived: *the frequency to which people are subject to racial prejudice*, and *people's "beliefs about the appropriateness of expressing racial prejudice"*. Indeed, they show that racial hate speech is judged more offensive by people less often subject to racial prejudice and people who strongly believe that expressing racial prejudice is inappropriate. Williams et al. [109] find similar variables, and they additionally highlight the difference of perception resulting from different *ethnicities*. The participants of their experiments which evaluate perceived offensiveness of Internet memes are "People of Color" and "White" people. They observe that people in this first category, when they were more often subject of racial "microaggressions", perceive the memes as more offensive, whereas this phenomenon was not observed for the White participants. Still focusing on racial hate speeches, Boeckmann et al. [14] show that they trigger different kinds of emotional responses (fear, anger, sadness, outrage). Furthermore, participants with high membership esteem reacted more strongly to threats to their group than low identifiers.

Other variables: speech context (speech target and author, ...) and speech characteristics Several studies [32, 35, 57] are interested in understanding what are the characteristics of a sentence which influence how offensive it is perceived. They look at which categories of hate, which targets are expressed in the sentence and at the syntactic and semantic properties of the sentences (length of a sentence, usage of profanity or not, ...). Cowan et al. [32, 33] find that properties of the speech itself and its context influence how it is perceived. Mainly, *the targeted group of the speech, whether it is a public or private speech, and whether it received a response* are the factors influencing how the speech is perceived. Another study finds that the perceived offensiveness is also influenced by how the direct target of the speech behaved and felt [34]. Boeckmann et al. [14] also underline a difference between the *targets* of the speeches: a speech directed toward a person is seen as more offensive than a speech toward the group of people a person belongs to. Moreover, other studies investigate whether the author of the sentence is a variable which influences how the receiver of the sentence or a third person evaluate the offensiveness of the sentence [35].

Besides, Sood et al. [98] study the use of profane words on different Yahoo Web communities (political or not in the case of the paper). They underline that different communities not only use profanity with different frequencies, but also in different ways or contexts and judge the words differently. For them, the perception depends on the *community and on the domain of the community.*

Variable category	Variable	Measure	Paper
Internal characteristic	gender	question	[54], [40], [32, 33]
Internal characteristic	ethnicity	question	[32, 33], [109]
Internal characteristic	education (level of educational attainment)	question	[32, 33]
Internal characteristic	age	question	[32, 33]
Internal characteristic	liberalism inclination	question 7-point scale	[40]
Internal characteristic	"individuals' attributions of intent", individuals' angry and anxious dispositions	not investigated	[54]
Internal characteristic	frequency to which people are subject to racial prejudice, people's "beliefs about the appropriateness of expressing racial prejudice"	question with scale	[79], [109]
Internal characteristic	membership esteem to the offended group	question with several scales	[14]
Sent. charact. / context	targeted group or person	invented scenario	[32, 33], [14], [57]
Sent. charact.	category of hate speech	info in dataset	[57]
Sent. charact.	subtle or blatant prejudice, properties of the sentences	in the dataset	[37], [32]
Sent. context	public or private speech	invented scenario	[32, 33]
Sent. context	speech received a response or not, what type of response	invented scenario	[32, 33], [34]
Sent. context	author of the speech	invented scenario	[35]
Sent. context	Internet community in which the speech is published	info in dataset	[98]

Table 2.1: The factors influencing hate speech perception and/or offensive speech perception, ordered in 3 categories: internal belief (people's individual characteristics), sentence characteristics (how the sentence in itself is constructed), and sentence context (the surroundings of the sentence).

2.1.5. Methodologies of the different studies

Computer Science study

Guberman et al. [54] investigate perceived violence of tweets, and give recommendations on the process of tweet annotation. They had the tweets rated according to an adapted version of the Buss-Perry Aggression Questionnaire (BPAQ). This questionnaire consists of several propositions that annotators rate according to whether these are characteristics of the tweets or not. The workers on Mechanical Turk had to go over 14 gold questions, and 12 correct answers were required to go to the real task. The authors found 30% disagreement among the six workers, and explain it with different factors: the questionnaire might not be representative of tweet violence, no context around the tweets is specified, it is hard to judge whether tweets are published for promotion and whether they are written or targeting individuals or organizations, and the perception of violence depends on people's own beliefs which are different among annotators.

Psychology study

In the theoretical studies, the participants are presented with scenarios and asked to rate them according to some propositions, the ratings being averaged in the end.

Cowan et al. [32] ask their participants to rate the scenarios according to the coherence with the following proposition "the message is offensive" between 1 ("strongly disagree") to 7 ("strongly agree"). In the end of the experiment, they ask the participants to give their background information. Similarly, in [34], they present scenarios constituted of sentences and possible responses, and they ask the participants to rate the propositions "how offensive is the message?" between 1 ("not at all offensive) and 12 ("extremely offensive"), "how serious is the offense" between 1 ("not at all serious") and 12 ("extremely serious"), "how harmed was the receiver of the sentence" from 1 ("not at all harmed") to 12 ("extremely harmed"). Since the two first questions had highly correlated responses, they were aggregated together.

O'Dean et al. [79] defined 10 propositions to rate between 1 ("disagree very strongly") and 9 ("agree very strongly") and averaged the scores in order to evaluate the perceived offensiveness of the scenarios. In a same way, Boeckmann et al. [14] measure how harmful a scenario is by asking participants to rate 6 propositions on a 1 to 6 scale and averaged the ratings. To measure offensiveness, Williams et al. [109] ask the participants to rate images along "how comfortable (reverse scored), acceptable (reverse scored), offensive, hurtful, and annoying they were on a 7-point Likert scale ranging from 1 (Strongly Disagree) to 7 (Strongly Agree)", and averaged the scores for each image.

Cunningham et al. [35] employ a different method. They propose 4 scenarios to participants and ask them to select which one is the most offensive. The scenarios consist in asking the participants to imagine being in a specific situation, such as assisting a men's basketball game and select one out of four possible situations such as "A Caucasian, female said: "Of course we lost. We played like a bunch of girls.""

2.1.6. Discussion

RQ 1. In this section, we looked at the psychology literature related to the different kind of toxicity-related speeches. Considering that their definitions are blurry, we have to identify one clear type of speech to study in case we need to build a dataset of toxic-related speech. The perception is influenced by three kinds of variables: sentence characteristics, sentence context, and individual internal characteristics.

If sentence context is made clear and sentence characteristics is intrinsic to the data sample, the only variables influencing sample perception are individuals' internal characteristics. Depending on the studies, different internal characteristics variables are presented to have influence on the offensiveness judgement. Cowan et al. [32–34] highlight gender, age, ethnicity and education as the main influencing factors. We will first focus on these factors which are the easiest to measure. In other studies, factors on the "psychological" side are claimed important: liberalism sense, sense of belonging to a community, the frequency of being subject to prejudice, the belief of appropriateness to express racial prejudice, angry and anxiousness dispositions, attribution of intent. These factors which are more complicated to measure will be left out, but could be investigated in future work.

RQ1. To collect the data (perceptions of sentence toxicity), we would have to take into account how the psychological studies are conducted. For each sample sentence, we should consider only one possible context -which might have to be explained to the annotators- so that this factor does not influence the annotations they give. If we could run several crowdsourcing experiments, we could compare the crowdsourcing quality of experiments with more or less indications, for example where no context information is given.

RQ3. The perception of the toxicity-related speeches depends on many different variables, therefore it seems a good approach to simplify as much as possible the prediction task (depending on the available datasets) by restricting the properties of the speeches we investigate. Mainly, we can select one specific category of speech (for example, racist or sexist speeches). Moreover, the psychology literature identifies different target groups of hateful speeches. We can possibly use them to create different datasets or as additional information to help the Machine Learning classifiers learn which sentences are toxic in which toxic speech category. We could additionally consider the sentence context fixed in the experiments. A dataset containing sentences with similar characteristics could help the learning process since there would be fewer variations to learn.

2.2. Computational methods to detect toxic speech

A 2015 literature survey [75] lists scientific papers dealing with hatefulness detection on the Web, explaining that they search for the papers with the keywords "online hate speech", "offensive language online", cyber-bullying", "hateful language" because hate speech is not precisely defined in Computer Science. It investigates the preprocessing techniques, the extracted features, the feature selection techniques and the classification algorithms employed. However, it does not address the question of constituting a dataset and its evaluation to train such algorithms, nor does it looks at the recently developed Deep Learning methods. The most recent survey about the topic [94] dating from 2017 mentions these Deep Learning algorithms. It mentions the amount of data the different papers use, but it does not specifically address how the data are annotated - probably because this topic is not developed extensively in the papers. That is why this section focuses on the existing algorithms but also on their evaluation and training datasets.

2.2.1. Methodology to search for the papers

To search for the Computer Science papers dealing with undesirable speeches detection, we used the different queries "Machine Learning + [name of an undesirable speech] (+ "speech")", "Deep Learning + [name of an undesirable speech] (+ "speech")", and "automatic prediction + [name of an undesirable speech]". We selected all the papers which deal with datasets and with algorithms for undesirable speech prediction.

The MANDOLA project[2] did not appear in the research results but it is worth mentioning because its aim is to monitor and report hate speech on the Internet. Sadly, few detailed publications are available about the project, but a general overview is given. It focuses on finding a definition of hate speech, collecting a Web dataset annotated by experts to classify different hate speech categories, developing (Naive Bayes) classifiers to automatically predict these classes [3], and finally visualizing the hate speech distribution on a map. Ethical and legal reflections on the use and control of hate speech on the Web are also available.

2.2.2. Techniques used for toxicity detection

"Traditional" Machine Learning techniques

Until 2016, most of the techniques used for detection of hatefulness or insults were Machine Learning algorithms, without Deep Learning. They performed feature extraction and selection, and then used classification algorithms [75]. These methods are listed in Table 2.2, and compared in Table 2.3 and details on the datasets employed are listed in Table 2.7.

The most used algorithm is Support Vector Machine. Sood et al. [99] use list of profane words and Support Vector Machines, Chandasekharan et al [20] aggregate data from several Internet communities and classify sentences from another community with Support Vector Machine. Chen et al. [25] classify into abusive, not abusive and undecided (to take into account subjectivity of the judgments) and rate from 1 to 4 the harmfulness, by using Support Vector Machines. They do not get good results for the ratings. Davidson et al. [36] classify between hate speech, offensive but not hate, and neither of the two, while Burnap et al. [19] simply classify between hateful or not.
Warner et al. [107] define several categories of hate speech ("anti-semitic, anti-black, anti-Asian, anti-woman, anti-muslim, anti-immigrant or other-hate") and suggest to use distinct classifiers for each of them. Indeed, they explain that different categories of hate speech make use of different stereotypes with distinct lexical fields, and therefore it should be easier for each classifier to learn one unique type of speech. Sood et al. [100] in another paper explore insult detection using Support Vector Machines, they test different kinds of features. Their experiments suggest that classifying insults in a general domain or training separate SVM for different categories of comments (politics, news, entertainment, business, world) might not lead to much difference in performance, but they also suggest that it might depend on which categories, certain employing more specific language than others. Dinakar et al. [38] aim at detecting cyberbullying by classifying sentences into three different topics (sexuality, race and culture and intelligence), they test different classifiers trained on separate topic datasets and on a common dataset. They achieve better performances with topic-specific classifiers.

Certain works only focus on using lists of profane words, what Rojas et al [89] augment with genomics inspired techniques. Other works use language trees for detecting and filtering offensive comments [113], logistic regression [108] - classification based on text features and additional commenter features (gender, age, location)-, [39] - classification of agressiveness based on word embedding-, regression [77] with the Vowpal

[2] `http://mandola-project.eu/`

[3] `http://mandola-project.eu/m/filer_public/06/b9/06b92efd-cce2-4204-a2a9-5eb2c34912f7/mandola-d31.pdf`

Wabbit's regression model[4]. Chen et al [24] test both the Naive Bayes classifier and Support Vector Machines. Finally, Chen et al. [27] directly compute an offensiveness score from features extracted from sentences and authors' comment history. Chatzakou et al. [21] are interested in classifying Twitter users into bully, aggressive or normal users, with tree-based classifiers.

Wulczyn et al. [111] use a Multi-Layer Perceptron (MLP) and Logistic Regression. They published the only paper which addresses the problem of subjective judgments. They test two different models. A first one where the annotations are aggregated in a single one-hot encoded label, and a second one where the labels are represented as empirical distributions. They make the assumption that comments with high annotator agreement are different than the ones with lower agreement rate, and therefore empirical distributions should better represent the labels and help the algorithms learn. They get better performances with the Empirical Density model than the single label model.

Papers	Algorithms	Features	Task
[99]	Combination of SVM and lists	bigrams and stems	profanity or not
[20]	Bag of communities with NB, SVM, LR	Bag of Words, n-grams(1,2,3), feature selection with ANOVA F-values	abusive or not
[25]	SVM	n-grams, syntactic and semantic features, context (reply to a previous comment or new comment, news category, Twitter or Facebook account, number of comments for the article)	abusive, not abusive, undecided, severity between 1 and 4
[36]	LR, NB, decision trees, random forests, SVMs	n-gram (1,2,3), Penn Part-of-Speech tag, tweet quality, sentiment, syntactic, semantic, feature selection with logistic regression	offensive, hate but not offensive, neither
[19]	Bayesian LR, random forest decision tree, SVM, ensemble classifier	BoW n-grams(1,2,3), typed dependencies, feature selection by LR	hate speech or not
[107]	SVM	template-based strategy to generate features	anti-semitic or not
[100]	SVM, multi step classification	BoW n-grams (1,2), stems	insult or not, in or not categories, target (third party or previous comment), malicious intent or not
[38]	JRip rule-learner, decision tree, SVM	TF-IDF weighted unigrams, the Ortony lexicon of words denoting negative connation, a list of profane words and frequently occurring POS bigram tags	labels (sexuality, race, intelligence): multi-class, one/rest
[89]	distance matrix	characters of the sentence	presence or not of obscenity
[113]	relation tree construction	Part-of-Speech tags and typed dependency relation	identification of offensive sentence sections
[108]	LR	character n-grams, outside information about tweets and author (gender)	hate speech or not (sexist, racist, neither)
[39]	LR	Paragraph2vec embedding of comments and words	hateful / clean comment
[77]	Vowpal Wabbit's regression model	character N-grams, token unigrams and bigram, Linguistic, Syntactic features, embeddings	clean/abusive comments
[24]	NB, SVM	BoW and n-grams with term frequency values, text structural features - dim. reduction by document frequency reduction, chi-square, SVD	harassment or not, cyberbullying or not
[111]	LR, Multi-Layer Perceptron	BoW with word or character level n-grams	personal attack or not

Table 2.2: List of the Machine Learning methods for undesirable speech classification. LR (logistic regression), SVM (Support Vector Machine), NB (Naive Bayes)

Deep Learning for toxicity detection

The most recent researches use Deep Learning for classification of the sentences, we list them in Table 2.4 with their corresponding performance metrics in Table 2.5, and the associated datasets in Table 2.8. They usually train a Convolutional Neural Network (CNN), a Recurrent Neural Network (RNN), or its other variant Long-Short Term Memory neural network (LSTM), or combine several networks. All the following papers working on the classification of different types of undesirable speeches using deep neural networks claim to

[4] http://hunch.net/ vw/

Algo. cat.	Algorithm details	Performances	Comparison
SVM	SVM with lists [99]	P(0.90), R(0.20), maximal F1(0.32) and maximal A (0.92)	List-based methods: P(0.64), R(0.20), F1(0.30), A(0.91)
	SVM [25]	R: 0.64 for abuse classification, 0.2 for severity classification	list-based: 0.57
	SVM [107]	A(0.94), P(0.68), R(0.60), F1(0.63)	
	SVM [24]	average class A (= average R)(0.8), positive R(0.61)	NB, baseline without data balancing and dim. reduction: 0.67, 0.34
	SVM [38]	A(0.667) and kappa statistic (0.653) for SVM	JRip rule-learner, decision tree: lower performances
	SVM and LR [36],	P(0.91), R(0.90), F1(0.90), confusion matrix	NB, decision trees, random forests: lower performances
	SVM[19]	P(0.89), R(0.69), F1(0.77)	
	multi step classification [100]	P-R breakevenpoint(0.5009), maximum F1(0.5038), maximum A(0.9082)	SVM: 0.3781, 0.3953, 0.8980
LR	LR [108]	F1(0.7393), P(0.7293), R(0.7774)	
	LR [39]	AUC(0.8007)	Meth. using BoW with tf: 0.7889 or tfidf: 0.6933
	LR [111]	AUC(96.24), Spearman rank correlation(66.68)	
	Vowpal Wabbit's regression model [77]	P(0.773), R(0.794), F1(0.783), AUC(0.9055)	0.8007 AUC with LR [39]
	Bayesian LR [19]	P(0.89), R(0.69), F1(0.77)	
BoC	Bag of communities with NB, SVM, LR [20]	P, R, A(0.9118)	list-based: A(0.55), SVM: A(0.51)
ensemble	ensemble classifier [19]	P(0.89), R(0.69), F1(0.77)	
dist. mat.	distance matrix [89]	P(0.80), R(0.93), hit rate(0.86)	Levenshtein edit distance 0.67% hit rate
tree	random forest decision tree, ensemble classifier [19]	P(0.89), R(0.66), F1(0.77)	
	relation tree construction [113]	% of exact, excessive and insufficient filtering, A(0.9094), speed	
NN	Multi-Layer Perceptron [111]	AUC(96.59), Spearman rank correlation(68.17)	

Table 2.3: Comparison of the performances of the Machine Learning methods for undesirable speech classification. LR (logistic regression), SVM (Support Vector Machine), NB (Naive Bayes). The evaluation metrics are the accuracy (A), precision (P), recall (R), the F1-score (F1), the Area Under the Curve (AUC)

achieve better performances than traditional Machine Learning techniques.

Gao et al. [51] perform two-class classification by bootstrapping a Slur term Learner and an LSTM Deep Learning algorithm. In another paper [50], they compare Logistic Regression, LSTM, and an ensemble model. They introduce context (username and title of the commented news article), suggesting that the perceived offensiveness depends on the context of the sentence. This way of integrating context could be adapted to integrate the annotator belief profile to refine the prediction. Sax [92] also compared LSTM to traditional Machine Learning algorithms (mainly Logistic Regression), using only 4921 training samples (Kaggle dataset of insulting/non-insulting sentences). He finds that LSTM gets higher F1-score but slightly lower AUC score for a small architecture (adapted to the dataset).
Badjatiya et al. [9] classify between racist, sexist or neutral sentences. They test three different types of networks (CNN, LSTM, FastText) and achieve better accuracies than previous Machine Learning techniques. Similarly, Chu et al. [29] test three different types of neural networks (RNN, CNN with character embedding, CNN with word embedding) to classify whether a Wikipedia comment is an attack or not. They achieve higher performances than linear regression and multi-layer perceptron (MLP).
Gamback et al. [49] use CNN to classify sentences between racist, sexist, both or neither. Similarly, Pavlopoulos et al. [81] use RNN and compare with CNN.
Finally, Zhang et al. [119] combine both CNN and Gated Recurrent Unit (GRU) recurrent neural network. They evaluate their architecture on seven different datasets, comparing the performances with baseline models (SVM and CNN). They find out that their model obtains a F1-score higher than the other algorithms on six of the seven datasets.

The results of all these studies show that it is feasible to use Deep Learning to classify hate speeches, and they suggest that it might be more accurate than using "traditional" Machine Learning algorithms.

Papers	Algorithms	Features	Task
[51]	Co-learning of Slur term learner and LSTM	Word2vec embeddings of the sentences	hateful or not
[50]	bi-LSTM with attention mechanism and with context, ensemble model (LSTM, LR)	word-level and character-level n-gram features, lexicon derived features, emotion lexicon feature	hateful or not
[9]	CNN, LSTM, FastText as networks, and as word embeddings for classifiers	word embeddings with either random embeddings or GloVe embeddings	racist, sexist, neither
[29]	LSTM, CNN	GLoVe word embedding, character-level embedding	attack or not
[49]	CNN	word embeddings with Word2vec and random vectors, n-grams	racism, sexism, both, neither
[81]	GRU RNN with and without attention mechanism, CNN	Word2vec and GLOVE word embedding	reject or accept a user comment
[119]	CNN+GRU	Word2Vec	racism, sexism, neutral / hate,non-hate

Table 2.4: List of Deep Learning methods for undesirable speech classification. LR (logistic regression), SVM (Support Vector Machine), NB (Naive Bayes)

Cat.	Algorithms	Performances	Comparison
LSTM	Co-learning of Slur term learner and LSTM [51]	P(0.422), R(0.580), F1(0.489)	LR(P:0.088, R:0.328, F1:0.139), LSTM (P:0.791, R:0.132, F1:0.228)
	bi-LSTM with attention mechanism and with context [50]	A(0.766), P(0.614), R(0.499), F1(0.548), AUC(0.760)	LR(A:0.750, P:0.572, R:0.516, F1:0.542, AUC:0.778)
	ensemble model (LSTM, LR) [50]	A(0.779), P(0.650), R(0.496), F1(0.560), AUC(0.804)	see above
	LSTM+Random Embedding+GBDT [9]	P(0.930), R(0.930), F1(0.930)	char n-grams, BoW, tf-idf with LR, Random Forest, SVM, Gradient Boosted Decision Trees (P:0.816, R:0.816, F1:0.816)
	LSTM [29]	F1(0.70), A(0.94)	LR, feed-forward neural network: (F1:0.54, A:0.91)
	GRU RNN with and without attention mechanism [81]	AUC(80.41) with majority labels, Spearman correlation with human probabilistic gold labels (52.51)	word-list(AUC:64.19, Sp:24.33); [111] method(AUC:75.67, Sp:43.80)
CNN	word2vec+CNN [49]	P(0.8566), R(0.7214), F1(0.7829)	LR with character n-grams(P:0.7287, R:0.7775, F1:0.7389)
	CNN+GloVe [9]	P(0.839), R(0.840), F1(0.839)	see above
	CNN with character embeddings [29]	F1(0.73), A(0.94)	see above
	CNN [81]	AUC with majority labels(76.03), Spearman correlation with human probabilistic gold labels (42.88)	see above
Combination	FastText+Random Embedding+GBDT [9]	P(0.886), R(0.887), F1(0.886)	see above
	CNN + GRU [119]	F1(0.94)	SVM(0.89), CNN(0.90)

Table 2.5: Comparison of Deep Learning methods for undesirable speech classification. LR (logistic regression), SVM (Support Vector Machine), NB (Naive Bayes)

Evaluation of the algorithms

To evaluate and compare the algorithms, the labels collected by crowdsourcing are considered as ground truth. The datasets are separated into a training and a test set. The algorithms are trained on this first subset and tested on the second one. Usually, the papers measure accuracy, precision and recall of the prediction results (see Tables 2.3, 2.5).

Limitations of current techniques

We saw in the previous section 2.1 that three types of variables influence how hate speech is perceived (sentence characteristics, context, and people's background). In all these studies, a unique label per sentence is taken into account, which is predicted based on the extracted sentence characteristics. Additionally to the sentence features, research is done to judge whether a sentence is offensive based on who wrote the sentence and in which context. We can consider here that the two first variables of the theoretical studies are taken into account. However, no research is done to predict how people perceive the sentence depending on their background, beliefs, using people additional context besides the sentence context.

This is a simplification which would merit to be addressed. In this recent survey [94] on hate speech detection listing the employed methods, not taking into account people's beliefs is highlighted as one main limitation of current research: "Unlike other tasks in NLP, hate speech may have strong cultural implications, that is, depending on one's particular cultural background, an utterance may be perceived as offensive or not". Therefore, addressing the subjectiveness of perceived offensiveness of sentences on the Web is new and a useful task.

Additionally, Montoyo et al. [76], in their literature survey on text sentiment analysis and subjectivity, highlight an important limitation of current studies: they mention that sentiment and subjectivity depend on social and cultural aspects, but neither the writer nor the reader interpretations and personal backgrounds are considered in current studies. They also mention that depending on the news source "(i.e. in terms of bias, reputation, trust)", the sentiment perception of the reader might be different, within a same and different countries and cultures.

2.2.3. Dataset gathering methods

The dataset specifications for the previously cited papers are listed in Tables 2.7, 2.8.

Available datasets

The following Table 2.6 lists datasets which were made public by their authors. Most of the available datasets are too small to train Deep Learning algorithms. However, a few of them should be large enough, and could also be aggregated together. We could also use resampling methods such as SMOTE [23] to artificially upsample the data and obtain larger and more balanced datasets.

Data collection

Most of the datasets are constituted of data scraped on websites (Youtube video comments, Twitter posts and their comments, Wikipedia article comments, news article comments, question-answering forums). Then, these data are annotated using crowdsourcing.

Two different methods are used. In some papers, all comments/QA are collected. In other papers, a filter is used to only collect data which contain specific words (list of words usually found in hate speeches), and other random data are collected to constitute the "false" class of the dataset. The data are then annotated to precisely divide them into two classes. The problem which can be encountered here is a problem of unbalanced dataset: usually, more negative classes than positive classes are collected. This issue of the unbalanced dataset will have to be addressed carefully in our experiments. Certain papers perform data augmentation simply by duplicating some samples of the positive class to solve the issue.

Data annotation

The data are annotated by a few selected people (students of professors at university) or by crowdsourcing on crowdsourcing platforms, mainly Mechanical Turk or Crowdflower. For example, Sood et al. [100] collect comments from *Yahoo! Buzz* and have them annotated using MechanicalTurk. In all the papers, non-expert workers are asked to annotate the samples.

To label the data, majority voting is usually used to resolve the disparity between annotators opinion. For example, Reynolds et al. [87] ask three annotators to label the data and classify a label as positive if at least two of the three annotations are positive. In some cases, a consensus threshold is decided like in [100], and

Provenance	Source	Dataset size	Labels
[36]	Twitter	24802 tweets (and the number of annotations for each tweet and category - 3 or more)	hate, offensive but not hate, none of the two
Kaggle competition[a]		3947 sentences	neutral, insulting
Yahoo [77]	Yahoo finance, Yahoo news	759402 finance, 1390774 news	neutral, abusive (further classified into hate speech, profanity, derogatory language)
[108]	Twitter	16914	racist, sexist, neutral
[112], [87], [73]		684 sentences from the Web, 13652 posts (question answering), 626 tweets	cyberbullying or not
[50]	FoxNews comments	1528	hateful, not hateful
[81]	Greek news website	1.6M	aggressive, not aggressive
Kaggle competition 2[b]	Wikipedia comments	95851 sentences, 95851 labels	binary labels: toxic, severe toxic, obscene, threat, insult, identity hate
[111][c] same data as Kaggle competit. 2	Wikipedia comments [d]	between 100k and 160k samples, 1598289 judgments ≈ 10 per sample [e]	not/personal attack; not/toxic with score $[\![-2;2]\!]$. Binary labels with each annotator and personal background (gender, first language, age group, education)
[119]	Twitter	2435 tweets (414 hate, 2021 non-hate)	hate/non-hate for tweets about Muslims and refugees

[a] https://www.kaggle.com/c/detecting-insults-in-social-commentary/data

[b] https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data

[c] https://meta.wikimedia.org/wiki/Research:Detox/Data_Release

[d] https://conversationai.github.io/

[e] https://figshare.com/articles/Wikipedia_Talk_Labels_Toxicity/4563973

Table 2.6: Summary of available datasets

only the labels which reach a higher value than the threshold are kept in the dataset. They had minimum three annotators per sample, and only kept labels (6009) when there was a consensus higher than 60%.

Chen et al. [25] study the agreement of judgments, they highlight the low agreement even for only the 3 categories (abusive, not abusive, undecided). That supports the idea that the labels should be "parametrized" by people "beliefs". In [77], a short comparison of labels given by trained annotators and by crowdsourcing suggests that annotations by crowd workers have a lower quality. Hence the use of user modeling (expertise model) to choose which worker to annotate which sample.

Questions asked for the annotation

Usually, simply a yes/no question is ask. Reynolds et al. [87] ask "1) Does this post contain cyberbullying (Yes or No)?". They additionally asked "2) On a scale of 1 (mild) to 10 (severe) how bad is the cyberbullying in this post (enter 0 for no cyberbullying)? 3) What words or phrases in the post(s) are indicative of the cyberbullying (enter n/a for no cyberbullying)? 4) Please enter any additional information you would like to share about this post.", to collect data for future research but have not exploited them yet. Similarly, Burnap et al. [19] ask the question "is this text offensive or antagonistic in terms of race, ethnicity or religion?" yes, no, undecided.

Sood et al. [99] work on profanity detection and after a pilot study, they define their crowdsourcing question as follows: "Does this message contain any content you would describe as 'profanity?' (including profanity that is disguised or modified such as @ss, s***, and biatch) (Yes/No)". They conducted a pilot study [100] to identify how to formulate the questions for annotation. They say that asking to rate comments on a valence scale is too difficult and vague for the annotators. They annotate the use of profanity, insults, and the target of the insult. Gamback et al. [49] also ran several crowdsourcing tests to set up their crowdsourcing questions, and chose the question with the highest inter-annotator agreement. The final question is "Does the comment contain a personal attack or harassment? Targeted at the recipient of the message (i.e. you suck). Targeted at a third party (i.e. Bob sucks). Being reported or quoted (i.e. Bob said Henri sucks). Another kind of attack or harassment. This is not an attack or harassment."

Chen et al. [25] ask "Is the comment abusive or not?" with answers yes/no/undecided. If a comment is said to be abusive, it is asked to rate it on a scale from 1 to 4 in which 1 is very slightly abusive and 4 is very strongly abusive. Disagreement between workers is explained by several facts: the workers make mistakes or spam, or the samples are ambiguous. After removing the low-quality workers, [25] highlights the subjectivity of the task by explaining that less than 40% of the samples got total agreement between the annotators. Interestingly, non-abusive comments have 41% of unanimous labels while abusive comments only have 27% of unanimous labels, what suggests that subjectivity is even more important in the abusive content.

Waseem et al. [108] give a precise definition of offensive comments following several criteria: "A tweet is offensive if it 1) uses a sexist or racial slur. 2) attacks a minority. 3) seeks to silence a minority. 4) criticizes a minority (without a well-founded argument). 5) promotes, but does not directly use, hate speech or violent crime. 6) criticizes a minority and uses a straw man argument. 7) blatantly misrepresents truth or seeks to distort views on a minority with unfounded claims. 8) shows support of problematic hashtags. E.g. “#BanIslam”, “#whoriental”, “#whitegenocide” 9) negatively stereotypes a minority. 10) defends xenophobia or sexism. 11) contains a screen name that is offensive, as per the previous criteria, the tweet is ambiguous (at best), and the tweet is on a topic that satisfies any of the above criteria." They say that potential disagreements come from people’s subjectivity and the lack of context surrounding the tweets making people not able to define whether the message is hateful or not. Gao et al. [51] also give a detailed description of offensiveness in their crowdsourcing task: "tweets that explicitly or implicitly propagate stereotypes targeting a specific group whether it is the initial expression or a meta-expression discussing the hate speech itself". Similarly, in [50] they define hate speech as "the language which explicitly or implicitly threatens or demeans a person or a group based upon a facet of their identity such as gender, ethnicity, or sexual orientation".

2.2.4. Discussion

In this section, we were interested in the current automatic prediction algorithms used for classification of toxicity-related speeches and the creation of datasets to fulfil this task.

RQ3. Related to the prediction of toxicity-related speeches, the studies use different datasets and do not always use the same performance metrics. Therefore, it is not easy to compare the algorithms performance. In Machine Learning, several algorithms are always used: Logistic Regression, Support Vector Machine and Multi-Layer Perceptron. For Deep Learning, it might be necessary to test using both CNN and RNN neural networks for the experiments.

RQ1. Considering the collection and annotation of the dataset, there are already datasets available, related to undesirable speeches. However, in all these datasets, there is no mention of the subjectivity of the task (sentence being undesired for one person but not for another), except in [111]. Thus, the datasets could be used as samples which would have to be annotated by the crowd to have an idea of how subjective the data are, the current labels could serve as "ground truth" labels to investigate how several workers opinion differs from the majority opinion.

Based on [25], two different tasks could be addressed. The first one is to classify sentences as undesirable, not undesirable and undecided based on people’s beliefs. The second task - which might be more complex since the disagreement between people is probably higher - is to predict the undesirability rating (from 1 to 4) that people would attribute to the sentences based on their beliefs.

All the studies identify labels by asking only one question per sample. However, the psychological studies used several questions and aggregated their results to determine whether a sentence is offensive or not. It might be useful to investigate whether the answer to one question and the aggregation of several judgements give similar results, or whether the algorithms learn better with one or the other method.

Papers	D. source	D. collect. meth.	D. annotat. crowd	D. annotat. meth.	D. augmentat., post proces.	Nb. D.
[99]	Social news site (Yahoo)	6500 randomly selected comments	Crowdsourcing (MT)	Gold questions. Minimum 3 ann./samp. Filter data with less than 66% consensus.		6354 comments, 9.4% positive
[20]	Posts of 9 Web communities	10M posts crawled	No crowd, except the moderators of websites	Assumptions on communities posting profanities or not, and data from moderators	Random selection for balanced dataset	10M posts, 50% positive
[25]	Comments on a news website	Crawled comments, pre-filtered based on lexicon lists, balance between 0 to 5 profane words per comment	Crowdflower: 6 annot./comment	Gold questions. If at least 3 votes for one category, label attributed, otherwise undecided	resampling (oversampling) for balanced dataset	2000 comments, 15% "abusive", 6% "undecided"
[36]	Twitter	tweets from users sometimes posting hate words, 25k randomly sampled	Crowdflower, min 3 annotators	majority voting		24802 tweets, 5% positive
[19]	Twitter	Collection of tweets containing hashtags related to "woolwich", 2000 randomly selected	Crowdflower, min 4 coder/sample	keep tweets with min. 75% agreement, remove undecided		1901 tweets, 11.68% positive
[107]	Yahoo news, American Jewish Congress identified offensive websites	paragraph containing words related to Judaism and Israel	3 annotators and review from the authors	Majority voting. Annotation between anti-semitic, anti-black, anti-asian, anti-woman, anti-muslim, anti-immigrant, other-hate.		1000 paragraphs
[100] same as [99]	Yahoo! Buzz social news site	Randomly collected, filtered by length	MTurk, min. 3 annotators	consensus, gold questions		6009 comments, 20.62% positive
[38]	YouTube comments	scraped from controversial videos grouped into clusters of physical appearance, sexuality, race	2 recruited annotators	select comments with an inter-rater agreement of kappa >= 0.4	annotated 1500 comments per cluster	1500 per cluster
[89]	Colombian news website comments, Portuguese sport news website	random selection	1 human expert annotator	1 annotation considered ground truth		300 Colombian comments, 2500 Portuguese sentences (521 positive)
[113]	11670 YouTube comments	first 40 comments of the 20 most discussed videos in 15 different categories	5 students			11670 comments, 1739 positive
[108]	Twitter	Filtering: search tweets containing terms referring to religious, sexual, gender, and ethnic minorities	author and review by outside annotator			16K tweets: 3383 sexist, 1972 racist
[39]	Yahoo finance user comments	random collection	"editorially labeled"			56280 hate speech, 895456 clean
[77]	Yahoo! Finance and News	random comments and comments reported as abusive	expert annotators, 3 experts for the evaluation set	majority voting		759402 finance (7.0% positive) 448436(3.4%), 1390774 news (16.4% positive) 726073 (10.7%)
[24]	8 datasets of user comments (Twitter, YouTube, MySpace, FormSpring, Kongregate, SlashDot)	datasets from previous papers annotated manually or by crowdsourcing			balance by randomly oversampling [22]	between 1340 and 13153 data with different pos/neg rates
[111]	Wikipedia comments	random sample and comments of blocked users	Crowdflower, at least 10 annotators/sample	gold questions, aggregation by majority voting or by empirical density		115737, 11.7% positive

Table 2.7: Dataset gathering in papers for undesirable speech detection using Machine Learning

Papers	D. source	D. collect. meth.	D. annotat. crowd	D. annotat. meth.	D. augmentat., post proces.	Nb. D.
[51]	Twitter	randomly collected tweets	2 annotators with prior discussions about the topic	automatic annotation with hate slur term list and online automatic annotation, 1000 tweets manually annotated for error estimation		10 million for training, 62 million for testing
[50]	Fox News user comments	manual selection of representative news threads	2 english speaking annotators with prior discussions	if disagreement: labeled as hate speech		1528 annotated comments (435 labeled as hateful)
[9]	Twitter dataset of [108]					
[29]	Wikipedia comments of [111]		10 annotators	majority voting		115,846 comments, 12% positive
[49]	Twitter dataset same as [108]		1 expert, 3 amateurs	majority voting with the expert having 2 votes, amateurs 1 vote		6655 tweets
[81]	Gazzetta Greek newspaper, Wikipedia same as [111]	Gazzetta comments manually moderated by the authors (accept or reject)	5 annotators			1.45M training comments
[119]	Twitter	filtering of tweets using keywords and manual additional filtering	one expert and one student	discussion to reach agreement		2435 tweets

Table 2.8: Dataset gathering in papers for undesirable speech detection using Deep Learning

2.3. Machine Learning and Deep Learning for subjective tasks

Few research is performed to take into account the subjectivity of samples in prediction tasks, whereas it is an important issue since subjective data can not always be described with one unique label. Instead of considering unique ground truth labels for subjective NLP tasks (what would not be representative of the reality), it is suggested to consider several labels as acceptable for each sentence [2]. This new approach to datasets creation for Machine Learning algorithms training is presented as an approach which merits attention [2]. We investigate the current research related to it.

2.3.1. Methodology to search for the papers

To search for algorithms related to subjective data prediction or to predictions adapted to different conditions, we used two kinds of searches. Using the keyword "subjective" with different algorithm names returns algorithms which classify speeches between subjective or not, or which classify the sentiment of speeches; but they do not investigate the different interpretations of one speech. Thus, we used more specific queries by adding "user model", "user modelling" to the initial queries, and also "chatbot" or "conversation" because the chatbot field has literature on response adaptation. We selected all the Deep Learning and Machine Learning papers which integrate some kind of modelling of one external condition into the training process.

2.3.2. Machine Learning and subjective tasks

Beigman et al. [10] consider that there are "easy" and "hard" to label instances. They suggest that Machine Learning classifiers should be trained on the easily annotated samples so that at least these samples are well predicted. Similarly, the subjectivity of the samples can be considered as the variance over the annotations of each sample, and only the algorithms are reported to have higher performances when trained on the less subjective data [110]. Moreover, it was shown that training algorithms with soft labels corresponding to the annotators agreement over the sample label has slightly higher performances than training with all the data, and filtering the samples with the lowest agreement rates enables to train algorithms with higher performances than when using soft-labels [64]. In all these set-ups however, the different opinions on the samples are "ignored".

Alonso et al. [3] relax this assumption by considering that, for each sample a ground truth label exists, but for the samples where annotators do not agree with it, the algorithms are "allowed" to make errors (if these errors are the same as human errors). When training the Machine Learning algorithm, they use a cost-sensitive loss function which takes into account these possible human errors.

Finally, Reidsma et al. [85, 86] differentiate between annotation tasks of manifest content (directly observable) and of "projective latent content" (the annotations depend on the annotators' mental conception of the samples and the possible categories). They explain that the latter tasks exhibit higher disagreement rate since the annotations rely on the subjective interpretation of the sample by each annotator, but do not try to resolve this disagreement. To keep the subjectivity present in Machine Learning algorithms, they define two different architectures and training procedures. 1) They propose to train algorithms solely on the data which have a high-agreement rate. 2) Or they train one classifier per annotator and use an additional voting classifier which returns an output only when the annotator-specific classifiers agree. Their aim is to investigate relations between the voting classifier decisions and the annotators disagreement. Their setup was tested for datasets consisting of only three annotators contrary to us who have around 3000 annotators. We differ from this study because we want to output predictions tuned to each person, and since we have a large number of annotators with differing numbers of annotations, it is not feasible to learn one classifier per annotator.

2.3.3. Deep Learning models integrating subjectivity into their predictions

We did not find literature on Deep Learning where each annotation in the dataset would be used instead of using labels resulting from annotation aggregation.

Deep Learning architectures integrating user modelling

There are some research interested in adapting the models outputs to each user specifically.

Liu et al. [72] investigate user modelling for response ranking in chatbots. This paper proposes a new method to build and integrate the user profile into neural networks. As they present, related works which perform user modelling usually take the conversation histories to infer implicit features about the users. However, we are interested in using explicit features that psychology literature mentions, thus we will need to replace these implicit feature models by explicit features representations. Liu et al. [72] make use of implicit features. First, they learn a post embedding and a user model embedding with a user modelling network. Then, they integrate this model to the ranking neural network as an input. They propose two different methods. 1) The first one is simply to concatenate as input the user model, the post and the response to rank. 2) The second method separately computes a post and a response representations conditioned on the user profile by inputting into two different fully connected layers the user profile and the post or response. Finally, these two new representations are combined by another fully connected layer. The weights of the network are learned by back-propagation using a cross-entropy loss function. In a similar way, Li et al. [70] learn a user embedding at the same time as they train their neural network architecture with conversation histories. However, they integrate the embedding differently: the embedding is added as an additional component of each cells in the LSTM decoder network, so that it is considered as a "parameter" of the network.

Tang et al. [104] integrate the user model in their neural network for review ratings by learning a user matrix for each user, and multiplying each sentence input by this matrix. It is learned by back-propagation with a cross-entropy loss function which includes the matrices as regularization term. The user matrices are composed of a low-rank approximation specific to each user, and a global diagonal common to each user, that enables to use this common part for previously unknown users. If we consider using this method, we could additionally explicitly add our user features in another way. We could compare having one user matrix per annotator or one user matrix per demographics category.

Deep Learning architectures with conditioned inputs

We also searched for research interested in conditioning the models outputs with different kinds of information. Indeed, it could be an inspiration to design algorithms which are conditioned on the annotator specificities.

In [102], a Deep Learning architecture is defined to memorize prior informations contained in sentences before replying questions. For that, the informational sentences are entered in the network as memory cells, while the input question is on one side transformed by these memory cells to form one output, and on the other side kept unchanged to form a second output. Then, these two outputs are added and passed through a softmax layer to output the answer to the question. Similarly, Joshi et al. [65] input a user profile in the memory network so that the network outputs are conditioned on this profile. The profile is passed to the network in a same way as if it would be a conversation history, and is defined by words such as the gender, age, favourite food.

Zhang et al. [117] are interested in both giving a personality to chatbots and also adapting the chatbot utterances to the person it is interacting with. For that, they define the personality as a set of sentences describing a person interests (for example, one persona in [117] is defined as: "RPGs are my favorite genre. I also went to school to work with technology. The woman who gave birth to me is a physician. I am not a social person. I enjoy working with my hands."). Then, they define new Deep Learning models for response generation taking into accounts the two sets of personalities (chatbot and interlocutor). The first model simply extends the input by concatenating the persona with the sentence input, and feed them to the LSTM neural network. The second one (called generative profile memory network) enters the persona into memory cells on which the output is conditioned.

Another method which is used to personalize Deep Learning outputs to each user is to choose some parameters of the network and make them user dependent. The networks are then trained by back-propagating the error on the specific user training data to the specific user parameters. This is done for example to train speech recognition neural network personalized for each speaker [103].

Finally, in the field of recommendation systems, some examples can be found of ways to integrate user models to Deep Learning models. Mainly, the users are represented as a set of features which is concatenated to the other inputs. For example, the features can be represented as normalized continuous values between -1 and 1 as in [31].

2.3.4. Discussion

RQ3. The literature where algorithms outputs are adapted to certain users is mainly related to chatbot personalization. We can investigate how to adapt these neural network architectures to our problem in order to integrate a user model in the learning process. Another possibility to make the outputs user-specific is to use the identifier of each user without additional user modelling in order to model the correlations between annotations and users without considering the properties of these annotators. A future extension of our work could be to use techniques for user modelling using previous conversations and integrate these to our offensiveness prediction network, in order to take into account the context of the sentences.

2.4. Definition and evaluation method of algorithmic fairness

Fairness of Machine Learning algorithms is an important property to take into account considering that they are now used for human-related tasks such as credit scoring in mortgage or consumer credit, racial profiling, ... [90] Usually the performance of algorithms are evaluated by computing a metric such as the error between the algorithms' outputs and the expected outputs on a test dataset. However these metrics are not informative about the fairness of the algorithms. That is what Chouldechova et al. [28] show on the task of predicting recidivism: they compare the performance of two algorithms using global metrics like the accuracy and the Area Under the Curve, and show that even if these values are very similar for the two, their fairness performance are very different. This is why it is necessary to create specific metrics to assess the fairness of the algorithms.

However, still few research, usually found in the FAT* conference [5] (Fairness, Accountability, and Transparency) and FAT/ML conference [6] (Fairness, Accountability, and Transparency in Machine Learning), are interested in defining and evaluating the fairness of Machine Learning algorithms' outputs [46], mainly in investigating whether they discriminate certain categories of population. In a majority of the papers, the definition of fairness is confused with the metrics to measure it, and sometimes with the solutions to make the algorithms fairer.

2.4.1. Methodology to search for the papers

To search definitions and evaluation metrics dealing with fairness of Machine Learning algorithms, we did a Google Scholar search and went through all the papers of the new FAT* and FAT/ML conferences, as well as the references of the retained papers. We selected the papers which mention evaluation metrics of learning algorithms, related to bias or fairness because bias is often related to fairness of the algorithms. For example a gender bias can be considered as unfair towards one gender.

2.4.2. Definitions of algorithmic fairness

Dictionary's definitions

According to the dictionary's definition, fairness is "the quality of treating people equally or in a way that is right or reasonable" [7], or is also defined as an "impartial and just treatment or behaviour without favouritism or discrimination" [8]. The definition of fairness is therefore not strict. Certain definitions are targeting equality between individuals, while others do not specify to which entities fairness is related to. In a same way, certain definitions deal with equality while others do not but mention impartiality of treatment as well as discriminatory attitudes.

Definitions from a Machine Learning perspective

Certain Machine Learning algorithms are used to classify people or use information related to people to perform classification tasks. People are represented by two types of features: ***protected features*** (features such as race, gender, religion on which people should not be discriminated) and ***non-protected features*** (other features used to describe people such as their age, the number of prior convictions, ...). These features might constitute the inputs or part of the inputs of the system which makes a prediction, or correlations between the actual inputs of the system and these features might exist.

Several essays [12, 106] investigate fairness and discrimination in machine learning from a philosophical point of view. Most papers agree on the definition of algorithms' fairness: ***a fair algorithm is an algorithm whose outputs do not discriminate between different classes of people***. The papers only differ in the details of the formulation of the definition. For example, Binns [12] sees ML algorithms' unfairness as the "differences in treatment between protected and non-protected groups". This means that a chosen metric to evaluate the performance of an algorithm should have equal values when evaluated for different categories of population, these categories being defined by differentiating between people whose protected features' values are same or different. Similarly, Kamishima et al. [66] define fair predictions as "unbiased and non-discriminatory in relation to sensitive features such as gender, religion, race, ethnicity, handicaps, political convictions, and so on" (*sensitive features* is another way to call the *protected features*).

[5] `https://fatconference.org/`

[6] `https://www.fatml.org/`

[7] `https://dictionary.cambridge.org/dictionary/english/fairness`

[8] `https://en.oxforddictionaries.com/definition/fairness`

Zliobaité [120] and Agarwal [1] present a similar but more precise definition of ML algorithms' fairness, still assimilating fairness to non-discrimination. They consider an algorithm fair when it verifies the ***demographics-parity*** definition: the predictions of the algorithm are independent of any protected attribute. Precisely, for Zliobaité [120], "(1) people that are similar in terms of non-protected characteristics should receive similar predictions, and (2) differences in predictions across groups of people can only be as large as justified by non-protected characteristics." which means that 1) the predictions should not be dependent on the protected characteristics but only on the non-protected ones, 2) and if the protected ones are dependent on the non-protected ones the outputs should be analysed and justified. Kamishima et al. [66] on the contrary see these two cases not as actual definitions of fair predictions but as potential causes of unfairness. They describe three possible causes of unfairness: ***direct or indirect prejudice*** (what corresponds respectively to cases (1) and (2) of Zliobaité [120]), ***underestimation*** (the difference between the real distribution and the distribution resulting from the training on a finite dataset of a Machine Learning model, of the classified attribute conditioned on the protected feature), and ***negative legacy*** (the unfair sampling or labelling of the training data which results in biases of the trained models). These three aspects of unfairness, contrary to the other definitions, do not refer directly to the difference in treatment between different categories of population but investigate the unfairness for each category of population without comparing them explicitly.

Corbett-Davies et al. [30] also consider fairness as equal treatment of protected and non-protected categories of people, but they highlight three trends in the definitions and corresponding evaluation methods of fairness for the case of algorithms made to select defendants to detain: ***statistical parity*** (equal proportion of defendants detained in each protected group), ***conditional statistical parity*** (equal proportion of defendants detained in each protected group, conditioned on some reasonable attribute of the defendants, such as the number of prior convictions), and ***predictive equality*** (equal accuracy of decisions across race groups, measured with the false positive rate).

Zemel et al. [116] highlight that there are two different goals related to fairness: ***group fairness*** (which is equivalent to statistical parity), and ***individual fairness***. Group fairness is limited because it might have unwanted and unfair consequences (for example when ensuring statistical parity by choosing unqualified individuals in the protected group), that is what individual fairness aims at correcting by ensuring that individuals who are similar with respect to a particular task are classified similarly.

Zafar et al. [115] object that there is a trade-off between increasing the fairness of algorithms according to the current definitions of fairness (parity of treatment or impact between categories of population) and ensuring a high global accuracy of the algorithms. To avoid this trade-off, they propose a relaxed definition of fairness, which they call ***preference-based fairness***s: "under preferred treatment, no group of users (e.g., men or women, blacks or whites) would feel that they would be collectively better off by switching their group membership (e.g., gender, race)."

Hardt et al. [55] object to the demographics-parity definition that they claim is not appropriate for fairness because it only aims at getting equal percentage across groups but does not make sure that the positive classifications in each group correspond to the ground truth classifications. Instead they propose a definition relying on ***equalized odds***: "a predictor Yp satisfies equalized odds with respect to protected attribute A and outcome Y, if Yp and A are independent conditional on Y", Y being the ground truth. This is equivalent to the equality of true positive and true negative rates across groups for binary classification problems. They also proposed a relaxed definition called the ***equal opportunity***: "a binary predictor Yp satisfies equal opportunity with respect to A and Y if Pr(Yp = 1 | A = 0, Y = 1) = Pr(Yp = 1 | A = 1, Y = 1)", what is only the equality of the true positive rates. Zafar et al. [114] propose a similar definition that they call ***disparate mistreatment***: a model suffers from disparate mistreatment when the misclassification rates differ for groups of people from the different protected and non-protected categories. The only difference lies in the metric chosen for the calculation of the misclassification rates which can be overall misclassification, false positive or false negative rates, ...

Most papers are interested in fairness related to the discrimination power of the algorithms' predictions (relationship between the outputs of the algorithms and implicit features) due to the trained prediction models or the dataset used to train them. However, there is also another rarer direction to study fairness. Certain researchers such as Binns et al. [13] study the bias introduced by the persons who participated in the creation of the dataset to train the algorithms on, and its effects on the fairness of the algorithms' outputs, what is close to the negative legacy [66]. For them, a fair algorithm is an algorithm which exhibits equal performance for the different categories of population who participate in the dataset annotation -the categories of population are also usually based on protected features.

2.4.3. Metrics to characterize and evaluate algorithmic fairness

Zliobaité [120] surveys the traditional metrics to measure discrimination in algorithms by distinguishing between statistical, conditional, structural and absolute measures. These metrics all compare the performance or distributions computed over the predictions of the algorithms across groups of people. For example, Chouldechova et al. [28] define "fairness metrics" as metrics which correspond to differences in a particular classification metric across groups, and they take the example of the difference between the false positive rates calculated for one category of the population and the other as an indication of possible unfairness of the algorithm. Binns [12] lists these current metrics for fairness of the algorithms, and points out that none of them is preferable to the other but that their combination to optimize the algorithms is not possible mathematically and thus one metric has to be chosen depending on the task at stake. It is advised not to aim at equalizing the percentage of positive/negative classification rates between the identified groups because it would not take into account legitimate discriminations. Thus, more "nuanced" measures are mentioned such as the equalization of the "accuracy equity" (the accuracy of the classifiers for each group).

Kamishima et al. [66] on the contrary define indexes which enable to evaluate the different causes of unfairness, without explicitly comparing the different groups.

Binns et al. [13] have a tilted angle in their evaluation of algorithmic fairness because they investigate fairness towards the dataset's annotators by comparing the performance of algorithms for data corresponding to different categories of population. They show on a toxicity dataset that the implicit norms that annotators of the data samples have lead to discriminating-biases in the dataset and consequently in the automatic prediction systems. Specifically, they show that the gender of the annotators influence the final labels of the samples in the dataset and therefore training one unique classifier on the majority vote labels leads to higher prediction performances for one of the genders' collective judgements (majority vote). This quantification of the inequality of accuracy performance on the majority vote for different populations is what they define as unfairness of the system.

Several papers are interested in identifying automatically for which sub-groups of the population the algorithms are unfair. Zhang et al. [118] investigate the automatic identification of the sub-groups of protected characteristics for which the algorithms are biased, and propose an iterative algorithm which outputs these most biased sub-groups. Vzliobaite et al. [121] are particularly interested in fairness for algorithms which classify samples which are partly described by sensitive features. They propose measures to quantify how much the outputs of the algorithms are biased towards certain feature values, and how certain categories of population have their accuracy lower than the accuracy of the most frequent demographics. Chouldechova et al. [28] work on the comparison of the fairness of different algorithms. They propose a method to automatically find out which are the sub-groups of the population for which the difference of fairness performance between two algorithms is the largest.

2.4.4. Methods to mitigate algorithmic unfairness

Certain research aim at making the algorithms independent of the sensitive features [121] to make them fairer. It is shown that the unfairness is partly due to the datasets which are not balanced for these sensitive features and different methods are used to resolve the unfairness. Certain papers propose to resample the datasets. For example, Buolamwini et al. [18] investigate biases present in automated facial analysis algorithms and datasets with respect to phenotypic subgroups (based on gender and skin color). They divide the population present in the dataset in categories and measure how well represented they are in the full dataset. They propose a fairer dataset by balancing these categories. Additionally, they define a non-biased evaluation of current algorithms, by computing their accuracy separately for each category.

Other papers define new objective functions [116, 121], or regularization terms for the objective functions of the classifiers [66], which take into account the discrimination. Others propose to train different classifiers for different demographics, such as in [13, 45].

Binns et al. [13] train classifiers for each category of population defined with the protected attributes (gender in their case) separately to make the predictions fairer. They discover that the outputs' performance remain unfair for one of the genders. Second, they highlight that even within each gender category, there is disagreement on the annotations. Although they consider fairness as algorithms' performance equality on collective judgements between different population categories, since there is disagreement between these categories, it seems that this might not be justified to consider fairness along this criterion, but that each annotator judgement should be considered separately.

2.4.5. Discussion

RQ2. The increase in research tackling fairness and accountability of Machine Learning algorithms is very recent. Although there are multiple definitions of fairness, we noticed that unfairness is usually considered as discriminative behaviours of the algorithms, discrimination being related to protected categories of population. Similarly we could define protected categories of population for our task of toxicity prediction, mainly categories based on demographic information like age, gender, education level since we saw previously that these variables influence the perception of toxicity. We could evaluate our models on the different categories separately (with accuracy or other metrics) and aim at balancing the error rates across groups to mitigate the potential unfairness.

Most definitions of fairness proposed in the literature are related to distinct groups of people. However, our final aim is not fairness as a discrimination-related property of algorithms identified by looking at specific categories defined by demographic information. We aim at achieving algorithmic fairness as a property related to the equity of opinions' representation by respecting all the different legitimate opinions in the dataset. Moreover, we are not exactly considering the same task in our problem since our input samples are not the persons to classify and consequently are not dependent on sensitive features (the demographic features), but the samples to classify are sentences whose associated labels are dependent on the persons who judge them and on their possible descriptive "sensitive" features.

Despite these differences, we could get inspiration from these works to measure whether our algorithms are fair in representing the different opinions on a same sample. In order not to focus on possible demographic discrimination and to equally represent all the different judgements on one sample, we could evaluate fairness of our models on an individual-level instead of a group-level, by using the metrics proposed on the group level in the fairness literature. For example we could separate the dataset into bins of annotators or sentences divided according to specific criteria such as the average agreement rate of the annotators with the opinion of the majority (equality of representation between annotators' opinions) or the sentence ambiguity (equality of representation between clear and ambiguous sentences), and comparing the performance of the algorithms' predictions on these different bins.

2.5. Crowdsourcing and subjectivity

In the following section, we study what are the techniques employed to crowdsource subjective properties of data samples. The aim is to understand how to proceed to collect and evaluate a training dataset.

2.5.1. Methodology to search for the papers

In this section, we search for the literature dealing with the annotation of subjective tasks. Therefore, we started by using the query "crowdsourcing subjective + (task or annotation)". Later on, we also found out that Quality of Experience experiments were related to subjective tasks, so we added the query "crowdsourcing quality of experience". First, we retained the papers explaining how to ensure the quality of the crowdsourcing outputs. We also chose the papers which explained methods or metrics to evaluate the quality of crowdsourcing tasks related to subjective data.

2.5.2. Crowdsourcing methodology for corpus annotations

Sabou et al. [91] present guidelines on how to operate crowdsourcing tasks for corpus annotations. They identify four steps in the crowdsourcing process and give recommendations: the project preparation, the data preparation, the project execution, and finally the data evaluation and aggregation. The project preparation consists in defining the specificities of the crowdsourcing task, how it will be presented to the workers. This has an importance when using crowdsourcing for subjective evaluations because techniques to ensure the annotation quality must be set up at this step; and should be supplemented by pre-filtering the crowd workers for example based on their language proficiency or on their answers to training questions. The last step also merits investigation since it might not be meaningful to aggregate all the annotations for one sample together when we want to leverage the subjectiveness of the worker judgments. These two steps are reviewed in the following sections.

2.5.3. Identification of variables influencing subjective annotations

Hossfeld et al. [59] perform a study to determine the parameters which have an influence on the evaluation of the YouTube video quality. For that, they gather a number of parameters, and compute metrics indicating correlations between these parameters and the annotations. First, they compute the Spearman rank-order correlation coefficient between the subjective user rating and the variables. Then, they also use Support Vector Machines to classify between high and low-quality videos, and check which features (corresponding to the variables) obtain larger weights, what suggests more importance.

Ghadyaram et al. [53] separately look at the influence of gender, age, and experiment set-ups, these last variables having more influence in image quality appreciation. For that, they divide the annotations of the workers belonging to the different demographics categories and compare whether they are similar or different.

Wulczyn et al. [111][9] investigate biases when crowdsourcing to constitute a dataset of toxicity in Wikipiedia comments. They compare using binary labels ("probability that the majority of annotators would consider the comment an attack") and empirical distribution labels ("predicted fraction of annotators who would consider the comment an attack"), to train Machine Learning classifiers that they evaluate with the Spearman rank correlation and the Area Under the Curve. They show that aggregating the annotations as empirical distributions leads to better classifier performances. However, they do not investigate what are the crowd worker characteristics which influence their ratings. In their dataset, they record some of the workers background information.

2.5.4. Techniques to ensure crowdsourcing quality

General methods for ensuring quality in crowdsourcing

First, we look at the literature which investigates crowdsourcing for Quality of Experience (QoE) evaluation, which is a subjective task since this evaluation depends on people's appreciation of the samples based on their internal beliefs.

Hossfeld et al. [59] study QoE of YouTube users. To collect subjective evaluations of the annotators while ensuring a certain annotation quality level, they use several techniques. First, they make use of gold standard questions over samples whose evaluation is objective. Additionally, they perform consistency tests: they ask several times the same question to workers (formulated slightly differently), and eliminate workers whose

[9] `https://github.com/conversationai/unintended-ml-bias-analysis`

answers vary a lot. They also ask content questions (questions about the content of a sample) with objective answers; and design mixed answers: they vary the presentation of the answer scales so that a worker always clicking on the same answers will give inconsistent results. Finally, they suggest that user monitoring can be employed, for example by checking that the users at least spend a minimal amount of time on each question, they do so by monitoring the focus time on the crowdsourcing webpage. In summary, to filter the crowd workers, they proceed in three steps: they eliminate users who provide wrong answers to 1) content questions, mixed answers or consistency questions; 2) gold standard questions; 3) user monitoring verifications. They show that their method is efficient since each step eliminates 25% of the workers. Similarly, Redi et al. [84] combine several methods for ensuring quality such as content questions, and target countries with a large English speaking population. They also add a mandatory training for each worker. They filter the workers based on their answers to the content questions and on their attention time, and they delete the workers who are considered as outliers.

Ghadiyaram et al. [53] prefilter workers by selecting only the ones who have a high confidence value on Amazon Mechanical Turk. Additionally, they show several times the same samples and exclude workers whose annotation difference exceeds a certain threshold on these samples. Moreover, other studies [16, 74] perform worker selection, by employing the workers who possess the specific skills required for the specific task (for example by modelling the workers based on their social media data).

Ribeiro et al. [88] choose not to perform pre-filtering of workers but they do post filtering. Once enough data are collected, they compute the Pearson sample correlation coefficient between the MOS estimates from a worker and the global MOS estimate. They eliminate the worker annotations from the workers whose correlation value is lower than a threshold. They additionally pay bonuses to best performing workers.

Snow et al [97] recalibrate the data to correct the individual biases, by comparing each worker annotations to the gold standard examples, and modifying the labels according to these comparisons (detecting noisy annotators and anticorrelated workers).

Alonso et al. [4], although not focusing on subjective QoE evaluation but relevance assessment, investigate the importance of different crowdsourcing aspects. They propose to have the workers take a qualification test (general questions about the task topic) along the gold questions, in order to filter them. Additionally, they show that the quality of the user interface (instruction clarity, text presentation) impacts the quality of the annotations. Finally, they ask one open-ended question to get user feedbacks, what enables them not only to get useful incites on their task design, but also to detect spammers.

These studies do not aggregate the annotations of each worker.

Methods specifically interested in ensuring the subjectivity of the workers

Speck et al. [101] aim at collecting a dataset for music pieces classification. They ask crowd workers to choose arousal and valence values while listening to a music piece. Since this task is very subjective, they are specifically interested in removing incorrect workers while making sure they account for the workers subjectivity. For that, they train a one-class SVM with the positive expert labels (music Information Retrieval researchers) on a few data samples, later they exclude workers whose annotations on the verification samples are on average outside the decision boundary of the trained SVM. They assume that the expert labels differ enough to represent the different possible judgments over one sample, and therefore the crowd workers annotations should not differ much from them. They also add a second stage of filtering simply based on the consistency of the answers. Even though they present high precision, recall and F-measure with this method to automatically classify workers, they do not compare it with other baseline methods.

Brew et al. [17] perform sentiment classification of news content and collect data via crowdsourcing to classify news between positive and negative sentiment, as well as relevant or irrelevant (dealing with economics topic or not). They show that certain samples are more subjective than the others, depending on the worker agreement rate; and that their learning algorithms learn better using high-agreement samples. Moreover, they show that coverage (using more data samples with fewer annotators) enables better learning than consensus (using less data samples with more annotators), in cases where the consensus rate is globally high. They conclude that it is necessary to identify which workers are close to the consensus opinion and to select them for annotation.

Dumitrache et al. [42] are interested in crowdsourcing for medical relation extraction (for a Watson service) since it is not possible to collect enough data with expert annotators. They compare semi-manually labeled annotations to 1) automatic annotation method, 2) expert annotations, 3) single crowdsourced annotations, and to 4) aggregated crowdsourced annotations. They show that with appropriately-tuned aggregated crowdsourced annotations, they get a higher annotation quality than with the other annotation methods.

Moreover, they show that when choosing the correct parameters, these non-binary aggregated crowdsourced annotations enable to train classifiers with an higher accuracy than with the other constituted datasets. When investigating the number of crowd workers required, they find that with 10 annotators, the annotation quality is maximal. Their annotation aggregation method is as follows: 1) first, they add all the workers annotations in one vector (sentence vector), 2) then, they compute for each dimension the cosine similarity between the vector and the corresponding unit vector, 3) finally, they rescale the resulting vector with positive and negative values between [0.85;1] and respectively [-1;-0.85] according to a determined threshold defining which values are positive or negative. In [43], they first run a sentence metric to eliminate the sentences which are too ambiguous before running the annotators metric to eliminate low-quality workers with the worker disagreement metric [7]. They train models on the cleaned data.

2.5.5. Metrics to evaluate crowdsourcing quality

Hossfeld et al. [59] consider two different types of reliability of the user studies: intra-rater and inter-rater reliability. The first reliability considers the consistency of the annotations of one unique annotator (averaged over all the annotators), by computing the Spearman rank correlations over the answers of each user. This is possible because each user rates three videos for which only one known parameter varies with a known effect on the subjective judgment. The second reliability measures the agreement between annotators by computing the Spearman rank-order correlation coefficient between all user ratings and the varied stalling parameter for all user ratings in a campaign. Ghadyaram et al. [53] also compute inter and intra subject consistency with gold standard data. They also employ gold standard questions to compute the correlation between the annotation Mean Opinion Score (MOS) and the laboratory data MOS to check reliability of the annotations.

Similarly, Snow et al [97] compare the inter-annotator agreement of individual expert annotations to that of single non-expert and averaged non-expert annotations. They also compute how many non-experts are needed to get similar performances than with experts.

Moreover, to evaluate whether the disparity between user judgments is too high to be realistic or is low enough to suggest high quality annotations, Hossfeld et al. [59] use the Standard deviation of Opinion Scores (SOS) hypothesis [58], which determines a relationship between the SOS and the MOS. It is specified that this relationship holds only for Quality of Experience studies, and these studies must consist in ratings on a K-point scale.

Ribeiro et al. [88] explain a method to compute the MOS score and the confidence interval of the crowdsourcing experiment. Redi et al. [84] employ normalized MOS scores. They compute the correlation between the crowdsourcing experiment and the laboratory experiment MOS scores. Similarly, Keimel et al. [67] and Speck et al. [101] compute the correlation between the results obtained by crowdsourcing and in lab-experiments.

Hsueh et al. [62] define three quality metrics. They compute the deviation of each worker annotation to the gold standard (what they call the noise level). They consider the gold standard as the majority voting annotations. Secondly, they compute the ambiguity of each sample based on the sample annotations. Finally, a third metric called confusion combines the previous two metrics. These measures are used to remove certain annotations from the dataset, and improve the accuracy, especially with the confusion. Thus, not only do they show that it is useful to remove low-quality annotations for training, but they also point out that active learning would then enable to considerably reduce the amount of data needed to train classifiers.

Chen et al. [26] crowdsourcing approach is different since they ask the workers to compare each time between only two samples to simplify the task. To measure the quality of the annotations, they check the individual consistency and the overall consistency of the annotations.

Dumitrache et al. [42] highlight as factors which create disagreement between workers not only the expertise of the crowd workers, but also the way the task is defined, and the ambiguity of each sample. Indeed, interpretation and annotation rely on three different related concepts named as the "triangle of reference" [8] (the interpreter, the sign and the referent): "the interpreter perceives the sign (e.g. a word, a sound, an image, a sentence) and through some cognitive process attempts to find the referent of that sign (e.g. an object, an idea, a class of things)". The CrowdTruth framework proposes several metrics to evaluate the three corners of the triangle: the crowdsourcing workers (low quality or spam), the sentences (clarity), and the relations (similarity). Representing the annotations as vectors, they iterate to compute several cosine similarities in order to obtain quality metrics.

2.5.6. Discussion

RQ1. In this section, we investigated the best-practice methods for crowdsourcing annotations and we gave a special attention to the methods dealing with subjective data.

First, considering that we want to use demographics information to make the subjective predictions, we should investigate whether there are correlations between the different features and the labels in our dataset, in order to identify whether some demographics are more important than the others to predict offensiveness perception. This can be done using one of the methods cited above.

In order to run a crowdsourcing task to collect high-quality data on subjective undesired speech perception, we should filter the crowd workers. For that, we should use one or several techniques cited above in order to have a way to measure the crowd workers quality. Mainly, we should ask some questions with objective answers about the sentence sample context, and we could also find data samples with obvious offensiveness judgements, in order to judge which crowd workers input correct answers. Moreover, considering that in our dataset, the workers did not all annotate the same samples, we cannot use the usual crowdsourcing quality metrics. We could instead use the CrowdTruth framework in order to identify which are the low quality workers, the ambiguous sentences, and possibly the annotation classes which are too similar. Since we aim at keeping the annotators and annotations which might differ from the consensus opinion, but which are still true, we can use one or several of the above methods to filtrate the annotations of totally wrong users. However, we have to pay attention not to eliminate workers which give correct but unusual answers.

Finally, it would be useful to detect the most ambiguous samples with the annotator agreement rate or [43]'s metrics, and request more annotations for these samples; while also eliminating the low-quality workers. This way, we would have more data to train the algorithms on the subjective samples and improve their performances on these types of data.

2.6. Summary

With this chapter, we produce the first contribution of the thesis: the extensive literature review about Machine Learning, Crowdsourcing, fairness and subjectivity, with a usecase in sentence toxicity prediction.

Related to ***toxicity*** and related speeches (RQ1), we found out that toxicity is a subjective property of sentence, and that toxicity perception depends on three main types of variables (the sentence characteristics, the sentence context, and the individual judge internal characteristics).

For the ***Machine Learning algorithms*** to perform sentence toxicity prediction (RQ3), we found out several algorithms which are traditionally used. We will experiment with these ones since it is not possible to deduce the most performing ones from the literature. To adapt the algorithms' predictions to each user, we identified to main possibilities: adapting certain parameters of the algorithms to each user, or inputting additional user-related inputs to the algorithms.

Concerning the ***evaluation*** of the algorithms (RQ2), we found out that usual evaluation methods do not enable to make conclusions concerning the fairness of the algorithms and that we consequently need adapted evaluation metrics. Research currently interested in fairness look at the discrimination power of the algorithms. However, we want to look at a more general kind of fairness, fairness towards each user of the algorithms, and therefore we will adapt the usual definitions to our purpose.

Concerning the creation of the ***dataset*** (RQ1), we found out that the toxicity-related speeches have blurry definitions and it is advised to use a precise definition to collect toxicity annotations via crowdsourcing. Several methods can be employed to ensure a high-quality of the crowdsourced data. Moreover, in order to decrease the amount of variations in the crowdsourced annotations, we should fix as many variables as possible which influence the perception of toxicity-related speeches, mainly the variables related to the sentence context.

3

Dataset to study sentence toxicity as a subjective property

3.1. Introduction

In this section, we are interested in the creation of datasets for the training of algorithms for subjective properties classification, and aim at answering the first challenge through the first question (RQ1): ***how can a dataset be built to train algorithms for the prediction of subjective properties?*** There are three main directions in this study. We prove that our main research question is justified with the use-case of toxicity classification, by investigating whether the chosen dataset (Jigsaw toxicity dataset) contains multiple subjectivities. We investigate the crowdsourcing techniques to harness the different subjectivities in the dataset while creating it. We reflect on how to collect the annotations of subjective properties at a low cost. Each of these aspects correspond to different sub-questions.

1. *How can the prediction of subjective properties be studied from a Computer Science perspective?* **(RQ1.1)**

    (a) *Is toxicity a subjective property of sentences? What are the human-related variables which influence sentence toxicity perception?*
    → We search the psychology literature about offensiveness (and possibly toxicity) to show that it is a subjective property of sentences, and to find a set of variables influencing toxicity perception.

    (b) *Are there different valid opinions in available toxicity datasets?*
    → We show that the Jigsaw toxicity dataset exhibits different valid toxicity judgements per sentence, and therefore that sentence toxicity is a valid use-case to study subjective properties prediction. We assume that disagreement between annotations of a same sample is a sign of different possible opinions. After cleaning the selected dataset, we compute the distribution of disagreement rate of the workers' annotations with the majority-voting label and show that there are different valid opinions for each sentence.

2. *How to collect crowdsourcing annotations of high-quality when the property to annotate is subjective? How to identify and remove spammers' annotations while keeping the valid annotations?* **(RQ1.2)**
    → We review the Crowdsourcing literature and list the methods to create clear crowdsourcing tasks and to filter low-quality crowd workers and annotations during these tasks and during the annotation post-processing step. We investigate to which extent these techniques are applicable to subjective classification tasks, with a special focus on the CrowdTruth method for which we manually verify that it is applicable to our use-case.

3. *How to collect crowdsourcing annotations on a large dataset while maintaining a low cost?* **(RQ1.3)**
    → We hypothesize that certain clustering methods enable to cluster the samples on which to collect annotations so that only annotating a reduced number of samples inside each cluster and spreading the annotations inside the clusters would provide correct annotations for the whole dataset. We reject this hypothesis by conducting experiments on the retained dataset.

3.2. Toxicity in psychology and dataset choice

In this section, we focus on the first sub-question: ***How can subjective properties prediction be studied from a Computer Science perspective?*** **(RQ1.1)** Our aim is to study the prediction of properties which are subjective, in other words we want to study properties on which there is disagreement between the judgements of different people. Consequently, supposing that sentence toxicity is subjective, we hypothesize that:

H1: **the use-case of sentence toxicity is a valid domain of application for the study of subjective properties from a Computer Science perspective.**

Here, we validate this hypothesis by verifying whether sentence toxicity is a subjective property in theory, and we later verify it in practice by investigating whether toxicity datasets found in Computer Science research comprehend disagreement on the judgements of sentence toxicity. The quality of these datasets should first be evaluated and possibly improved before being able to judge whether several valid subjectivities are contained in them. That is why we proceed here only to a theoretical investigation of sentence toxicity and make a choice of a Computer Science dataset to investigate. In Section 3.3 we proceed to the quality evaluation so that in Section 3.4 the presence of different subjectivities can be judged.

3.2.1. Sentence toxicity as a subjective property in the psychology literature

First we look at the theory about sentence toxicity found in the psychology literature to investigate its subjective character and answer: ***is toxicity a subjective property of sentences? What are the human-related variables which influence sentence toxicity perception?*** "Subjective" is defined as "based on or influenced by personal feelings, tastes, or opinions."[1]. From our literature review on the toxicity of Web content and related concepts (hateful speech, abusive language and offensiveness) (Section 2.1), we conclude that offensiveness is a subjective property of sentences. Although toxicity and offensiveness might be two different concepts, since there is no precise definition of toxicity we claim they are closely related and consequently we can consider that toxicity is also a subjective property of sentences.

The literature enables to draw a list of the influencing variables in sentence toxicity perception. These variables are listed in Table 2.1 (Section 2.1) and are divided into three categories -the *variables related to the internal characteristics of the person reading the sentence* (the most important ones being gender, age, ethnicity and education level), the *variables related to the sentence characteristics*, and the *variables related to the sentence context.* According to the definition of subjectivity, a dataset which exhibits subjective judgements should comprehend variations along the first category of variables, while variations on the other categories would participate to the differences in toxicity perception but not because of the subjective character of toxicity. Variations of the sentence characteristics are intrinsic to the sentences themselves, they are common to each reader of a sentence and thus do not participate in the variation of opinions about one sentence. Variations due to the sentence context do not participate to the variation of opinions if the context is explicit and thus common to all the readers. If the context is implicit, it might lead to different interpretations of a same sentence by different persons and consequently the dataset would exhibit multiple judgements but these would not be directly related to the subjective character of toxicity. Thus, a dataset to study subjective property prediction for sentence toxicity should be constituted of different sentences and different types of readers of these sentences, and at best should present the sentence context explicitly.

3.2.2. Subjectivity in Computer Science datasets of sentence toxicity

We now choose a Computer Science dataset of sentence toxicity in order to analyse in the following sections whether the subjectivities of toxicity can be found in practice in the Computer Science domain.

Requirements for the dataset

We wish to study the toxic property of sentences and how its subjective character influences crowdsourcing tasks and Machine Learning models. Based on the findings of the previous subsection, we define a list of requirements that a dataset should respect to enable this study.

In order to train Machine Learning models, the dataset should contain several sentences and toxicity judgements of these sentences. To compare multiple algorithms (traditional Machine Learning classifiers or Deep Learning neural networks) which require more or less data to be trained on, we need a large dataset of many sentences. In previous papers working with Deep Learning for hate speech detection, the size of

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

the datasets varies greatly, between 6655 and 10 million samples (Section 2.2), but the performance reported for the different algorithms trained on these different datasets are similar. Therefore, we assume that 100000 comments should be enough to predict binary labels, and it might also be sufficient to predict empirical density labels since it is done successfully in [111] with a dataset of this size.

To study the subjective character of sentence toxicity, we need to make the different variables which influence toxicity perception vary independently. Consequently, we need different sentences to investigate how judgements differ according to the variables related to the sentence characteristics ; and multiple toxicity judgements per sentence by different people whose internal variables vary. In order for the sentence context variable not to vary simultaneously with the other two variables, it should always be the same, or it could vary but should be explicit so that the annotators all base their judgements on the same context.

Existing datasets are collected via crowdsourcing with multiple annotators per sample. Having access to the different annotations that each annotator gave during the crowdsourcing task and not only to the final labels after post-processing of the crowdsourcing results would enable to study the different toxicity judgements emitted on a same sentence. The more annotations available, the higher the chance to collect differing judgements due to differing annotator internal variable would be. In the crowdsourcing literature, it is explained that collected annotations might be incorrect and it is reported that using 10 annotations per sample leads to the highest algorithm accuracies with distributed labels when training Machine Learning models on these annotations [42]. However this number of annotators is adapted to the case where one unique label or density estimated labels are predicted for each sample. In our case, we aim at considering several labels correct depending on the annotators of the sentence, as long as the annotators gave their true opinion on the sample. Thus we cannot affirm that 10 annotations per sample are sufficient for our problem.

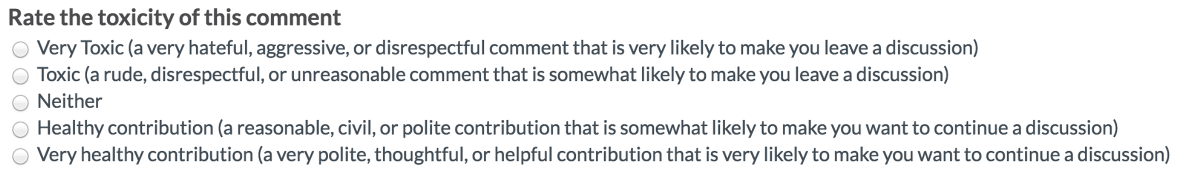
Dataset choice

From the literature review, we found multiple datasets available related to sentence toxicity, hatefulness or flames, but none of the datasets completely fulfil the requirements. Most of them do not make the annotations of the annotators available but only give access to the final label. We eliminate these datasets since they do not allow to study the differing judgements. The Jigsaw toxicity dataset [111] is a large dataset freely available of the Google team Jigsaw, with around 100000 Wikipedia comments rated as toxic or not along a $[\![-2;2]\!]$ scale (a more detailed description of the dataset is given in Appendix A.2). It presents around 10 annotations per sample (1598289 annotations in total) collected on CrowdFlower [2], with information about part of the annotators (gender, first language, age group, education level). It should enable to infer labels of high quality, and to study correlations between people's internal variables and toxicity judgements. It is the only dataset with these characteristics, that is why we set up to conduct our experiments on it. The toxicity score is ranging from very toxic (-2), to neutral (0), to very healthy (2). The Jigsaw team then aggregated the scores into toxicity labels: 1 for toxic and 0 for non-toxic.

The main people's internal variables cited in the psychology literature are the age, gender, education, and ethnicity. The Jigsaw dataset nor contains information about the annotators' ethnicity, neither on the sentence context. That leads to limitations that we could overcome by collecting a new set of annotations based on the Jigsaw dataset. We suppose for now that the dataset should enable to study the subjective character of toxicity since different opinions for the available data samples should be found in the dataset due to the different internal characteristics of the annotators. Subjectivity is only related to the people's internal characteristics but not to the sentence context, but because we can not eliminate this second variable we decide to ignore it for now. For the whole project, we consider that the crowd workers background varies along 3 dimensions (gender, age, education level), since the first language was not reported in the psychology literature as an influencing factor for offensiveness perception.

The following crowdsourcing task (Fig. 3.1) was presented to the crowd workers when asked to label the comments. In order to ensure annotation quality, the workers were pre-filtered according to the correctness of their answers to 10 golden questions, and additionally the annotations of workers who gave opposite answers to the same question were removed. Although it is not time-feasible in the extent of the master thesis, rerunning a crowdsourcing task on the sentences with the lowest agreement rates or on the least frequent demographics information, and capturing additional information about the annotators would enable to investigate how large the dataset should optimally be for our task.

[2] https://www.figure-eight.com/



Rate the toxicity of this comment

○ Very Toxic (a very hateful, aggressive, or disrespectful comment that is very likely to make you leave a discussion)
○ Toxic (a rude, disrespectful, or unreasonable comment that is somewhat likely to make you leave a discussion)
○ Neither
○ Healthy contribution (a reasonable, civil, or polite contribution that is somewhat likely to make you want to continue a discussion)
○ Very healthy contribution (a very polite, thoughtful, or helpful contribution that is very likely to make you want to continue a discussion)

[a] `https://github.com/ewulczyn/wiki-detox/blob/master/src/modeling/toxicity_question.png`.

Figure 3.1: The question asked to the crowdworkers for the dataset creation. Source: Jigsaw team's Github [a]

3.3. Crowdsourcing treatment of dataset annotations of subjective properties

Limitations of crowdsourcing, explained in the introduction and literature review, are mainly that crowdsourcing techniques to solve annotations' quality issues rely on annotation filtering and annotation aggregation based on the idea that one unique judgement of the sample is correct, whereas we are interested in keeping several different but all valid judgements about a same sample. Consequently we need to investigate how to collect annotations of high-quality without using current crowdsourcing techniques, what is the second sub-question of this chapter: ***How to collect crowdsourcing annotations of high-quality on subjective labelling tasks? How to identify and remove low-quality annotators' annotations while keeping the valid subjectivities?*** **(RQ1.2)**

To answer these questions, we proceed in two steps. First, we investigate whether the collection method of the Jigsaw dataset is in par with the methods mentioned in the literature to *design crowdsourcing tasks* to collect high-quality annotations, by searching the literature from three different fields (crowdsourcing, Machine Learning and psychology), and create a list of possible improvements in the crowdsourcing task. Second, we investigate the *post-processing methods of crowdsourcing results* to clean the collected annotations, and check whether they are applicable to annotation tasks for subjective properties.

3.3.1. Design of the crowdsourcing task

We investigate ***how to design a crowdsourcing task to collect annotations of high-quality on subjective properties***. Current research about the design of crowdsourcing tasks is aimed at making the task as clear as possible in order to eliminate as much ambiguity as possible and get as similar annotations as possible, and at facilitating the understanding of the task for the annotators to be fast and accurate at providing annotations. These are also the objectives of task design for the annotation of subjective properties, mainly elimination of ambiguity is of special focus since only the variables related to the annotators' individual characteristics should vary. This leads us to formulate the following hypothesis:

> H2: ***general crowdsourcing research on task design is applicable to the collection of annotations of subjective properties***.

In order to verify this hypothesis, we could compare the quality of the annotations returned by two crowdsourcing tasks on a same dataset, one task following the design recommendations given in the crowdsourcing literature and one task without a careful design (baseline). However, due to cost limitations, we cannot run a new crowdsourcing task and we cannot verify the hypothesis. Instead, we compare the recommendations of the crowdsourcing literature on crowdsourcing task design to 1) the observations of the psychology literature interested in studying subjective properties related to toxicity, to 2) the suggestions given by researchers who created datasets for sentence toxicity prediction, and to 3) our observations of the sentences in the Jigsaw dataset. We verify whether the methods proposed in the crowdsourcing literature follow the suggestions from other fields of research and could solve the limitations observed in the task of annotating subjective properties.

Identification of limitations and suggestions in the literature

From Section 2.1.5 where we look at how ***psychology studies*** conduct experiments to collect judgements over hate speech, we see that tests consisting of multiple questions over a sample whose answers are averaged to obtain one unique judgement are used. This enables to get the correct perception of the test subject about the sentence. The questions asked are part of psychology questionnaires, or are grouped around a same concept, or are sets of propositions that the annotators must rate depending on the validity for the samples. The samples to judge are usually scenarios, what provides a context to the sentence to judge.

In Sections 2.2.3 and 2.1.5, we investigated how the ***Machine Learning researchers*** interested in toxicity prediction collected their datasets and what future recommendations they gave. The Machine Learning papers which analyse the collection of annotations to build datasets related to human judgement of tweets or other social media posts highlight the presence of disagreement between annotators despite the use of psychology tests to obtain the labels. They propose several explanations for the disagreement: the psychology questionnaires are not adapted to social web data, the sentences lack context (such as the writer, the target and the context of the potential discussion) and the annotators have different perceptions depending on their subjectivities. In this work we aim at highlighting the different judgements annotators have about one same sample, therefore we should eliminate the disparities due to the lack of context but keep the ones due to the annotators' subjectivities.

The ***crowdsourcing literature*** (Section 2.5) highlights that the questions asked to the annotators should be as clear as possible in order to collect high-quality annotations. For that, the questions should be as simple as possible, possibly the property to annotate can be explained before starting the task, and examples can be given of the expected annotations for some samples.

Analysis of the design of the Jigsaw dataset crowdsourcing task

We analyse the task created to collect the Jigsaw dataset (Fig. 3.1) based on the recommendations from the literature and our observations on the data. The aim of the task is to collect high-quality annotations, what we define as annotations which reflect the annotators' judgements on the data samples to judge on one property, excluding any possible judgement ambiguity coming from the task design and data samples.

Design of the crowdsourcing question. In the case of the Jigsaw dataset, only one proposition is given to rate to the annotators, what is different from psychology studies where multiple propositions are given to the participants to rate them and average the results. Using multiple propositions would enable to collect more accurate judgements from the annotators and would help detecting possible spammers (in cases where the answers do not match between each other) leading to higher-quality labels. Moreover, the scale ranges from 1 to 5 whereas the psychology literature have upper ranges varying from 6 to 12, what possibly changes the perception of the annotators.

Design of the presented samples to annotate. From the psychology literature we identified the three main variables which influence the perception of offensiveness and by extension toxicity. 1) The annotators' individual characteristics are human-related variables and so they are not dependent on the design of the crowdsourcing task, these are the variables we do not interfere with. 2) The sentence characteristics are what makes the data sample and therefore can not be modified. 3) The sentence context (target, author, speech in the overall discussion) consists in several variables which influence the perception of the speech. These variables are not represented in the crowdsourcing task but they are also pointed out by the Machine Learning papers, and they are specified in the psychology literature. Therefore we propose for further crowdsourcing tasks to give out information to specify the sentence context of each data sample. It would enable to eliminate one dimension of the causes of ambiguity. We give examples of sentences where the lack of context leads to disagreement between annotators in Table 3.1.

sentence	annotations	causes of disagreement
"Ummm... you are very narcisistic. You wrote an article about yourself."	-1(4), 0(6)	The target of the sentence is unknown. Subjective perception of the adjective narcissistic.
"Everywhere, you were also very disruptive as well."	-2(1), -1(2), 0(5), 1(2)	No context to identify whether the truth or a criticism is reported.
"Shush sweetie, the adults are talking."	-1(5), 0(3), 1(2)	Lack of context about the actors of the discussion: it is difficult to judge whether the sentence is a mockery or not.

Table 3.1: Example sentences from the Jigsaw dataset which lack context indications.

Design of the explanations of the question. Another cause of misunderstanding and ambiguity in the task is the design of explanations surrounding the question (namely explanations of the terms of the question) asked to the annotators. Sentence toxicity is not explained at the beginning of the task. It is requested to rate the toxicity of a sentence on a [-2;2] scale, but no example is given for each scale value, only descriptions (accumulation of different adjectives which do not refer to the same concepts), what makes it for example difficult to distinguish between neutral, healthy or very healthy sentences. The lack of instructions also make the range of valid answers broad, such as for sentences which convey a positive message but also use abusive language or criticize a person or a group. It is not specified how annotators should judge these sentences with two opposite toxicity aspects. The sentence might be healthy for the target of the speech (e.g the target is thanked by the person writing the sentence) but toxic for the persons mentioned in the speech (certain persons are criticized possibly using aggressive language). We cite examples of these sentences with the annotations associated in Table 3.2. This leads to differing interpretations of the questions by the annotators.

sentence	annotations	causes of the disagreement
"Oh... I.. yeah i feel like an ass now...."	-2(1), -1(7), 0(2)	Self-blaming and the rudeness of the vocabulary might be judged differently by the annotators without instruction.
"so wait, mr porcupine, when did you say you were planning on blocking yourself?"	-1(5), 0(3), 1(2)	No instruction say how to resolve the lack of information on the context (interpretation of the name used to qualify the target and of the joke).
"I hereby wish to thank you for your continuous efforts in protecting our templates from those gutless vandals who get their sick kicks off removing them. May the Force be with you !"	-1(4), 0(2), 1(4)	First and last parts are positive while the middle might be toxic since it is disrespectful towards a specific target. No instruction to deal with two opposite judgements.
*"[...] including your friends and buddies maybe you can do that 100 times a day (let me explain it in your IQ level. Say you have 33 buddies like you. 33*3 = 99 edits.) [...] Your childish desperations trying to humiliate me with suggestions regarding the sandbox are just matching the low ethics and discrimination policy of a person like yourself. [...] You just have a good sense of humour. I suggest you read Merchant of Venice of Shakespeare. [...] that is going to help you be cleansed from your racist feelings against the fellows of other nations. [...] Sincerely."*	-2(1), -1(1), 0(5), 1(3)	The author exposes his rights and uses some respectful language, but also mocks the target of the speech. No instruction given for "mixed" sentences. Lack of context does not enable to know whether the respectful words are sarcastic.
"Did you even bother to read what it said? He said he wanted me to delete it. Wake up."	-1(2), 0(6), 1 (2)	First and second sentences are neutral but the last one is negative. Lack of instructions for "mixed" sentences.
"The transition between the first two paragraphs is horrible. I'm not sure how to fix it; the content in para 1 does not appear related to para 2."	-1(3), 0(1), 1(6)	The sentence gives constructive criticism, but it might be considered negative with the use of the adjective "horrible".

Table 3.2: Example sentences from the Jigsaw dataset which receive differing judgements because of a lack of precisions in the crowdsourcing task.

Sentences with multiple interpretations due to annotators' subjectivities. If the crowdsourcing task was totally clear, the disagreement would only come from the perceptions that annotators have of the toxicity of the elements of the sentence, and not from the ambiguity created by the lack of information about the sentence or the task. We cite in Table 3.3 example sentences which are simply perceived differently depending on people's subjective perception (and possibly some interpretation of the context).

Discussion, conclusion and recommendations for our task

From the above observations, we were able to answer **RQ2.2** by identifying the main limitations of current crowdsourcing tasks for the collection of toxicity annotations: the explanation of the task lacks details (explanations of the terms of the question, of the scale and citation of examples, instructions about extreme cases), the design of the question might not be adapted to the collection of judgements of subjective properties (use of multiple-question questionnaires in psychology but of unique questions in Computer Science), and the samples lack context (sentence conversation context absent in Computer Science but present in psychology experiments). On the contrary, the crowdsourcing literature advices for disambiguation and clarity of the question for example by giving examples of annotations. It does not specifically address the issue of choosing the question to ask the annotators based on psychology literature -what seems to be a point which would merit more investigation-, but it does address the other current limitations. Therefore we conclude that ***hypothesis H2 is partly verified: general crowdsourcing research on task design is applicable to the collection of annotations of subjective properties, but the choice of the question asked to the annotators should be investigated in more details in future research.***

We now propose a list of recommendations to run a new crowdsourcing task on the same dataset.

sentence	annotations	causes of the disagreement
"if you want to leave the article a bunch of lies, controlled by a child. okay. you better be able to prove what you say. it is not vandalising when i correct lies. you better contact someone who knows the truth before printing lies. there was never a porsche motor in a yugo, etc. i guess this 13 year old knows more about cars than i do."	-1(3), 0(5), 1(2)	Sarcasm is subjective (perceived toxic or not). Potential criticisms are judged differently.
"WHY CAN I NOT POST? PLEASRE UNBLOCK ME I DO NOTHIN WRONG FRENS I DO NOT COMPREHNDE WHAT IS GONE HAPPEN"	-1(2), 0(4), 1(3), 2(1)	Lack of instructions when no explicit target appears. Neutral sentence but use of upper case letters appear aggressive.
"I already asked. He pretty much told me to stick my head in a bucket of lava."	-1(6), 0(3), 1(1)	Joke or real attack. Possible lack of context or annotator's subjective perception.
"What the hell is wrong with this thing? why are my changes not showing?"	-2(1), -1(5), 0(3), 1(1)	No target. Annotator's subjective perception, or lack of instructions about sentences without precise target.

Table 3.3: Example sentences from the Jigsaw dataset, whose interpretation is mainly influenced by the annotators' individual characteristics.

- Add context information to each sentence sample. This context can consist of the surrounding conversation of the sentence, information about the writer or the target of the sentence, ...
- Clarify what toxicity and the different labels ([-2;2] range) are, possibly with examples.
- Give instructions on how to judge extreme cases such as sentences which contain both negative and positive aspects, sentences without direct target or sentences with self-criticism.
- Adapt psychology questionnaires to the task of judging toxicity in order to aggregate several answers from each annotator into one judgement (it could give more accurate and consistent judgements instead of asking to rate the toxicity on the [-2;2] scale which is difficult to interpret).

With these restrictions on the crowdsourcing task, especially if we choose to focus on sentences of one specific type of hate speech, we could aim at reaching less than 30% disagreement since this value is given by most papers which perform collection of annotation about subjective quality of experience.

3.3.2. Crowdsourcing treatment of the dataset: annotation cleaning, low-quality annotators and annotations removal

In this sub-section, we are interested in crowdsourcing methods to filter out wrong annotations collected from crowdsourcing tasks. The research sub-question we answer is ***how to identify and remove low-quality annotators' annotations while keeping the valid subjectivities?*** **(RQ1.3)** The study of the literature interested in crowdsourcing result processing (Section 2.5.4) conducts us to formulate an hypothesis concerning crowdsourcing for subjective properties. The literature is divided into two main directions. On one hand, certain research recommend to filter annotators during the crowdsourcing task. The crowdsourcing literature advocates for using several types of questions combined with varied question presentation to check whether the annotators answer randomly or consistently: gold questions, content questions (which are not subjective), consistency tests. User monitoring, the use of Amazon Mechanical Turk user scores, the use of qualification tests are also proposed to filter low quality workers. In the Jigsaw dataset, the workers are filtered using 10 golden questions and consistency tests over the annotators' answers. Additional methods could be used if required to run a new crowdsourcing task. On the other hand, other studies propose to remove the low quality workers after collecting all the data. To do so, they compute annotators' scores investigating the correlations between annotators' annotations (for example the MOS score [88] or CrowdTruth worker quality score [43]). When some expert annotations are available, techniques to compare the annotators' annotations to the expert ones and eliminate the outliers are also investigated. Considering that we cannot run a new crowdsourcing task, we focus on the second kind of annotators' filtering methods. We decide to study the CrowdTruth framework because 1) it does not require any expert annotation that we do not have, and since the Jigsaw dataset is large, annotating a sufficient amount of samples would be time-consuming, and 2) it does not aim at transforming the annotations into one unique label but at harnessing disagreement. The hypothesis we investigate is:

H3: ***the CrowdTruth framework enables to filter out the annotators' annotations of low-quality while keeping the annotations which correspond to valid judgements different from the majority.***

Experimental set-up

To verify the hypothesis, we apply the CrowdTruth framework to our dataset and investigate whether the CrowdTruth results make sense by comparing the framework's output scores to our manual evaluation of the annotations or annotators. The CrowdTruth framework takes as input a set of annotations with the annotators' identifiers on different samples, and outputs three scores: the Unit Quality Score (UQS), the Worker Quality Score (WQS), and the Annotation Quality Score (AQS) [44].

For one sample, the UQS is computed as the average cosine similarity between all worker vectors, weighted by the WQS and AQS. It represents the degree of agreement for each sample. The WQS is computed as the product of the worker-worker agreement (WWA) and the worker-media unit agreement (WUA), and assigns one score to each worker measuring the correctness of her annotations. The WWA is the average cosine distance between the annotations of a worker and all other workers that have worked on the same media units, weighted by the worker and annotation qualities. The metric gives an indication as to whether there are consistently like-minded workers. This is useful for identifying communities of thought. The WUA is the average cosine distance between the annotations of a worker and all annotations from the rest of the workers for the media unit, weighted by the media unit and annotation quality. It calculates how much a worker disagrees with the crowd on a media unit basis. The AQS is the weighted average with the WQS of the probability that when an annotator i selects label a for a sample, annotator j selects the same label. It represents the agreement for each label over all the annotations given in the crowdsourcing task.

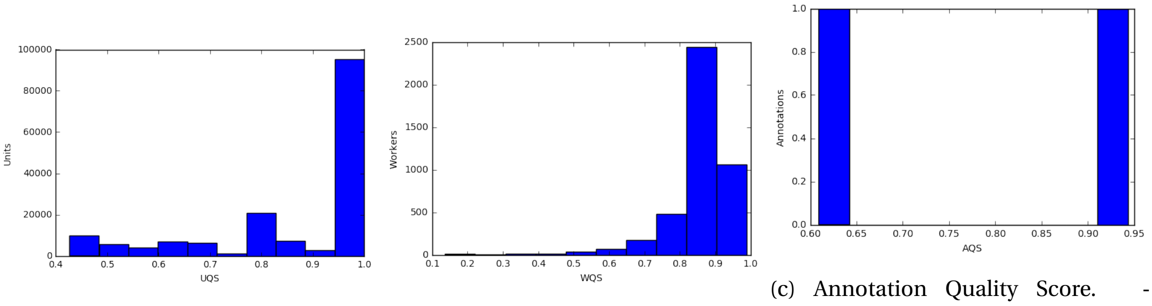
The Jigsaw dataset is constituted of samples and their annotations on a scale between -2 and 2. This scale is criticized in the crowdsourcing literature because it might be unclear to the annotators. Consequently, we apply the CrowdTruth framework to 4 different data set-ups with 4 different scales obtained with 4 different label aggregations. For the WQS, we sample from the dataset 30 annotators randomly in the annotators whose CrowdTruth scores on the binary labels are very low or very high, and we manually compute for each of them a quality score by averaging the number of their annotations which could be considered correct (according to our appreciation of the sample) over the total number of annotations they made. Then we compute the mean-squared error between the WQS that we computed and the CrowdTruth WQS for each set-up. The smaller the error, the more the CrowdTruth results should highlight the correct or incorrect annotations because the manually-computed score simply correspond to the fraction of correct annotations. To evaluate the UQS, we sample randomly 100 different sentences from the dataset, selecting sentences in the whole range of CrowdTruth results on the binary labels. For each of them, we give an ambiguity score (0: ambiguous, 1: non-ambiguous) depending on whether the sentence can be evaluated clearly or whether it is subject to multiple interpretations. Because we give binary labels to each sample but the results returned by the framework are continuous between 0 (low-quality unit) and 1 (high-quality unit), we compute the Area Under the Receiver Operating Characteristic Curve (AUROC) score so that the difference of score type is taken into account. The higher the score, the more the manual score and the CrowdTruth score are in agreement, the more the CrowdTruth framework is accurate at identifying more or less ambiguous sentences. We cannot give a score manually to evaluate each label and compare it to the AQS because it is hard to quantify how clear each label is. We only qualitatively compare the AQS results with our intuition of the labels. The mean-squared error and the ROCAUC score are computed on the same data samples for each of the set-ups.

Results

We compute the CrowdTruth scores on the 4 set-ups and report the resulting plots below.

- **Set-up 1**: 2 labels: -2 and -1 labels are considered as toxic and 0, 1, 2 as non-toxic. (Fig. 3.2)
- **Set-up 2**: 2 labels: -2, -1 and 0 labels are considered as toxic and 1, 2 as non-toxic. (Fig. 3.3)
- **Set-up 3**: 3 labels: -2, -1 labels are considered as toxic, 0 as neutral, and 1, 2 as non-toxic. (Fig. 3.4)
- **Set-up 4**: 5 labels: the -2 to 2 labels are considered separately. (Fig. 3.5)

We observe that for the first set-up (Fig. 3.2), most sentences and most annotators have a high-quality but that some have lower quality. This is probably because the labels to annotate are easily interpretable and

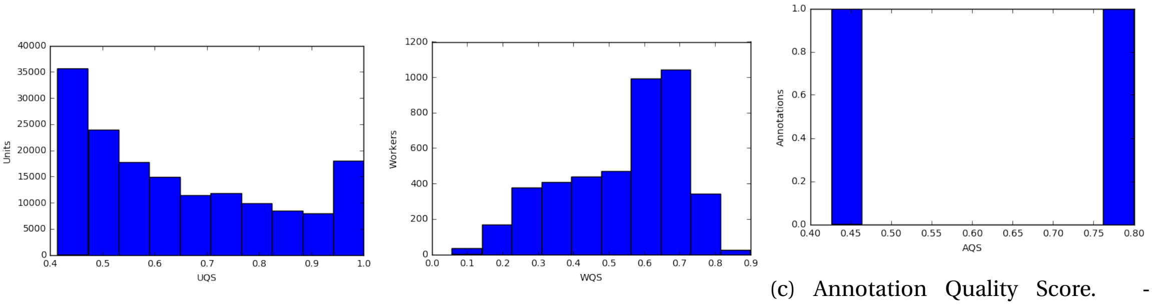


(a) Unit Quality Score

(b) Worker Quality Score

(c) Annotation Quality Score. -1(0.609), 0(0.944)

Figure 3.2: Set-up 1. Results of the CrowdTruth metrics with binary labels: (-2, -1) = toxic, (0, 1, 2) = non-toxic. Most annotators and samples are of high-quality with a long-tail of very low quality samples and annotators.

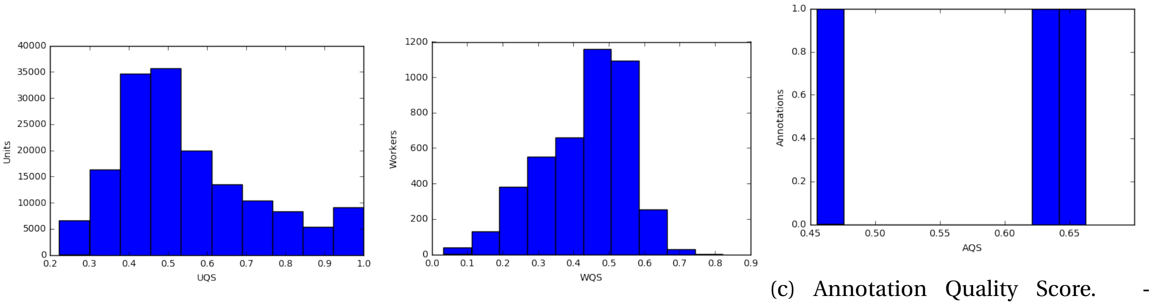


(a) Unit Quality Score

(b) Worker Quality Score

(c) Annotation Quality Score. -1(0.799), 0(0.427)

Figure 3.3: Set-up 2. Results of the CrowdTruth metrics with binary labels: (-2, -1, 0) = toxic, (1, 2) = non-toxic.

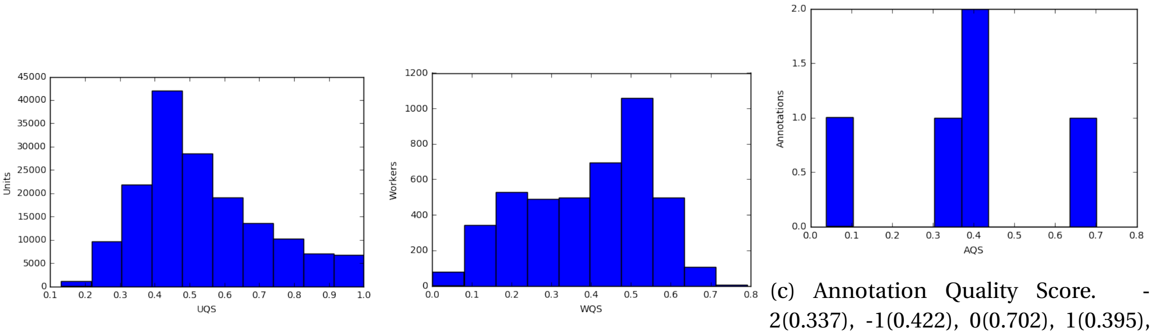


(a) Unit Quality Score

(b) Worker Quality Score

(c) Annotation Quality Score. -1(0.625), 0(0.663), 1(0.455)

Figure 3.4: Set-up 3. Results of the CrowdTruth metrics with three labels: (-2, -1) = toxic, 0 = neutral, (1, 2) = healthy.



(a) Unit Quality Score

(b) Worker Quality Score

(c) Annotation Quality Score. -2(0.337), -1(0.422), 0(0.702), 1(0.395), 2(0.039)

Figure 3.5: Set-up 4. Results of the CrowdTruth metrics with labels between (-2;2). The distributions of UQS and WQS are very different from the distributions for set-up 1, with lower average UQS and WQS.

thus there are less errors of annotations. For the other set-ups, the distributions of scores are concentrated on lower-quality scores, the distributions between set-ups 3 and 4 being of similar shapes, because there are more errors since the labels are more complex to understand and there are more different perceptions of the same labels for a same sample.

We report in the Appendix A.1.2 the Wikipedia comments with the lowest and highest UQS calculated for set-up 0 and set-up 4, and we manually study the annotations of the annotators with the lowest and highest WQS. We observe that the low-UQS samples with set-up 0 are sentences in foreign languages and sentences which seem ambiguous about their interpretation, while the high-UQS samples are usually long sentences which give constructive comments on the Wikipedia articles. With set-up 4, the low-UQS samples are more difficult to interpret, they do not seem to be more ambiguous than other samples, while the high-UQS samples are also not very clear compared to the samples returned for set-up 0. For set-up 0, we observe that low-WQS annotators seem to provide completely random annotations, while the high-WQS annotators have mostly valid annotations. For set-up 4 the low-WQS annotators behaviours is not always easily differentiable between spams or simply valid but different judgements.

For the study of the WQS, we compute the following mean-squared errors in the order of the set-ups: [0.0103, 0.1826, 0.2679, 0.3133]. For the UQS, we find the following AUROC scores in the order of the set-ups: [0.9452, 0.4792, 0.8607, 0.7906].

Discussion

The WQS and UQS returned for set-up 1 are much closer to our judgement of the workers annotations than for the other set-up since the error is much lower and the ROCAUC is the highest. Set-up 1 is the most intuitive to make sense of the possible annotation labels and that explains why the CrowdTruth results are closer to our judgement of the samples. This supports the idea of selecting its results to filter the annotators' annotations. The other set-ups are less close to what we expected for several reasons. The aggregation of the labels in set-up 2 is not meaningful since we differentiate between toxic and non-toxic samples and neutral is not usually confused with the toxic label. For the set-ups 3 and 4, there are too many different labels possible and the annotators cannot identify precisely which ones to use whereas having binary labels make the task clearer. When applying the CrowdTruth framework on these 3 or 5 labels, there appears to be a higher disagreement between workers, that we do not take into account since it is not related to the subjectivity of the property to annotate but only to the difficulty for a human to interpret the labels.

The AQS for set-up 1 exhibits high scores for both types of annotation, with non-toxic being the clearest. For set-up 2, the scores are a little lower, the toxic label having a higher score than the non-toxic label. For set-up 3, the scores are slightly decreasing again with the neutral label being the clearer. For set-up 4, the scores decrease again, the neutral label seems to be very clear to the annotators since the score is very high compared to the other scores, and the other labels have similar scores. The label very-toxic has very low score, which means that there is very few agreement over it. This trend in the scores among the different set-ups supports our previous explanations. The more possible labels there are, the less clear it is for the annotators and the more disagreement there is, thus the scores decrease. Consequently, we again conclude that the CrowdTruth results provide sound results over the Jigsaw dataset for our subjective property annotation task.

Although we do not compute the mean-squared error and the ROCAUC scores using all the annotations in the dataset, we assume that the results are representative of the whole dataset because our observations on the data themselves also correspond to the evolution of the scores returned by the experiments. The mean-squared error being very close to 0 and the ROCAUC score being close to 1 for the first set-up, we consider reasonable to use the CrowdTruth results to eliminate the low-quality annotators' annotations.

Conclusion

We conclude that ***hypothesis H3 is verified***, we can use the CrowdTruth framework with binary labels to eliminate the low-quality annotations, by eliminating the annotations of the annotators of low WQS (**RQ1.2**).

We assume that there are three kinds of annotators whose annotations differ from the majority: 1) spammers who randomly pick labels, 2) annotators which make infrequent mistakes, and 3) annotators whose interpretations of samples differ from the majority interpretation but still give "correct" labels. To obtain a dataset of high quality without eliminating the different judgements, we need to remove the first type of annotators while making sure not to remove the second and third types -or possibly only the wrong annotations for the second type of annotators. In order to select a threshold on the Worker Quality Score to filter out low quality annotators from the dataset, we manually check the annotators with the lowest scores and define a threshold from which the annotators seem to be spammers. We identify spammers by annotators with completely random-order wrong answers or annotators who always use the same label. The annotators with a quality score between 0 and 0.5 all seem to give annotations randomly. This can be seen by the fact that objective sentences receive wrong annotations and the annotations of the surrounding samples in the crowdsourcing task are similar (probably the annotators simply always use the same label). This represents around 50 of the annotators out of 4301 (1.16%) and 18000 annotations out of 1598289 (1.13%). Annotators with a WQS between 0.5 and 0.65 are also using random annotations but not for all the annotations, a majority of annotations is correct, so we make the choice not to remove these annotators. There are around 100 annotators (2.33%) in this category who annotated around 32000 annotations (2.00%). More precise treatment of each annotation of each annotator should be made later but this can not be realized with the CrowdTruth framework. Annotators with a higher quality score usually correspond to annotators who make very occasional mistakes, or have some judgements different from the majority, or annotators who do not make mistakes.

The CrowdTruth framework can be used to filter the spammers' annotations. However it is not possible to differentiate the annotators who make occasional mistakes from the annotators who express valid judgements but different from the majority, and it is not possible to filter out only the wrong annotations of these annotators who make occasional mistakes. These points remain to be investigated in future work.

3.4. Highlight of the presence of different subjectivities in a Computer Science toxicity dataset

We now verify hypothesis H1 of the chapter from a Computer Science point of view, by investigating whether the subjectivities highlighted in the psychology literature appear in the available Computer Science datasets of sentence toxicity. We answer the question: ***are there different valid judgements in available toxicity datasets?***, with the hypothesis that ***part of the samples are associated with different valid judgements***. We assume that different valid subjectivities are equal to different valid annotations on one same sample, what is called "disagreement" in the crowdsourcing literature, and we check for disagreement in the annotations.

3.4.1. Experimental set-up

Disagreement is measured by computing the Average Disagreement Rate (ADR) with the majority vote (MV) for each annotator. We define the ADR as the number of annotations of an annotator which differ with the MV labels divided by the total number of annotations of the annotator. It quantifies whether the annotators agree on judgements of data samples because the MV represents the common perception of toxicity on each sample (calculated as the most frequent annotation for each sample) ; and enables to verify whether all the annotators follow the same line of thoughts or whether they have different perceptions. We plot the distribution of ADR over the whole population to check for the disagreement repartition.

Because the wrong annotations of certain annotators could participate in the disagreement measures without being indications of subjectivities, we remove the wrong annotations with the CrowdTruth framework (Section 3.3.2), and then we study the disagreement in the dataset. Since it is difficult to distinguish between annotators which give wrong annotations and annotators whose perceptions differ from the MV, we repeat the process of removing the annotations of the low quality annotators, computing the ADR with the MV and plotting the ADR distribution, with the annotators of 0 to 0.6 CrowdTruth WQS removed so that we see whether removing spammers or valid but uncommon judgements has an influence on the disagreement.

3.4.2. Results

The histograms of ADR are computed using the ADR over binary labels (Fig. 3.6) and over the full range of labels (Fig. 3.7). We also plot the number of annotations removed and the average ADR over the whole dataset as a function of the minimum Worker Quality Score allowed in the dataset (Fig. 3.8).

For the binary labels, we observe that the more annotations are removed, the more the disagreement with the MV increases until it stabilizes and reaches a specific distribution with very few annotators always agreeing with the MV but most annotators only disagreeing around 15% of the time. The CrowdTruth WQS is computed as a combination of the sentence ambiguity and the disagreement rate of the annotator with the other annotators. It does not only take into account the disagreement with the majority and that is why the evolution between the ADR distribution and the number of low quality annotators removed is not linear. Since many samples are not ambiguous and incorrect annotators are incorrect only for a portion of their annotations, the more annotators (and their partially correct annotations) there are, the more agreement can be found, and decreasing the number of annotations make the annotators with slight disagreement appear. When decreasing the number of annotations, the annotators with very high disagreement are filtered out by

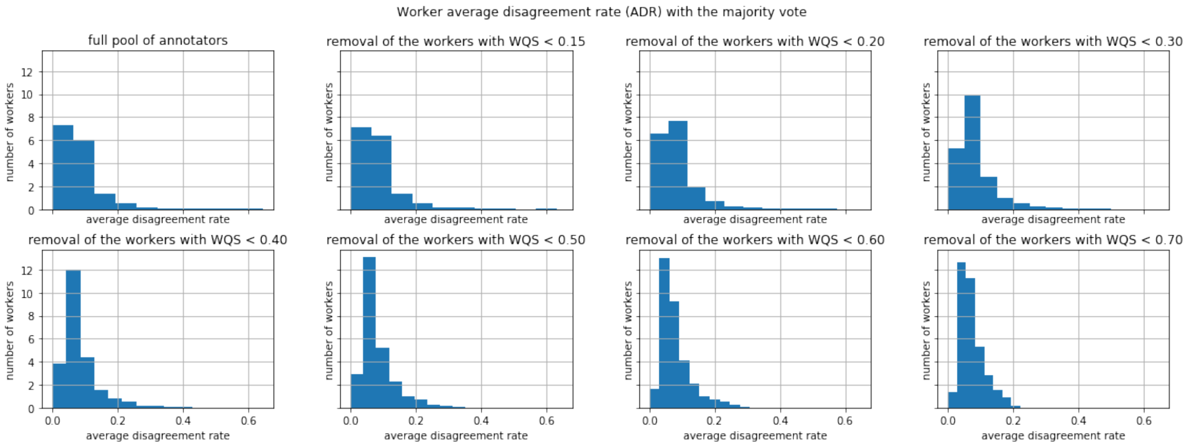


Figure 3.6: Worker Average Disagreement Rate with the majority after removing 0 to 300 lowest quality workers (Worker Quality Score between 0 and 0.7). The histograms are normalized, and the labels considered are binary. The more low quality annotators are removed, the more disagreement with the majority-vote is observed.

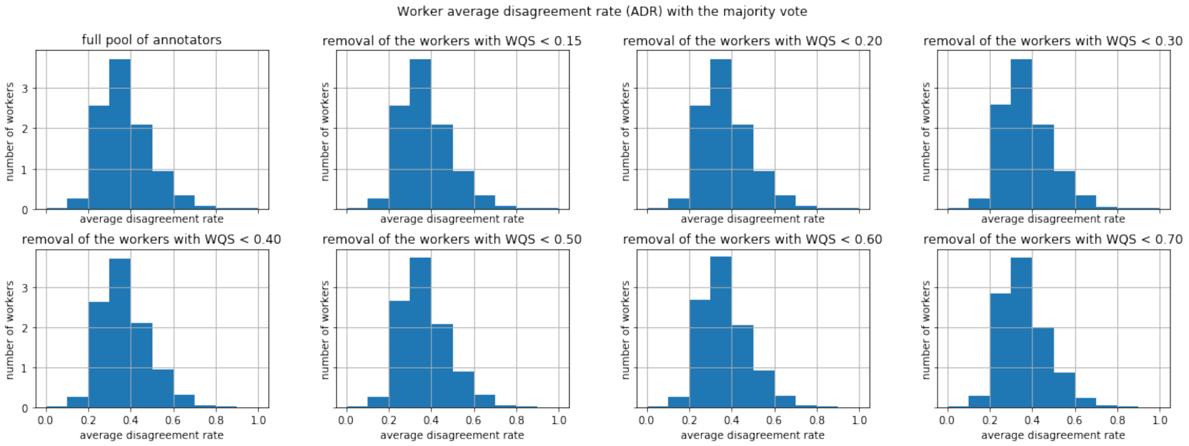


Figure 3.7: Worker Average Disagreement Rate with the majority after removing 0 to 300 lowest quality workers (Worker Quality Score between 0 and 0.7). The histograms are normalized, and the labels considered are within the $[\![-2;2]\!]$. Removing low-quality annotators almost does not influence the distribution of disagreement with the majority-vote.

the framework since they make many mistakes (it is not only that they always have a different opinion on sentences). This is supported by the plot of the CrowdTruth UQS (Fig. 3.2) which shows that most samples are non ambiguous and therefore people should not have many opinions which differ with the MV. The plot of the CrowdTruth framework results (Fig. 3.2) shows that the UQS is high and the WQS follows a distribution which is similar to our plot of the ADR with the MV, what is a good indication of the validity of the experiments.

For the full range of labels calculations, the histograms remain similar when removing any number of low quality annotators' annotations, and the average disagreement is much higher than for the binary labels. This is explained by the fact that the disagreement is always very high since it is difficult for workers to understand the difference between the labels such as toxic and very toxic, or healthy and neutral, and these differences are subjective, so there are several causes of differing perceptions.

3.4.3. Discussion

Since the disagreement has many causes (the labels are not well defined) when studying the full range of labels, we prefer studying the results on the binary labels which diminish the sources of the variations in the perceptions of toxicity. With the binary labels and approximately 50 low quality annotators removed, we obtained a stable ADR distribution with only a small percentage of workers who always agree with the MV. This proves that the Jigsaw dataset present different subjectivities since all the other annotators partly disagree with the MV and thus disagree among each other.

With the binary labels, only 10.5% of the annotators (around 400 annotators) always agree with the MV

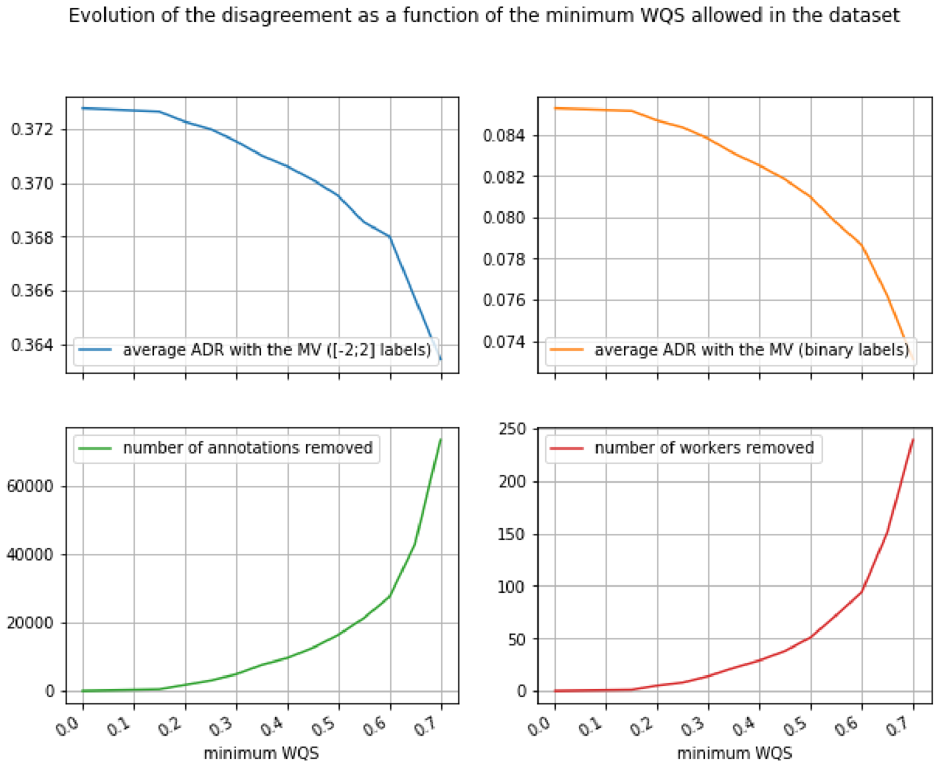


Figure 3.8: Average Average Disagreement Rate and number of annotators removed as a function of the minimum Worker Quality Score (WQS) allowed in the dataset. The average ADR computed on the available set of annotators with the binary labels is 5 times lower than with the 5 labels. The number of low-quality annotators removed as a function of the annotators' quality follows a similar curve as the number of annotations removed as a function of the quality of the annotators of these annotations, but with different steepening of the curves. The more low-quality annotators are removed, the lower is the average ADR of the available annotators.

for 50 or more low quality annotators removed. This shows that MV annotation aggregation is not representative of most individuals' line of thoughts but only of a sentence-level common opinion (the majority vote). Therefore current algorithms which are trained to output this MV are unfair towards most people since they are not in par with their line of thoughts. This supports our claim that the Jigsaw dataset is adapted to study the fairness of the prediction of subjective properties.

From this analysis, we additionally conclude that to study the subjective character of toxicity, it is important to remove at least the 50 lowest quality annotators (the spammers) and their annotations. Otherwise a majority of annotators seem to agree with the MV, what hides the unfairness of the true data. Besides, removing only 50 annotators among the lowest-quality annotators enables to keep the ones which disagree very often with the MV but express their true opinions and do not make many mistakes.

3.4.4. Conclusions

From the literature and the previous experiments, we conclude that sentence toxicity is a subjective property, and that the Jigsaw dataset enables to study the effects of the presence of different subjectivities on the fairness of datasets and Machine Learning models, with certain pre-requisites and limitations (**RQ1.1**). Thus ***hypothesis H1 is verified***: sentence toxicity is a valid use-case to study the prediction of subjective properties. Choosing a subjective property such as sentence toxicity, with an available Computer Science dataset which exhibits disagreement between annotators, are the main requirements to investigate the fairness of datasets and Machine Learning models made to predict subjective properties.

With the Jigsaw dataset, we identified that around 51% of the annotators (around 2500 annotators) disagree 15% of the time with the majority vote, 34% of the annotators (1445 annotators) disagree 20% of the time, and 4.5% of the annotators (192 annotators) disagree more than 20% of the time. A Machine Learning model which would be trained on the majority vote labels resulting from the aggregation of the annotations, assuming that its accuracy is very high, would consequently give predictions which would suit the line of thoughts of at most 10% of its users. That could be considered unfair towards the rest of the users, and that shows the necessity of further studying unfairness of the predictions of Machine Learning systems.

The low quality annotators' annotations should be removed from the dataset to make the different subjectivities more obvious. In the Jigsaw dataset, the disagreement among annotators is partly due to these subjectivities but also to the lack of context information about the sentences. In our work we are forced to confuse these two causes of disagreement and cannot distinguish them in the study of unfairness. However, future work could make sure to remove this second source of ambiguity and consequently of disagreement.

3.5. Clustering data for crowdsourcing cost-reduction

In order to create a dataset of sentences and judgements of perception of toxicity, which is large enough to train Deep Learning algorithms, we cannot hope to obtain crowdsourcing results of high quality on the full dataset because performing the crowdsourcing task enough times would be too expensive. The number of samples to annotate is too large to keep the cost of the crowdsourcing task low, and so we cannot have the full dataset annotated. That is why we attempt to answer the third sub-question: ***how to collect crowdsourcing annotations on a large dataset while maintaining a low cost?***. We hypothesize that:

> H4: ***automatically clustering data based on the sentence properties, annotating the cluster centroids and propagating the annotations to the rest of the cluster provide accurate labels for the dataset.***

This hypothesis is divided in three cases: 1) the clusters each reflect different toxicity judgements; 2) the clusters are more or less homogeneous on the toxicity judgements and only homogeneous clusters should be annotated; 3) the clusters reflect different types of sentences whose toxicity interpretation is more or less ambiguous and consequently more or less annotations per cluster or centroid should be collected.

3.5.1. Experimental set-up

To test the hypothesis, we run different clustering algorithms on our dataset, and investigate whether the clusters have a human-interpretable meaning different from only sentence properties, that is they reflect one of the three cited cases above.

Data preparation. First we clean the available data. We turn all the letters into lower case letters and remove all the formatting elements which could be in the sentences. We perform tokenization and remove all the English stop words. Then we encode the sentences. We experiment with two types of features. First we use the term frequency–inverse document frequency (TFIDF) computed over each sample. The second type of features is the Paragraph Vector representation of sentences. It was introduced by Le et. al. [69], with the idea that common feature representations of text lose the information about the order of words, while this information might be useful for further use in other algorithms. We train this representation with our whole corpus of sentences, using the python implementation Doc2Vec of the gensim library [3].

Training of the K-Means algorithms. We train the K-Means algorithm over the dataset, using between 1 and 30 clusters for the value of K. We perform different measurements (both intrinsic metrics and metrics using the binary and true labels of the samples) and plot these measurements as a function of K, in order to determine the optimal K (Fig. 3.9, 3.10). It seems that a K between 2 and 5 is sufficient.

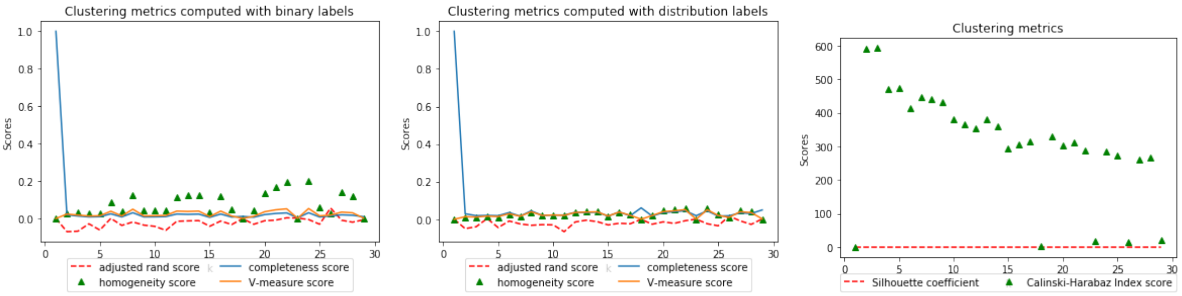


Figure 3.9: Search of the optimal K for the K-means algorithm. Computed for the TFIDF features. The completeness and Calinski-Harabaz indices show a clear change of the curve slope for a number of clusters K equal to respectively 2 and 3.

[3] https://radimrehurek.com/gensim/models/doc2vec.html

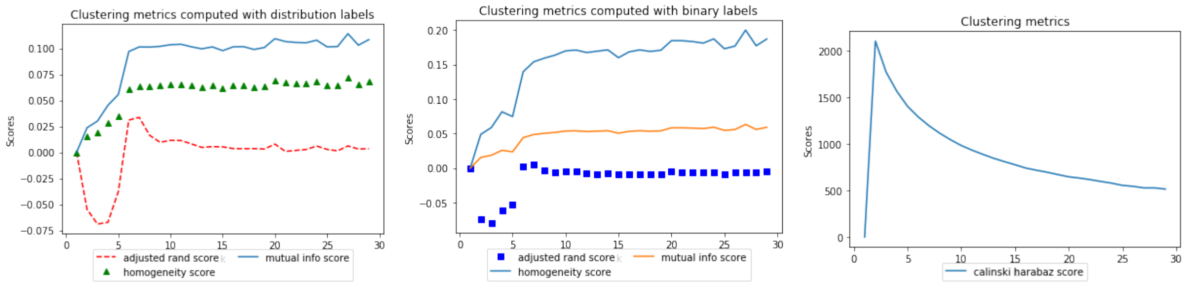


Figure 3.10: Search of the optimal K for the K-means algorithm. Computed for the Paragraph Vector features. All the scores exhibit a clear slope change for a number of clusters K equal to a value between 3 and 5.

Evaluation of the clustering algorithms. For each cluster, we collect separately for each demographic category the annotations corresponding to the sentences in the cluster, and aggregate the annotations into the majority-vote (binary label) per demographic category. Then we compute the mean and standard deviation over these aggregated labels. The mean enables to compare whether the labels are similar among different demographics inside a cluster. The closer to 0 or 1 the mean is, the more all the labels in the set are similar. The standard deviation enables to compare whether the labels among annotators of a demographic category in one cluster are similar. Consequently we can observe whether the clusters' annotations are homogeneous or not.

3.5.2. Results and discussion about the clustering algorithm

The results are plotted in similar plots as Fig. 3.11 for the TFIDF features. We can see that the values are consistent among the different demographic categories.

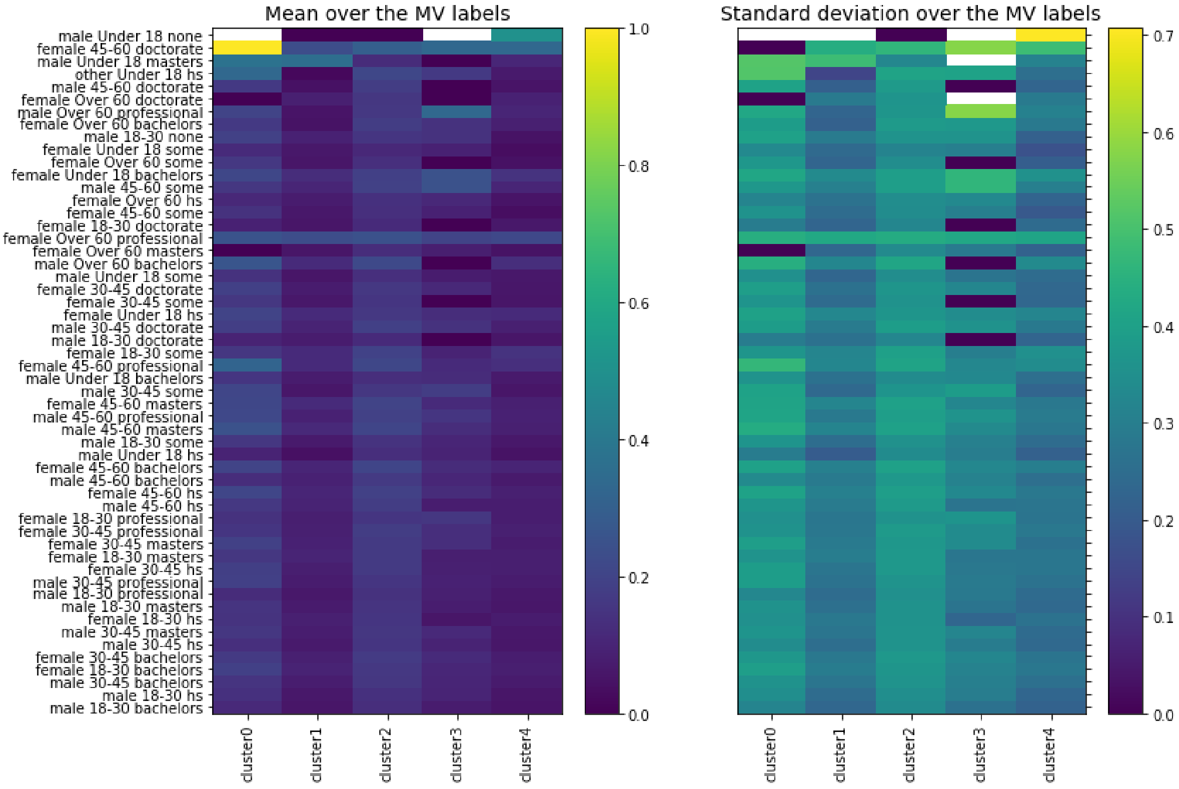


Figure 3.11: Results of the clustering algorithms with the TFIDF features. Clustering with K = 5
The categories of population are from bottom to top most to less frequent in the dataset. Most demographic category exhibit similar mean and standard deviation within a same cluster: a same number of annotations per category would have to be collected.
The values are different across clusters with more (clusters 1 and 4) or less (clusters 0 and 2) homogeneity in the annotations of each demographic category. Possibly more or less annotations should be collected for the samples of the different clusters.

For the TFIDF features, there is few difference between the results on the majority vote and on the density estimated labels. The clusters are quite homogeneous with a standard deviation around 0.3, each of the clusters having similar standard deviations. They do not enable to identify the clusters for which the labels would be all the same for one category of population since the most frequent demographics categories (with the most data samples at the bottom of the y-axis) have a standard deviation of approximately 0.2 to 0.45 which indicates that the two types of labels (positive and negative classes) exist in the clusters.

The Paragraph Vector features result in larger value differences in-between the clusters with a standard deviation between 0 and 0.5 for different clusters when using 5 clusters. When increasing the number of clusters, the differences between the clusters increase. The clusters with a standard deviation around 0.5 can be used to indicate the samples on which to collect many data. The clusters with smaller standard deviation would correspond to samples which all have the same label and therefore only few annotations on few of these samples could be asked. However, for these clusters the standard deviation remains still quite high (around 0.2) and therefore it might not be accurate to use only one label for these clusters.

3.5.3. Conclusions

In conclusion, we consider that ***the third case of the hypothesis H4 is verified***. A few clusters seem homogeneous over the label for specific demographic categories and thus few of the samples can be annotated in these clusters by these categories. A larger number of clusters have a higher standard deviation, in these ones each sample is annotated differently, and consequently they would require more annotations by a same demographic population.

Since there are many more non-homogeneous clusters, the clustering algorithm combined with the features that we use do not seem adapted for the selection of a few samples to annotate. (**RQ1.3**) However, future work could consist in investigating whether it is possible to set different minimum numbers of annotations to collect for different clusters by different demographic population, in order to obtain a set of annotations large enough to represent all the different possible valid judgements of the samples. Possibly we could order the clusters by standard deviation values, and study how the number of annotations per cluster type influence the performance of the trained models with these datasets. We project that there would be a trade-off between model's performance and number of annotations. This study could enable to define a threshold on the number of annotations per cluster type, or a relationship between this number and the performance of the models, in order to obtain performance as allowed by the trade-off.

3.6. Summary

In this section we investigated three aspects of the collection of annotations via crowdsourcing to create datasets of subjective properties used to train Machine Learning algorithms for the automatic classification of data samples on subjective properties.

First we verified that sentence toxicity is a valid use-case for the study of the creation of these datasets by reviewing the psychology literature about sentence toxicity and analysing the disagreement in an available Computer Science dataset of sentence toxicity. **(RQ1.1)**

Then we focused on how to collect high-quality annotations of subjective properties and identified the current limitations of crowdsourcing techniques. We gave a list of recommendations for the crowdsourcing task design, and pointed out that the use of questionnaires inspired from the psychology literature would merit being investigated to collect high-quality annotations. Furthermore we investigated whether the CrowdTruth framework is applicable to crowdsourcing tasks of subjective properties. We showed that although the framework is usable to eliminate spammers' annotations among the annotations of all the annotators, it does not enable to distinguish the incorrect annotations from the valid annotations which are different from the judgement of the majority on a sample. Thus we concluded that a future work's interest could be to propose new methods for the filtering of annotations of subjective properties. **(RQ1.2)** This constitutes contribution 3).

Finally, because the Machine Learning algorithms require many data to be trained on, we attempted to propose unsupervised clustering methods of data samples to reduce the amount of data to annotate and consequently to decrease the cost of dataset annotations' collection via crowdsourcing. **(RQ1.3)** However these propositions did not lead to satisfactory results and that is why we identify the following research direction for future work: how to collect crowdsourcing annotations of subjective properties on a large dataset while maintaining a low cost?

4

Method to evaluate algorithmic fairness for subjective properties classification

4.1. Introduction

In this chapter, we investigate the second research question RQ2: ***how to evaluate algorithms' fairness when predicting subjective properties?*** The aim is to find an objective and quantitative way to measure fairness of algorithms related to our task of predicting subjective properties of samples, which means predicting properties for which one unique judgement does not exist but different persons would have different interpretations of one unique sample. In order to do so, we divide the question into several sub-questions.

1. *How to define algorithmic fairness when predicting subjective properties?* **(RQ2.1)**
   → We review the literature about bias and fairness of Machine Learning algorithms and adapt previous definitions to the specific case of algorithms made to predict subjective properties. The proposed definition is based on equality of the algorithms' performance for each user.

2. *How to characterize possible unfairness of the algorithms when predicting subjective properties?* **(RQ2.2)**
   → We define a set of possible clustering criteria on which to divide the dataset and performance metrics to measure the performance of the algorithm on each subset of the dataset. After defining different algorithms with different expected fairness-related behaviours, we check whether the clustering criteria and performance metrics enable to exhibit the expected fairness-related behaviour of each algorithm. We propose to cluster the dataset on an annotation, annotator or sample level, and measure true positive and true negative rates.

3. *How to translate the fairness characterizations into a fairness measure?* **(RQ2.3)**
   → We propose metrics which summarize the previous characterizations, apply them to the different algorithms and verify whether they return significant results. We decide to compute the standard deviation of true positive and negative rates across data clusters.

After defining what algorithmic fairness is for the classification of subjective properties, we need to find ways to observe potential unfairness and to summarize the observations into values. For this, we propose several characterizations of unfairness (Section 4.3) and then several computations to sum them up (Section 4.4). At the end of the chapter, we answer the main question by selecting one of these computations as a method to evaluate algorithmic fairness. This is our second contribution of the thesis.

4.2. Definition of algorithmic fairness, necessity to propose new metrics

In this section, we answer the first question: ***how to define algorithmic fairness when predicting subjective properties?*** **(RQ2.1)** To define what fairness is for our task of predicting subjective properties, we investigate the literature about Fairness, Accountability and Transparency of ML algorithms, and adapt our findings to the task. Additionally we look at the Machine Learning literature to define which metrics are usually employed to evaluate algorithms, and investigate whether they can be used to measure algorithmic fairness.

4.2.1. Definition of algorithmic fairness for classification of subjective properties

Fairness in the literature

According to our literature review (Section 2.4), there are multiple definitions of fairness in the literature, but they all refer to possible discriminations between groups of people. For example a popular mathematical definition of fairness is the error rate parity, saying that for an algorithm to be fair the false positive rates should be equal across groups of people. These definitions are motivated by the fact that the algorithms directly or indirectly make use of protected properties characterizing the persons who are classified to classify them according to one criteria, such as classifying between people who will commit a crime or not or classifying between people to whom a bank should or not give a credit.

We do not aim at classifying people, but sentences depending on both the sentences' and people's properties - we classify sentences conditioned on people. The aim of our task and its impact are different. In the other papers the outcomes of the algorithms' outputs can be harmful to certain people and categories of people since people are defined by a set of properties which make them member of a category. Thus it is possible to talk about a potential discriminative power. In our task we might not only use sets of properties to distinguish between people but also user-specific identifiers and we do not aim at creating algorithms which generalize over categories of population but we aim at algorithms with different outputs for each different user. Thus there might be discrimination if we put the users into categories to evaluate the algorithms and we find inequalities, but these "discriminations" would not be harmful to the population categories directly, and this way of analysing the algorithms would not be justified since it does not make sense to categorize the users by their protected properties only.

Binns et al. [13] analyse the fairness of their algorithms to predict toxicity on the Jigsaw dataset. Although they use a traditional definition of fairness measuring the accuracy differences of the algorithm to predict the majority-vote for different categories of population (male or female), they note one limit of their approach. Taking the majority-vote as ground truth for each category of population they define is not representative of the opinion of each member of the population, and thus their measure of fairness for the categories of population is itself not fair towards each member of the category. This is another research which supports our idea not to define fairness as the discriminative impact of algorithms.

From these observations, we claim that we can not use the traditional definitions of algorithmic fairness to evaluate the fairness of our algorithms, but that we must propose a new definition which is directly related to the task of predicting subjective properties.

Fairness related to our task

A danger with current Web systems is the filter bubble because it represents a threat to democracy since the opinion of the minorities remains unheard [15]. That is why the objective of our work is to predict subjective properties accurately, which means that we want to predict how each user would judge each sentence's toxicity instead of predicting the majority vote only. An unfair algorithm would be an algorithm for which the prediction accuracy would not be equal for each user, but which would return only the opinions of the majority or of certain groups in the population.

We propose to orient our definition of algorithmic fairness towards the following direction: *fairness is when an algorithm returns accurate predictions on samples for each user, disregarding whether the user's opinions are part of the minority or majority.* This generalizes over previous definitions which focused on discrimination towards protected categories of population. The definition (**RQ2.1**) we choose is:

Definition: **fairness is when an algorithms' prediction performance are equal for each user.**

That means that each user sees his/her opinions taken into account. If an algorithm is 100% fair, it will predict accurately for each sample and each user the judgement (annotation) that the specific user would make on the sample. When mentioning prediction performance, we do not refer to structural performance such as the speed or the memory of the algorithms, but to performance related to the accuracy of the outputs compared to the ground truth.

To additionally investigate whether our algorithms are discriminative towards certain categories of population, we use a second definition of fairness related to the protected features. **The algorithm is discrimination-related fair if its prediction performance are equal for each category of population defined by every possible combination of protected features' values.**

4.2.2. On the necessity of creating an adapted fairness evaluation method

Here, we show that it is necessary to propose a new algorithmic fairness evaluation method. As pointed out in the literature review (Section 2.4), usual methods to evaluate algorithms might exhibit high performance but hide biases in the outputs. We verify that it is the case for our task, making the hypothesis that ***traditional methods to evaluate algorithms' efficiency performance do not account for algorithmic fairness.***

From the literature review about Machine Learning algorithms for toxicity prediction (Section 2.2), we constitute a list of metrics usually used to evaluate the algorithms: precision, recall, F1-score, accuracy, Area Under the Curve, Spearman correlation, and confusion matrix to distinguish between each class. These metrics serve to compute a performance score of the algorithms, which is considered as their evaluation. We verify whether these evaluations give indications about algorithmic fairness.

- The **accuracy** is the ratio of the number of correctly classified instances over the total number of tested instances.
- The **Area Under the Curve (AUC)** is the area under the Receiver Operator Characteristic (ROC) curve which is the plot of the true positive rate (TPR) against the false positive rate (FPR) at various prediction probability threshold settings. Contrary to the accuracy, it enables to take into account the possible imbalance of classes to evaluate the performances of the algorithm.
- The **Spearman correlation** measures the strength and direction of association between two ranked variables[1]. We compute it between our targeted labels and the predicted labels from the algorithms. The formula is the following: $\frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}}$, with x and y the two ordered variables we study.
- The **precision** is the ratio of true positive over the total number of instances predicted as positive.
- The **recall** is the ratio of true positive over the total number of positive instances.
- The **F1-score** combines both the precision and recall: $2 * \frac{precision * recall}{precision + recall}$.

Experimental set-up

We set-up four different Machine Learning models for which we expect to observe different fairness-related behaviours and apply the usual evaluation metrics to them. We study whether these behaviours are observable with these evaluation metrics. The four Machine Learning models on which to make the observations are summarized in Table 4.1 with the number of the models. There are 2 dimensions of the models which make their expected behaviours change: the classifier's architecture and the training data. The classifiers' architectures are the following:

- **Traditional ML:** The traditional Machine Learning algorithms for which the inputs are sentence samples and the outputs are toxicity labels.
- **Input-augmented ML:** The traditional algorithms whose inputs are augmented with demographic information about the readers of the sentences. The outputs are also toxicity labels. The demographics are encoded in two different ways (continuous or one-hot encodings, explained in Chapter 5).
- **User-specific ML:** Algorithms which have reader-specific entities: they distinguish between known users by learning user-specific parameters.

The training data are divided into three main categories:

- **MV data:** The dataset is constituted of samples and their majority-vote (MV).
- **Disaggregated data:** The dataset is constituted of samples and their annotations (identical samples appear several times in the dataset with different corresponding annotations).
- **User-specific data:** The dataset is constituted of samples, their annotations and their annotators' information (annotator identifier, and demographics informations).

We expect that the models made for distinguishing between annotators trained with adapted datasets (models 3 and 4) are fairer because a completely fair algorithm should return the individual annotations of each annotator instead of aggregated labels -what models 3 and 4 should be better at doing.

[1] https://statistics.laerd.com/statistical-guides/spearmans-rank-order-correlation-statistical-guide.php

ML / data	MV	disaggregated	user-specific
traditional ML	1	2	NA
input-augmented ML	NA	NA	3.1 (continuous representation)
input-augmented ML	NA	NA	3.2 (one-hot representation)
user-specific ML	NA	NA	4

Table 4.1: The different models on which to make the observations

Results, discussion and conclusions

We evaluate the models with the above cited evaluation metrics. The algorithm evaluated here is a simple classifier, the Logistic Regression, because it is fast to train and test. Other algorithms should lead to similar observations since their general behaviour is the same. Since we can make similar observations for each of these metrics, we only report in Table 4.2 the results concerning the accuracy and F1-score computed on test data where annotators' information are available.

model	F1-score	accuracy	accuracy positive class	accuracy negative class
(1)	0.6644	0.6885	0.6164	0.7607
(2)	0.6663	0.6897	0.6194	0.7600
(3.1)	0.6740	0.6899	0.6410	0.6899
(3.2)	0.6727	0.6892	0.6387	0.7398

Table 4.2: Efficiency performance of the models on the ambiguity balanced dataset. Along rows: the different training models, along the columns: the different evaluation metrics. The performance measures are similar for the four models, whereas they are expected to have different levels of fairness.

We observe that for each column, the accuracy and F1-score measures are similar whereas the models are expected to behave differently. For example, it is expected that configuration (3) should have higher accuracies than the others on the disaggregated dataset but the results do not show this. This has two explanations: the test dataset is unbalanced with few sentences which have different associated opinions so the use of a complex algorithm does not change the output performance ; or the performance over a certain type of samples decrease while the performance on another type increase, what hides potential changes in the behaviours of the models. From Chapter 3, we know that there are different subjectivities in the dataset and thus the first explanation does not hold, but the second does.

This is an indication that using traditional metrics is not meaningful to investigate algorithmic fairness for the prediction of the different subjectivities because these metrics average over the whole test dataset, whereas the samples with differing opinions do not constitute the whole dataset and so the performance differences over these samples are not clear in the evaluation. Therefore, our hypothesis is verified: usual evaluation methods of algorithm performance are not adapted to evaluate algorithmic fairness.

4.3. Characterization of algorithmic fairness

Usual evaluation methods do not enable to observe potential unfairness and understand it, so we focus on: ***how to characterize possible unfairness of the algorithms?*** **(RQ2.2)** We make several hypotheses to characterize algorithmic fairness. For that we list the behaviours that Machine Learning classifiers made or not to resolve unfairness should exhibit so that we have a list of properties that a fairness metric should transmit. We list possible test set splitting criteria, that we hypothesize should enable to observe the behaviours (characterization hypotheses). We apply these characterization hypotheses to our trained algorithms to evaluate whether they enable to exhibit the previously listed properties. The characterizations which are meaningful and human-interpretable are the foundations to propose evaluation metrics in the next section.

4.3.1. Formulation of fairness-characterization hypotheses

List of the expected behaviours of the Machine Learning models

Algorithmic fairness corresponds to performance equality on the user level. Accurate predictions correspond to the judgements of each user (the annotations of each annotator in our case). A 100% accurate model is also a 100% fair model but a 100% fair model does not imply a 100% accurate model because its prediction performance could be very low (low accuracy) but equal (high fairness) for each user. Considering a model fair and accurate when each annotation is correctly returned, algorithmic fairness can be studied from different angles. An unfair model returns incorrect annotations for certain users and consequently for certain samples, thus the annotation-level and sample-level study of the algorithm predictions might give indications of the potential causes of unfairness. We list the behaviours that we expect the four previously introduced Machine Learning models (Table 4.1) to exhibit, divided into four classes depending on what aspect of fairness is investigated (user-, sample-, annotation-, or discrimination- related fairness). For the interested reader, in Appendix B.1 can be found the detailed list of expected behaviours.

Models (1) and (2) are respectively trained on the aggregated labels by majority voting (MV) and on all the annotations, and both are not able to make any distinction between the different annotators of the data. Consequently, we expect them to output predictions which correspond to the opinion of the majority. That is, they would perform better for annotators who often agree with the MV, for annotations which correspond to the MV, and for samples whose annotations present high-agreement (most of the annotations are equal to the MV), compared to data corresponding to low consensus (with few agreement and consequently different from the MV). Thus, these two models are expected to be mostly unfair on the user, sample and annotation levels. The analysis of the dataset (Appendix A.2) showed that the quantities of data of low or high consensus are similar across all the demographic categories, and consequently these models which do not distinguish between the categories should be fair on a discrimination-level.

Models (3) and (4) are expected to output predictions which more often correspond to the exact annotations of each annotator instead of the MV. Consequently they should be fairer than models (1) and (2), but not 100% fair because there are not enough training data and not enough known features describing each annotator in order to learn the line of thoughts of each annotator. On the discrimination-level, model (3) is expected to be less fair than the others because it will learn distinctions between demographic categories with different accuracies depending on whether there are many or few training data corresponding to each category and consequently its performance will be unequal between categories. Model (4) should be fairer on this aspect since the performance of its predictions should not depend directly on the demographic categories.

Formulation of hypotheses

We propose possible ways to check whether these behaviours hold when evaluating the models. First we define several scores to compute on the data in the dataset, that we use in the chapter.

- *Annotators' average agreement rate with the majority vote (ADR)* score: general disagreement of one selected annotator with the majority vote, computed as the number of annotations of the annotator which differ from the MV labels divided by the total number of annotations of the annotator.

- *Sentence ambiguity (AS)* score: disagreement of the annotators on the label of the selected sentence, computed as the number of identical annotations divided by the total number of annotations for the sentence.

- *Annotation popularity (AP)* score: disagreement of one selected annotation with the other annotations of a sample, computed as the percentage of a sample's annotations which are in agreement with the annotation.

We formulate the following hypotheses.

H1: ***Comparing the performance of the model for each user shows whether the model is fair or not.***

Indeed, if the performance are different, the definition of algorithmic fairness is invalidated.

H2: ***Clustering the test data into several clusters and computing the efficiency performance for each cluster enable to account for different algorithmic fairness aspects.***

Comparing the performance of the model on an annotation, user or sample-level would not explain any possible cause of unfairness, whereas clustering annotations, users or samples according to an interpretable criterion and comparing the cluster performance would. We hypothesise that the following clustering criteria should enable to highlight potential causes of unfairness:

- ***User clustering criteria:***
    - The ***workers' average agreement rate with the majority vote (ADR)***: differences of performance between these clusters would help explain the unfairness. We expect a linear increase of performance for clusters with low to high disagreement workers.
    - The ***CrowdTruth Worker Quality Score (WQS)***: this score partly represents the disagreement between workers so we expect the same behaviours as for the previous clustering criteria.
    - The ***demographic categories***: should show whether the models are discriminative.
- ***Sample clustering criteria:***
    - The ***sentence ambiguity (AS)***: we expect low ambiguity sentences to receive higher performances since their annotations would be closer to the majority vote.
    - The sentence ***Unit Quality Score of CrowdTruth (UQS)***: this score is also made to represent sentence ambiguity thus we expect the same observations as for the above clustering criteria.
- ***Annotation clustering criteria:*** The ***annotation popularity (AP)***: we expect the annotations which are less popular (minority opinion) to have lower performance.

H3: ***Different performance metrics make the potential unfairness more or less observable***.

The traditional algorithm evaluation metrics usually exhibit different aspects of algorithmic performance and therefore they should also influence the observed behaviours about fairness.

4.3.2. Experimental set-up

We apply the proposed evaluations on models (1) to (3) and verify whether the expected behaviours are observed. We separate the dataset into clusters with the different clustering criteria, compute the usual evaluation metrics on these clusters, and check whether the results enable to make interpretations about fairness of the models. After selecting only the most meaningful characterizations (details in Appendix B), we focus on the analysis and exploitation of the retained characterizations with the accuracy and F1-score. We only study the differences between models (1) and (3) because models (1) and (2) have similar behaviours.

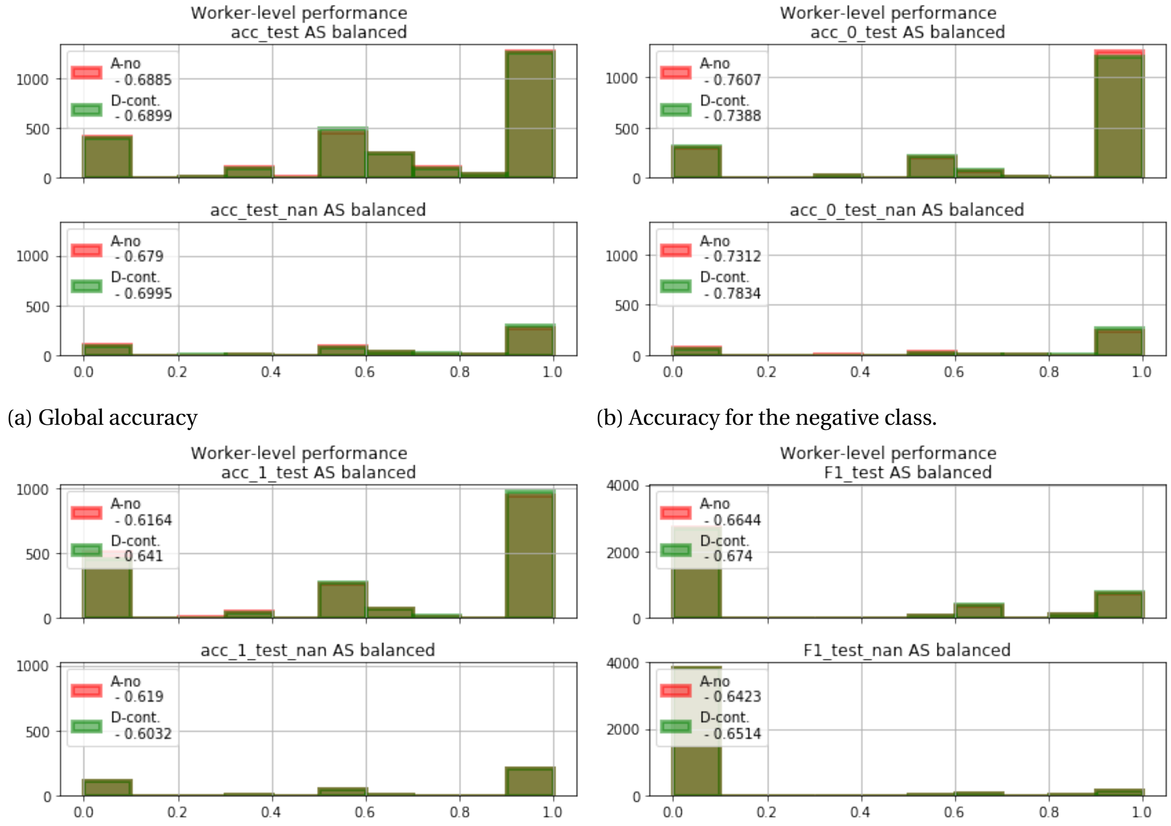
We plot on a same heatmap the performance of the different models to have an identical scale to compare them. To investigate the accuracy, we not only cluster the samples or users or annotations according to the previously defined criteria, but also according to the binary toxicity judgement that each user gave. This is equivalent to computing the true positive and true negative rates over the annotations inside the clusters. It enables to judge whether one label is more easily predicted than the other. It is not possible to do the same with the F1-score because it requires both classes to be computed.

4.3.3. Results and discussions

H1, H3: Fairness at the user level The distributions of the performance of models (1) and (3) for each user are reported in Fig. 4.1. We observe inequalities in the treatment of the different users with many users having very low or very high accuracy performance, the F1-score presents in majority low performance. This representation gives an indication about the fairness of the models: a fair model would be constituted of a unique-bar histogram while an unfair algorithm shows several bars of performance for different users. This characterization however does not enable to find out about the possible causes of unfairness.

The performance distribution over the samples is reported in Fig. B.2. The total accuracy distribution is almost linear with more sentences with a high accuracy prediction rate. Computing the distribution only on the positive or negative classes returns different types of results, mainly with similar number of sentences with 100% or 0% accuracy. This representation enables to compare whether the sentences receive equal performance, what is not the case here.However it does not give any indication to understand and check for the causes of the potential unfairness.

The user- and sample- performance distributions are highly dependent on the performance metric used and sensitive to the dataset employed to compute them. The distributions are affected by the test dataset constitution since datasets containing few or many hard to classify sentences would show very different distributions. Therefore, although the distributions show whether the algorithm is fair towards each user, this evaluation depends on the dataset and cannot be used to compare models evaluated on different datasets.



(a) Global accuracy

(b) Accuracy for the negative class.

(c) Accuracy for the positive class.

(d) Global F1-score.

Figure 4.1: Distribution of the performance of the predictions per user, evaluated with several performance metrics. Comparison of models (1) (in red named "A-no") and (3) (in green named "D-cont").
On each plot, several bars are present: different annotators receive predictions of different performance. Model (1) shows slightly higher bars corresponding to low-performance than model (3), but the comparison is not clear.

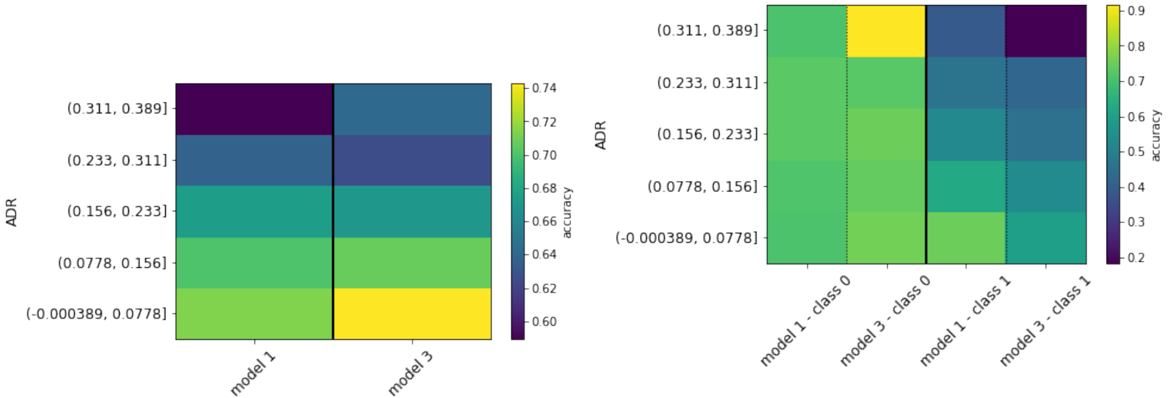
H2, H3: Performance on clusters A detailed description of the results is given in Appendix B.2.2. Clustering on the user level with the ADR score, the WQS or the demographic categories and computing the performance on the clusters enables to observe the expected behaviours of the models. The same procedure on the annotation-level leads to the same observations, the behaviours are observed more clearly when computing the accuracy separately on the positive and negative classes than with the average accuracy. The same procedure on the sample level (AS or UQS scores) also shows the expected behaviours, however the performance trend across clusters is not as linear as expected. The lowest consensus data show very low performance even with model (3), probably because there are not enough data to learn the exact annotations.

An example characterization is given in Fig. 4.2. The horizontal separations correspond to different clusters of data depending on the values of the ADR score (reported on the y-axis), and the vertical separations to the evaluations of different models on the positive and negative classes. Each cell correspond to the accuracy of one model on one cluster. For the accuracy over the two classes, it confirms that high consensus data (bottom cells) receive higher performance predictions than low consensus data (top cells), and that model (1) performs with lower performance than model (3) on medium-consensus data (middle cells). That enables to conclude about the relative fairness of the models: model (3) is fairer than model (1) since its performance are more equal across clusters. For the accuracy on the two classes separately, model (1) seems to perform better. The chosen characterization should depend on which performance metric to focus on.

4.3.4. Conclusions

We discuss whether the different hypotheses are verified.

H1. Plotting the distribution of the model performance for each user shows performance inequality between users and thus shows unfairness in the model. However, no clear difference between the models appears, and the distributions are dependent on the constitution of the test set. Thus ***hypothesis H1 is verified***



(a) Characterization using the accuracy.

(b) Characterization with the true positive and true negative rates (separate accuracy on the two classes).

Figure 4.2: Visualization of the accuracy, and of the true negative and positive rates (class 0: non-toxic, class 1: toxic) based on the ADR clustering criteria (intervals reported on the y-axis). Comparison of models (1) and (3).
Model (3) performs better than model (1) on data of very low consensus (0.13 to 0.39 ADR) and of high-consensus (0 to 0.23 ADR) according to the accuracy values. These observations are inversed for the true positive rate.
The performance metric used changes the observations: it should be chosen depending on the aim of the model (optimization of the accuracy or of the true positive rate).

but it does not seem to be a convenient characterization to create a fairness evaluation metric.

H2. The different clustering criteria proposed on the different levels enable to highlight different causes or explanations of unfairness: ***hypothesis H2 is verified***. We select four criteria whose interpretations do not overlap: on the user-level 1) the workers' average disagreement rate with the majority vote, 2) and the demographics categories ; on the sample-level 3) the sentence ambiguity ; on the annotation-level 4) the annotation popularity. We select these criteria over the CrowdTruth UQS and WQS because the resulting characterizations are similar but the selected criteria are faster to compute than the CrowdTruth scores.

H3. ***Hypothesis H3 is verified*** since different metrics gave different interpretations of the models (**RQ2.2**). We decide to work on the accuracy metric since it is the easiest metric to interpret and on the F1-score, so that we can compare the results of the two metrics. In order to get meaningful information about the accuracy in case the test dataset is unbalanced over classes, we decide to compute the accuracy separately on the positive and negative classes (true positive and true negative rates).

We conclude that algorithmic fairness can be characterized on several levels which explain different causes of unfairness. To highlight the fair or unfair character of a model, the annotations of the evaluation set can be clustered according to criteria depending on the level on which to focus, and the performance of the model on each cluster should be compared to identify potential inequalities.

4.4. Metrics for fairness evaluation

We now answer **(RQ2.3)**: ***how to translate the fairness characterizations into a fairness measure?*** The characterizations show different aspects of unfairness of the predictions of the models, thus we propose different metrics based on them, experiment on the different parameters of the metrics and evaluate their significance on our models. We conclude on a final choice of metrics to evaluate fairness.

4.4.1. Formulation of hypotheses

Requirements for a new fairness metric

The algorithmic fairness metric should return a score computed for each algorithm to be evaluated. We list the properties that the fairness value should respect:

1. The metric gives an ***indication about the fairness*** of the Machine Learning models.
2. The metric gives an ***indication about the causes*** of potential unfairness.
3. The metric also gives an ***indication about the global performance*** of the algorithms. If not, an algorithm with low but equal performance for each user would be seen as highly fair while the algorithm is fair but is also completely inefficient.

4. The metric is ***independent of the dataset*** on which it is evaluated. In this way, algorithms evaluated on different datasets would still be comparable.

Proposition of different metrics

Based on these requirements and the previously proposed characterizations, we formulate hypotheses on the possible way to evaluate algorithmic fairness.

Requirement 1. Previous section showed that the metric can not be an average over the whole dataset since the dataset constitution biases the computation, what is against requirement 4. For example a model evaluated on an unbalanced dataset with many high-agreement samples would exhibit a high accuracy since it is able to return the majority-vote, while a dataset with a majority of low-agreement samples would lead to low accuracy. The previously proposed characterizations mainly consist in dividing the dataset into clusters, computing the average algorithm performance per cluster, and comparing the algorithmic performance between clusters. We can *quantify the dispersion between the values of each cluster to account for the comparison.* If the clusters' properties are identical across datasets, the performance values should be comparable. Considering that our definition of fairness is equality of performance across users, we hypothesize that:

> H1: Main fairness indicator: ***quantifying the dispersion between the performance of the clusters with the user-level clustering over the average disagreement rate with the majority vote serves as the main measure of fairness (requirement 1)***.

Requirement 2. The values obtained with the other clustering criteria of the selected characterizations would be secondary measures to explain potential unfairness.

> H2: Side aspects of unfairness: we could interpret unfairness based on inequality of certain ***properties of the users (demographics for discrimination-related fairness), samples (ambiguity of the sentence) or annotations (popularity of the annotations)***.

To measure the gap between the different clusters, we propose to

> H3: Cluster dispersion: ***compute the standard deviation across the performance values associated to each cluster***, or to ***compute the range (absolute value) between the highest and lowest performing clusters (requirement 2)***

Requirement 3. We need to give one measurement of the performance of the model. However, like the previous considerations, we can not compute the accuracy performance over the whole dataset since it is not independent of the test set (against requirement 4). We propose to report the:

> H4: General performance: ***average performance across the performance of the different clusters***, or the ***lowest performance among all the clusters' performance (requirement 3)***.

For all these hypotheses, the performance metric on which the computations are based is the:

> H5: Performance metric: ***F1-score or the average between the true positive and true negative rates dispersions and general performance***.

Requirement 4. Different datasets comport different ranges of values of the clustering criteria and so the constitution of clusters is different across datasets. Small datasets would not enable to compute the performance over the whole range of clusters and the fairness values would be biased towards the properties of the available clusters (low performance if the clusters corresponding to easily-predicted high-agreement data are missing, or conversely high performance if low-agreement clusters are missing). To counter this, the datasets to evaluate algorithmic fairness should fulfil minimum requirements such as a minimum number of data to form each cluster, or:

> H6: Dataset: ***The computations could be spanned over different ranges of clusters.***

Instead of computing the metrics over all the clusters spanning over the whole range of values the clustering criteria can take, the metrics could be computed with a parameter which represents thes extent of the clustering criteria range used. The range starts with clustering criteria values which correspond to data with high-agreement between annotators (for example low ADR correspond to high agreement of the workers with the majority vote), and ends with clusters of lower-agreement data (high ADR reflects low agreement of the worker's annotations with the other annotations). In this way the evaluation can be made on any dataset with the specification of the parameter.

4.4.2. Experimental set-up, results and discussions for the experiments on the variables of the metrics

Experimental set-up

To investigate whether the hypotheses are valid, we apply them to models (1) to (3) and check whether the value trends correspond to the expected behaviours on these different models. The impact of the different variables which intervene in the computation of the metrics (clustering criteria, evaluation metric, number of clusters to make the computations and range of clustering criteria based on disagreement of the data) is analysed by making these variables vary and comparing the returned values to the expected behaviours of the models. The range of clustering criteria values spans between 0 and 1, and the number of clusters between 1 and 13. The evaluation metrics used for the computation are the F1-score, the accuracy, the accuracy of the positive class or the average of the true positive and negative rates.

Results

Popularity-related fairness. The results of the experiments are plotted in Fig. 4.3. For the other clustering criteria, the figures are not put in the report for space considerations and because the results are similar to the previous ones. For the standard deviation based values, as expected only using one cluster gives a score of 0 on fairness (maximum fairness). Increasing the number of clusters does not make the score vary significantly. Considering the clustering criteria range, the smaller the range is, the less disagreement in the data remains, the more the final score varies for the F1-score, but the less it varies for the other evaluation metrics. For the accuracy-based metrics, the smaller the range is, the lower the value is because there are less variations across clusters since the more predictions are correct in total (the models are better at making predictions for high-agreement data). The smaller the range is, the more unbalanced are the data, so the F1-score varies in reversed direction to the other metrics. Compared to the standard deviation, the absolute value measurements exhibit a larger range of values. The other observations are identical to the standard deviation observations.

We select a number of clusters of 10 because this value presents few variations when varying the clustering criteria range. We consider that the default choice of clustering criteria range should be the full range, and if the test set does not enable it, it should be changed to a smaller range. Using the average of the clusters' performance, we observe that decreasing the clustering criteria range increases the average performance since only high-agreement data are considered. Using only one cluster represents the average accuracy over the entire dataset. Varying the number of clusters does not have a large influence because the average is in the end computed over the same data only with different weights. Similarly, using the clusters' lowest value we observe that the smaller the cluster range is chosen, the higher the minimum performance value is. This is because the models perform better on high-agreement data. The number of clusters used has a larger influence on the resulting values than for the average performance value: the larger the number of clusters, the lower is the minimum value because there are more chances that some clusters have lower performance (especially for clusters where there is a high disagreement on the annotations). Therefore, it is better to use the average of the clusters' performance since it is less sensitive to one of the parameters of the metric, and thus it makes it easier to compare across clusters.

Other clustering criteria. Concerning the experiments using the Average-Disagreement Rate with the majority vote, the bound of the clustering criteria range which varies is switched to the upper bound since the upper bound correspond to data of higher disagreement. The observations are the same as the ones for the popularity clustering criteria, and we draw the same conclusions. Concerning the experiments with the ambiguity score, the observations are similar to the previous cases. For the experiments with the Worker Quality Score and Unit Quality Score, we make the same observations as previously. However we note that making the clustering criteria range vary does not influence the performance value much, probably because the differences among the low ranges of the WQS are not significant enough to make a difference in the performance values. For this reason we decide not to base our fairness computation on this clustering criteria.

The results of the study of the fairness over the demographic categories are plotted in Fig. B.5. We observe that the more demographic categories are removed, the fairer the algorithms are until they are totally accurate when there is only one demographic category. Therefore, the proposed metrics seem to be an appropriate way to measure fairness related to discrimination.

Discussion on the metrics parameters

H1, H2: Choice of the clustering criteria. The AS-based computation exhibits whether there are differences between sentences subject to multiple interpretations and clear sentences. The ADR-based computation exhibits differences between the annotators who usually have judgements which differ from the majority and the annotators who always follow the majority. The AP-based computation represents an annotation-level point of view on fairness showing possible performance differences between judgements which belong to the minority and judgements followed by the majority, it gives a direct indication on whether the minority is represented in the output predictions or is ignored for the majority. Therefore, these three computations represent three different aspects of fairness, and enable to highlight three potential causes of unfairness since they all exhibit the expected fairness-related behaviours. That is why we decide to retain the three criteria and to present the three of them when evaluating fairness. Although discrimination-related fairness is not our targeted fairness, the metric proposed seems appropriate and does not overlap with the other three metrics. Therefore we choose to use it in the rest of the thesis to check for potential discriminations.

H3, H4, H5: Choice of the computation method. We select the standard deviation value over the absolute value because the behaviours observed on these two metrics are similar, but the standard deviation takes into each of the cluster values and thus the outlier clusters' influence is decreased. The absolute value only computes the difference between the highest and lowest performance values, so if one cluster exhibits unexpected values (for example if there are not enough data to compute a significant value), this value will have a large influence on the resulting fairness value. We choose the cluster-average performance (mean value) as the indication on the performance of the algorithms because it is less sensitive to the metric variables and because the combination of the average and the standard deviation enables to get a good approximation of the range of performance the algorithms have on the dataset, so there is no loss of information. The behaviours observed for the 4 different metrics are similar. We choose the computation of accuracies over the positive and negative classes as the metric on which to make the computations because it enables to take into account the classification on the two classes, and therefore to highlight possible gaps in performances in case the training dataset is unbalanced over classes. The F1-score could also be used for this purpose, however it is not as easily interpretable as the accuracy.

H6: Choice of the metric parameters. From the results of the experiments, we observe that there is a threshold on the range of the clustering criteria value where the fairness value changes much. The high-agreement clusters (clusters under the threshold) have their corresponding prediction performance higher with model (3) than with model (1) - model (3) is adapted to distinguish between different annotators-. The low-agreement clusters (clusters over the threshold) do not exhibit a large performance difference between the two models (sometimes even model (1) performs better than model (3)) because the algorithms might not have enough data to learn accurate predictions over very high-disagreement annotations, or the accuracy values computed for the high-disagreement data are not significant (lack of data to make the calculations). For the thesis, we propose to set a threshold to differentiate between these two behaviours for each clustering criteria and keep two fairness values -one before the threshold and one after- so that the two behaviours are exhibited. We list in Table 4.3 the two minimal clustering criteria range values. We choose a number of clusters of 10 because the returned values around this number of clusters are stable and the metrics are not very sensitive to this parameter.

binning criterion / range	min 0	min 1
AP	0	0.4
ADR	0.39	0.30
AS	0.5	0.7
demographic	0	100

Table 4.3: Threshold values for the different clustering criteria.

4.4.3. Experimental set-up, results and discussion on the metrics' significance

From our previous conclusions we selected a subset of the proposed metrics as a final fairness metric. We apply significance tests on the selected metrics between the clusters to check whether the differences in performance between each cluster are significant. We assume that if the value differences between the clusters are significant, it means that it is appropriate to create a measure of this dispersion to measure fairness. The experimental set-up and results are given in Appendix B.4.

The differences found between the clusters' performance are not always significant. This does not mean that the metric is not valid because it might be that the models simply do not perform differently on these specific clusters, which are usually consecutive clusters what justifies that there associated performance are similar. We can not reject one of the hypotheses we made previously based on these results.

4.4.4. Analysis of the extreme cases

We investigate the extreme cases to check for the validity of the metrics. For both the standard deviation or the absolute value range, the fairness value ranges between 0 and 1. A fair model presents a value of 0 because the performance of each cluster is equal. The less fair the model is, the more the value increases, the maximal unfairness value being equal to 1. The absolute value of the difference and the standard deviation remain under 1 in any case and are equal to 1 when the performance of the clusters are spanning over the whole range of possible values (for example in a case where there are two clusters, one with accuracy of 0 and one with accuracy of 1). Since it might be confusing that highest fairness corresponds to 0, we change the computation by adding 1 to the opposite of the previous calculation, so that the highest fairness score corresponds to 1 and the lowest score to 0.

We cannot use the global accuracy because if one class is better predicted than the other, although the average accuracy might show a good prediction quality, the accuracies of the two classes are very different. The global accuracy would hide potential low performance. If the standard deviation is computed over both the true positive and true negative rate values, the metric might not be a valid indication of fairness. If accuracies across the clusters of the positive class are equal, same for the negative class, but the values for positive and negative classes are different, the model is totally fair but the computation will return a low fairness value. Thus, we need to distinguish the fairness of the two classes. Possible ways to combine the two could be to take the average of the two scores or the minimum of the two. In order to have consistent indicators between the dispersion indication and the general performance indication, we choose the same combination method for these two scores. With the minimum the general performance and dispersion could be highly underestimated, thus we choose the combination by simple averaging of the fairness of each class. Simple average without class weights enables to obtain a value independent from the dataset used to evaluate the algorithms (no preference is given to one of the two classes).

4.4.5. Conclusion

H1, H2: Considering the previous observations, our final fairness metric (**RQ2.3**) is divided into 4 aspects:

- User average disagreement rate with the majority vote -based computation (ADR). It gives direct indication about fairness since it divides the dataset on a user level and our definition of fairness focuses on performance equality across users.

- Annotation popularity -based computation (AP). It informs on whether the potential unfairness is related to more or less accurate predictions on minority or majority opinions.

- Sample ambiguity -based computation (AS). It informs on whether the potential unfairness is related to more or less accurate predictions on samples with high or low agreement.

- Demographic -based computation. It informs on whether potential unfairness are related to discriminative behaviours between categories of population.

H3, H4, H5: Each of these aspects consists in two values computed using the true positive and negative rates of each cluster: standard deviation (dispersion metric) and cluster-average (general performance metric).

H6: In this thesis we use 10 clusters for each metric and decide to report the metrics computed on two different clustering criteria ranges (full range of data, smaller subset of higher agreement), but we advise to report measures on the whole range for more completeness later.

Mathematical formalization of the fairness evaluation method

We formalize the computation of the four fairness aspects in a mathematical way.

Definition 4.1. Fairness metric
$\forall i \in [\![1; n_G]\!]$, we note G_{i0} and G_{i1} each cluster of negative and positive class data in the partition of the dataset into (two times) n_G clusters. Each G_{i0} and G_{i1} consists in $n_{G_{i0}}$ and $n_{G_{i1}}$ samples $s_{i0,j}$ and $s_{i1,j}$ with $j \in [\![1; n_{G_{i0}}]\!]$ or $j \in [\![1; n_{G_{i1}}]\!]$. For each G_{i0} and G_{i1} is computed an average score S_{i0} and S_{i1} such as $S_{i1} = \frac{\sum_{j=1}^{n_{G_{i1}}} s_{i,j}}{n_{G_{i1}}}$ and same for $i0$. The average score in our case is the average accuracy of the model's predictions evaluated in the cluster on its corresponding level (user, annotation or sample level) since the $s_{i,j}$ are accuracies. The fairness indication is defined for each class as $F_0^* = \sqrt{\frac{\sum_{i=1}^{n_G}(S_{i0}-\bar{S_0})^2}{n_G-1}}$ with $\bar{S_0} = \frac{\sum_{i=1}^{n_G} S_{i0}}{n_G}$, and same for the other class i1. F_0^* corresponds to the sample standard deviation of the average score of each negative cluster, chosen because it is a classical measure of statistical dispersion of a distribution. The general performance indication is given by $S_0^* = \bar{S_0}$ for the negative class and same for the positive class. Then, the global fairness indication is computed as $F^* = \frac{F_0^*+F_1^*}{2}$, and the general performance indication as $S^* = \frac{\bar{S_0}+\bar{S_1}}{2}$.

Depending on the clustering criteria considered, the scores $s_{i,j}$ have different interpretations.

- **Demographics. Average Disagreement Rate with the Majority-Vote.** The G_i are clusters of annotators with their annotations. The samples $s_{i0,j}$ (and respectively $s_{i1,j}$) represent the average accuracy of the predictions of an algorithm for one annotator for the negative (respectively positive) class.
- **Annotation Popularity Score.** The G_i are clusters of annotations based on their ground truth label (two classes) and on their popularity score (separated into 10 equal length ranges) (20 clusters in total). The samples $s_{i,j}$ are binary labels, 1 if the prediction of the algorithm is equal to the ground truth annotation, 0 otherwise. Consequently, S_i represents the average accuracy of the algorithm predictions for the annotations of cluster G_i, for each class.
- **Ambiguity Score.** The G_i are clusters of sentences and their associated annotations. The samples $s_{i0,j}$ (and respectively $s_{i1,j}$) represent the average accuracy of the predictions of an algorithm for one sentence for the negative (respectively positive) class.

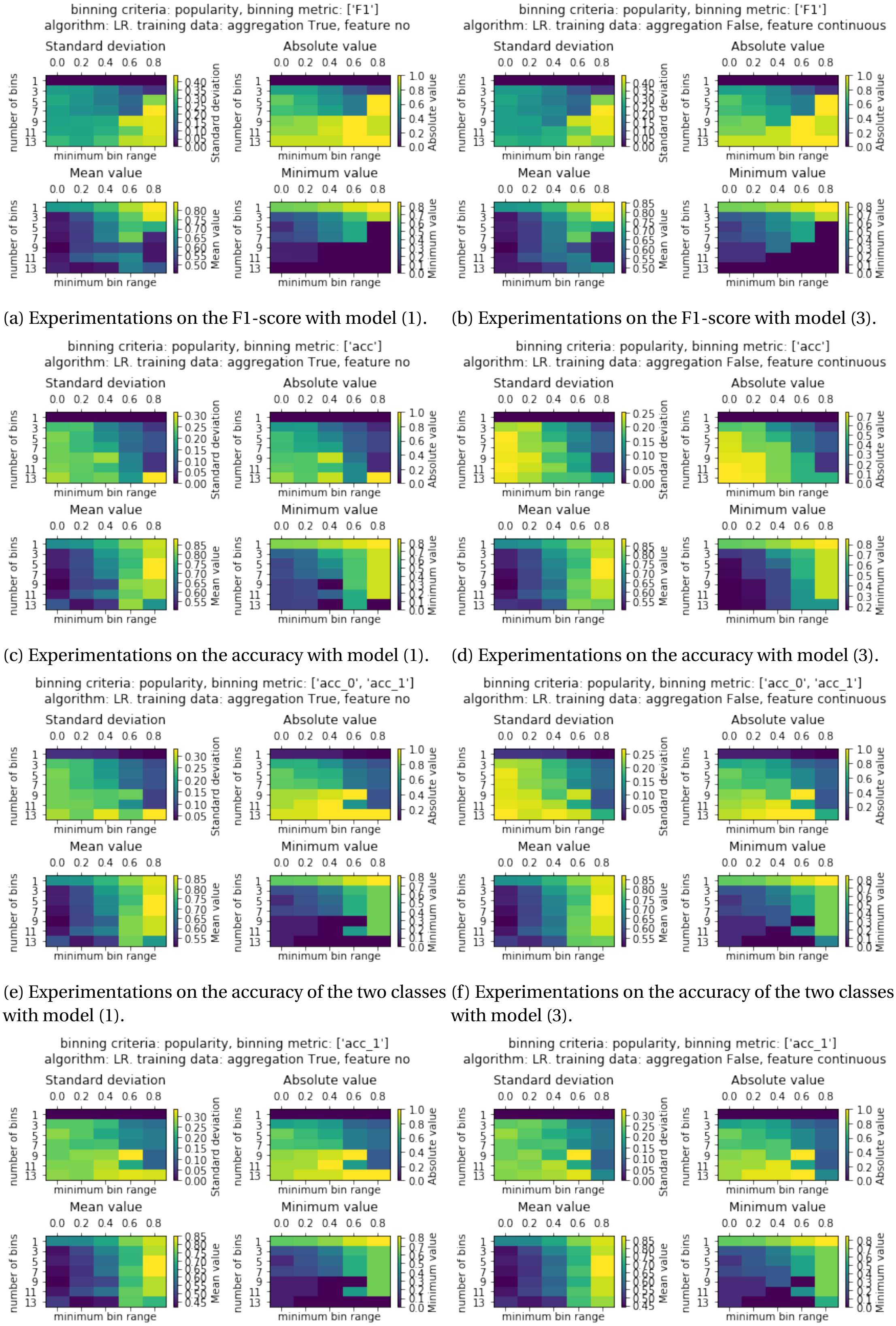
N.B: Here, each S_i follows a Beta distribution $Beta(\alpha_i, \beta_i)$. The Beta distribution is between $[\![0;1]\!]$ like the true positive and negative rates. It is the conjugate prior of a random variable defined as the number of success in n_{G_i} Bernoulli trials, here the trials are the different predictions of the model on the different annotations. The S_i are independent but not i.i.d since each distribution's parameters are different (the scores might vary a lot between clusters when the model is unfair). Thus, F^* is not a real standard deviation of one unique distribution, and cannot be computed analytically.

Application to the different models

In Fig. B.7 and B.8, we give the fairness performance of models (1) to (3). As expected, from the popularity score, ADR and ambiguity score -points of view over fairness, model (3) shows an improvement over models (1) and (2). This suggests that the metrics we propose are valid since they show the expected trends.

4.5. Summary

We focused on defining a new evaluation method of algorithmic fairness. We first proposed a definition of algorithmic fairness adapted to the task of classifying samples on subjective properties: an algorithm is fair when its prediction performance are equal for each user. **(RQ2.1)** Afterwards we investigated how to characterize potential unfairness of the algorithms by proposing different ways to cluster the dataset and measure the algorithm's performance on each cluster. **(RQ2.2)** Finally, we proposed different ways to quantify fairness by summarizing the observations enabled by the different characterizations, and investigated the validity of these different measures. We concluded that clustering the dataset on a user-level, sample- and annotation-levels, computing the true positive and true negative rates of each cluster, and measuring the standard deviation and average deviation across clusters give a valid measure of algorithmic fairness and of the general performance of the algorithm. This process is the evaluation method we propose for algorithmic fairness of models working on subjective properties ; the main characteristics of the resulting metrics being that they are almost independent of the test set, and that they inform on different causes of unfairness. **(RQ2.3)**



(a) Experimentations on the F1-score with model (1).

(b) Experimentations on the F1-score with model (3).

(c) Experimentations on the accuracy with model (1).

(d) Experimentations on the accuracy with model (3).

(e) Experimentations on the accuracy of the two classes with model (1).

(f) Experimentations on the accuracy of the two classes with model (3).

(g) Experimentations on the accuracy of the positive class with model (1).

(h) Experimentations on the accuracy of the positive class with model (3).

Figure 4.3: Experimentations on the popularity-based fairness computed with different evaluation metrics (F1-score, global accuracy ('acc'), accuracy of the positive class ('acc_1') or the average of the true positive and negative rates (['acc_0', 'acc_1'])). The y-axis represents the number of clusters on which is computed the metric, and the x-axis represents the clustering criteria low-agreement limit. The results on the F1-score and the other metrics are different. The standard deviation and mean value show more robustness to the variations of the parameters than the absolute and minimum values. A number of clusters and the clustering criteria ranges can be decided from these plots.

5

Increasing the fairness of algorithms for subjective properties classification

5.1. Introduction

In this chapter, we create algorithms for toxicity prediction which are fairer according to the definitions established in Chapter 4, and answer RQ3: ***how to build and train algorithms whose outputs are fair when predicting subjective properties of samples?***, by answering the following sub-questions:

1. *What are the current algorithms to perform sentence toxicity classification?* **(RQ3.1)**
→ We review the Computer Science literature to find the Machine Learning and Deep Learning algorithms currently used to perform toxicity prediction, and highlight their limitations.

2. *How to integrate the subjectivity of the property to predict into the algorithms' training process?* **(RQ3.2)**
→ We hypothesize that using the annotations instead of labels enables to take into account each opinion, what is tested by comparing algorithms trained on aggregated and non-aggregated annotations.

3. *How to integrate the user subjectivities into the classifiers?* **(RQ3.3)**
→ We test whether modifying the algorithms' architectures enables to take into account the different users this hypothesis, by comparing the performance of traditional algorithms with the ones of the proposed algorithms.

4. *How to model the psychology-related variables about toxicity perception to build a user profile and integrate it into the model architectures?* **(RQ3.4)**
→ We propose several encoding methods of the available variables and add them as inputs to the algorithms. We test these encodings by comparing the performance of these input-augmented algorithms with the performance of traditional Machine Learning algorithms.

5. *How to resample the dataset to enable the proposed algorithms to learn?* **(RQ3.5)**
→ We propose several criteria to resample the dataset. We show that balancing the dataset on these criteria increase the performance of the algorithms.

5.2. Formulation of hypotheses

In this section, we formulate hypotheses to answer RQ3. We investigate what current algorithms are for sentence toxicity prediction (RQ3.1) (subsection 5.2.1). Then, we propose modifications of these baselines to make the models fairer than they currently are (RQ3.2 to RQ3.5). We are both interested in proposing new architectures and training processes or modifying current ones, and new dataset resamplings.

5.2.1. Current methods for toxicity prediction

Here, we answer **RQ3.1**: ***what are the current algorithms to perform sentence toxicity classification?*** From the literature, we identify the Machine Learning algorithms employed for toxicity prediction, and quickly evaluate them since they will serve as a baseline on which to compare our new algorithms on.

There are three main directions employed to perform toxicity prediction (Section 2.2): using list of words and rule-based algorithms to compare the sentences on, using Machine Learning classifiers, and using Deep Learning neural networks. We are not interested in the first technique because it exhibits the worst performance, it is not modular -including the users to the process would be very memory-consuming since it would need word lists for each annotator-, and it is infeasible with the available data -collecting lists of words for each annotator by performing crowdsourcing would be very expensive. Consequently, we study Machine Learning algorithms and deep neural networks.

From the Machine Learning area, we choose three baselines: ***the Logistic Regression (LR) classifier, the Support Vector Machine (SVM) classifier and the Multi-Layer Perceptron (MLP)*** because they are the most researched classifiers and are shown to obtain higher performance than the list of words classifiers. Moreover, the LR and MLP are the two classifiers used in the paper working on toxicity prediction from which we take the Jigsaw dataset [111]. Although they present performance results on a different property annotated on the dataset (aggressiveness and not toxicity), we assume that the performance will vary in small ranges, and thus it is one more point of comparison to verify the correct training of the algorithms.

Since it is time-consuming to train Deep Learning models, we focus on one unique Deep Learning model, the study could easily be extended to other models later. The codes of papers [50] [1] and [9] [2] are available. In [50], they show that biLSTM with attention perform better than simple LSTM models. In [9], it is shown than CNN with GloVe embeddings perform better than LSTM. They do not compare with biLSTM with attention, so it is not possible to compare the two and make a decision. However, [50] use less training data (1528 comments, 435 labelled as hateful) than [9] (16K tweets, 3383 labelled as sexist, 1972 as racist), and the data might be more similar to our data (Fox News comments for [50], tweets for [9]). Therefore, we choose to use a ***biLSTM with attention*** for offensiveness prediction implemented based on [50]'s implementation.

We evaluated these baselines (Appendix C.1) and found similar results as in the papers. As expected, the performance of the more complex classifiers are higher than the simpler ones (in increasing order of performance Logistic Regression, Multi-Layer Perceptron and LSTM-RNN) because the Jigsaw dataset has many data that complex classifiers can learn more easily while the other classifiers might overfit.

5.2.2. Design of Machine Learning architectures for fairer models

Since the algorithms should output each annotation of the users of the system to be fair, we hypothesize that:

> H1: ***we cannot use the aggregated labels since the algorithms would then input unique labels per sample, but we have to use the disaggregated annotations.***

Simply feeding the algorithms with the annotations will not help them learn the users' opinions since they do not have a mean to distinguish the users, as seen in Appendix C.1. We devise a way to integrate the users' subjectivities in the models (**RQ3.2**, **RQ3.3**). The literature review identified three ways to adapt the outputs of algorithms to each user. Machine Learning researchers build one classifier per user in order to deal with different interpretations of a same sample. This approach is neither generalizable to unseen users, nor scalable if the dataset is constituted of many users. The Deep Learning literature employs different approaches to personalize algorithms. The algorithms are usually taught a representation of the users which is integrated into the neural networks in different places, be it the input or cells of certain layers, such as certain neural network- based recommender systems which use users' individual features as additional features to the inputs of the classifiers. Tang et al. [104] transform the network inputs into a user-specific representation by multiplying them with a user matrix learned for each user. This matrix has common parameters to each user and user-specific parameters. This method is meaningful for our use-case since it would transform the input sentences differently depending on the users' interpretation. We formulate hypotheses from these ideas.

We propose to evaluate whether augmenting the usual input features with annotators information leads to fair predictions in the case of sentence toxicity prediction. The psychology literature provided us with a set of human features which influence toxicity perception, among them gender, age, and education level are available in the Jigsaw dataset. That is why we use these features. We formulate the following hypothesis:

> H2: ***adding as input to classifiers the users' information that psychology literature defines as influencing variables for toxicity perception enables the models to output the opinions of each user.***

[1] `https://github.com/sjtuprog/hateful-speech-detection`

[2] `https://github.com/pinkeshbadjatiya/twitter-hatespeech`

The demographic information to input should be encoded in an exploitable way for the classifiers. We propose two encodings to model the demographic variables and input them into the models: 1) *one-hot encoding of the three variables and concatenation of these three representations **(H2.1)***, or 2) *continuous representation (between [0;1] for example) of each variable according to the available ranges in the dataset and concatenation of these three representations **(H2.2)*** **(RQ3.4)**.

This adaptation of the classifiers should return fair predictions only using a few features describing the users. It can be used with any kind of Machine and Deep Learning classifiers, what makes it adaptable. It is also a fast way to make predictions since adding a few features do not slow down the learning process and the prediction process much compared to only using the features describing the samples.

5.2.3. Design of training processes for fairer models

The hyperparameters of the different classifiers are usually chosen by performing a grid search over the ranges of values in which they are most likely to perform best. The grid search consists in training on a training dataset and evaluating on an evaluation dataset a classifier with its hyperparameters set to different values, and choosing the set of hyperparameters which exhibit the highest performance on the evaluation dataset. The performance metrics usually employed are the accuracy, precision, recall or F1-score, but they do not exhibit any information about the fairness of the model, so we hypothesise that using fairness measures instead of these usual metrics would improve the fairness of the models.

During the training process, the number of training data and features is chosen in order to reduce overfitting (because of a too large number of data or a too small number of features to describe the data) and underfitting (because of a too small number of data or a too large number of features). This is tested by plotting the learning and feature curves (model performance on the training and test sets as a function of respectively the number of training data and the number of features to represent the data samples) using the usual performance metrics. If the training and test curves are both low, the model is underfitting, if the training curve is high but the test curve is low, the model is overfitting. Consequently we hypothesise that:

H3: ***using the fairness measures as the performance metric to optimize when tuning the hyperparameters of the models with grid search, and when choosing the number of data features and training data according to the learning and feature curves, increase the fairness of the final model.***

5.2.4. Adaptation of the training dataset for fairer models

We assume that a raw dataset for which we have samples, annotations, and information about the annotators is not optimized to train Machine Learning models for two reasons. We observed in Appendix A.2 that the dataset is highly class-unbalanced, what could hinder the accurate training of the models, and thus we assume it is necessary to balance the toxicity classes. Moreover, we identified several aspects to understand potential unfairness of the models, and proposed to cluster the dataset on these aspects by uniformly dividing the dataset into equally-sized ranges of clustering criterion values (corresponding to the studied fairness aspect) to identify the unfairness. The largest clusters, mostly the clusters whose annotations correspond to high agreement between the annotators, receive predictions of higher performance than the other clusters. Sinces the disparity of cluster sizes is related to these unfairness, to answer **RQ3.5** we hypothesize that:

H4: ***balancing the training dataset over one of the clustering criteria used to study the multiple fairness aspects increases the fairness performance of the models trained with this resampled dataset.***

There are not enough data in the dataset to balance it using the four clustering criteria simultaneously. Possibly as future work we could quantify how much the resamplings improve the models' fairness and investigate several combinations of resampling methods using a subset of the four resampling methods.

5.3. Experimental set-up

In this section we explain the experimental protocol used to test the hypotheses of the previous section. Mainly we build the new models proposed in the hypotheses, evaluate their performance, and compare them with the performance of the baseline models (Section 5.2.1). Because of a lack of time due to the long training process of the deep neural networks, we only experiment on one Machine Learning algorithm.

5.3.1. Evaluation of the models

Evaluation process

We evaluate the models with the fairness evaluation method proposed in Chapter 4. Additionally, in order to compare the models to their baseline, we evaluate the models using the traditional Machine Learning performance metric (accuracy, precision, recall, and F1-score).

In Machine Learning, it is common practice to divide the dataset into a training set on which to train the models, and a test set on which to evaluate them. Having multiple samples on which to test the models enables to compute the mean of the performance value (what is a better approximation of the real performance of the models) and the standard deviation of the performance (what is a measure of the variation of the performance), so that an eventual comparison of the models could be done with significance tests.

In our case, the fairness evaluation method requires to use the whole set of test data to compute the fairness measures because they consist of both the mean and standard deviation over data from several clusters. Consequently, in order to still be able to perform significance tests when comparing the performance of several models, we perform a cross-validated evaluation. We divide the dataset into 10 folds, and 10 times we train the model on 9 of the folds and evaluate it on the remaining fold. Then we compute the average and standard deviation of the fairness performance over the 10 folds. We choose to use 10 folds because less folds would not enable to compute significant mean and standard deviation over the performance measures, and more folds is too time-consuming to evaluate the models.

Significance tests

Once the average and standard deviation of the fairness scores are computed over the 10 folds, we compare the scores between different models. This is equivalent to compare the means between two populations, the means being the average of the fairness scores, and the populations being the scores that are retrieved from each fold (the populations are constituted of 10 subjects). The null hypothesis for the difference between the means of models 1 and 2, μ_1 and μ_2, is $H_0 : \mu_1 = \mu_2$. The tests consist in comparing a theoretical statistic t_{th} and an experimental statistic t_{exp}. If $\left| t_{exp} \right| > t_{th}$, the null hypothesis is rejected, otherwise, they are considered equal with the confidence level α used to compute t_{th}. We choose $\alpha = 0.01$ and $\alpha = 0.05$.

In our case, the standard deviations and means of the two compared populations are not known but only estimated with the sample standard deviation and sample mean. Consequently we choose the two-sample t statistic as our test statistic. There are two versions of this test depending on whether the variances of the two populations are considered equal or different. The equality of variance can be tested with an homogeneity of variance test such as the Levene's test, however we did not do this because of time constraints, but we computed the two versions of the test to check whether there are differences in the results.

We note $\bar{X}_i$ the sample average, s_i the sample standard deviation and n_i the number of individuals in population i, i taking values 1 and 2 for the two populations. In the case where the variances are assumed equal, the test statistic is:

$$t_{exp} = \frac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}} \tag{5.1}$$

and the degree of freedom is:

$$df = \left\lfloor \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{n_1-1} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{n_2-1}} \right\rfloor \tag{5.2}$$

In the case where the variances are different, the test statistic is:

$$t_{exp} = \frac{(\bar{X}_1 - \bar{X}_2) - (\mu_1 - \mu_2)}{\sqrt{\frac{1}{n_1} + \frac{1}{n_2}}\sqrt{\frac{(n_1-1)s_1^2+(n_2-1)s_2^2}{n_1+n_2-2}}} \tag{5.3}$$

and the degree of freedom is:

$$df = n_1 + n_2 - 2 \tag{5.4}$$

For these tests to be used, because the number of samples per population is $n_1 < 30$ and $n_2 < 30$, the data must be normally distributed (that is an assumption we make). t_{exp} is computed considering $\mu_1 - \mu_2 = 0$. The theoretical statistic is found in the t-table of value by choosing $t_{th} = t\left(1 - \frac{\alpha}{2}, df\right)$.

5.3.2. The training process of the models (H3)

In cases where the hyperparameters of the models have to be tuned, k-fold cross-validation or its variants are employed, k evaluations being averaged in the end to choose the hyperparameters which return the highest performance in average. In our case, training and testing the algorithms is very time-consuming. Consequently we perform tuning of the algorithms simply by dividing the training dataset of the first iteration of the cross-validation into a training subset -80% of the data in the training dataset- and a validation subset -the remaining 20% of the data-, training the models with the different hyperparameters on the training subset, and evaluating them on the validation subset. The selected hyperparameters are the ones which return the highest performance on the validation subset. For the Logistic Regression classifier, we optimize the regularization clf_C and the tolerance for stopping criterion clf_{tol}.

As explained previously, it is necessary to plot the learning curve and feature curve to select the number of training data and the number of features which enable the models neither to overfit nor to underfit. Here the curves are plotted by evaluating the models on the training and test set after tuning the hyperparameters.

To study hypothesis 3 (H3), we compare the performance of the models trained with a number of features and a number of training data selected by computing the accuracy as performance metric or an average of the two measures (dispersion and general performance) of the main fairness performance metric (based on the annotation clustering using the ADR). We also study the influence of the metric choice for grid search by comparing the performance of models whose hyperparameters are selected based on the accuracy performance or based on a combination of the two main measures of fairness performance. The choice of the method to combine the two fairness measures for the grid search is detailed in Appendix C.3. The chosen combination is a weighted average of the normalized numbers of the standard deviation of the ranges in which vary the two measures during grid search. The models to compare are the ones that are adapted to increase fairness in H1 and H2, in order to be able to identify potential increase of fairness.

5.3.3. The datasets to train the models on (H4)

The dataset used is the Jigsaw dataset, that we clean by removing the annotations of the spammers using the CrowdTruth framework (Section 3.3.2). We filter the annotations which correspond to annotators who did not provide their demographic information and use them only for testing the models but not for training.

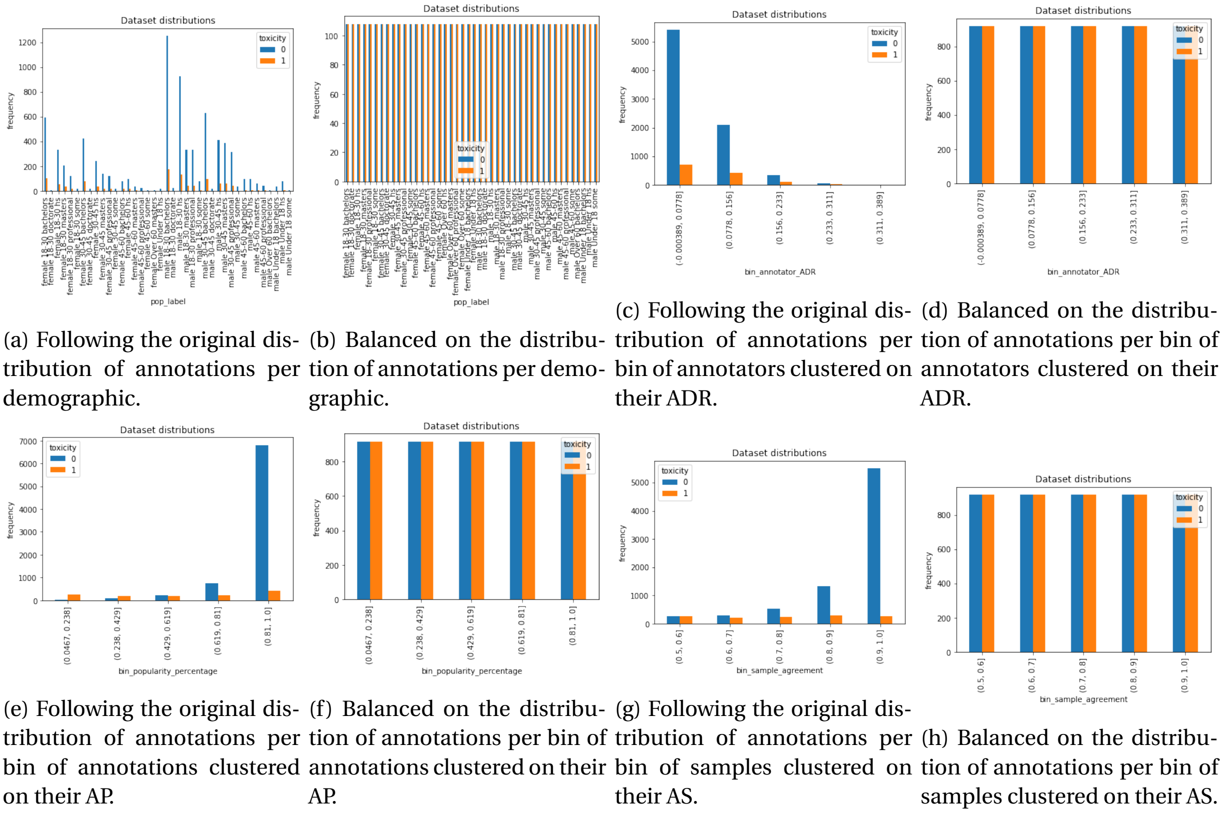
In order to test hypothesis 4 (H4), we prepare 8 resampled datasets: 4 datasets balanced on the two classes and balanced on the repartition of the clustering criteria values, and 4 datasets balanced on the two classes but following the original distribution of the clustering criteria values, the 4 clustering criteria being the annotators' demographic information, the annotators' average disagreement with the majority-vote (ADR), the annotation popularity (AP) and the sample ambiguity (AS). We train the models proposed in H2, H3, H4, on the disaggregated annotations of each resampled dataset, and compare the performance.

In order to compare the performance among the different datasets, we make them equally sized. According to the previous sections we need to divide each dataset into 10 folds, consequently we perform the same resampling on each fold so that each of them have a similar constitution. The resampling method is explained in Appendix C.4. The resamplings are presented in Fig. 5.1.

5.3.4. The Machine Learning models to compare (H1, H2)

In order to verify hypothesis 1 (H1), we compare the performance of a model trained on the aggregated labels (baseline) and a model trained on all the annotations. To test H2, we train the adapted models on the annotations, and compare their performance to their baselines (the simple models without additional input features) also trained on annotations, and trained on the aggregated annotations. The SVM and MLP classifiers and the Deep Learning models are time-consuming to tune and train, consequently we leave their evaluation as future work, and instantiate the models with the Logistic Regression classifier.

Each demographic information takes values corresponding to bins of possible values in the crowdsourcing task: age ('Under 18', '18-30', '30-45', '45-60', 'Over 60'), gender ('female', 'male', 'other'), education level ('none', 'some', 'hs', 'bachelors', 'masters', 'doctorate', 'professional'). The one-hot encoding (OH) considers each of these bin value as one feature, to which we add one feature per information category to represent the cases where the demographic information is unknown for the specific user. That constitutes $6+4+8 = 18$ features. For the continuous encoding, we attribute one value between [0;1] to each of the possible bins of each demographic category (also considering the additional unknown information bin values). For the age and education level which are ordinal variables, the values attributed to them follow the order of the bin values, with 0 representing the unknown information case. The continuous encoding consists in 3 features.



(a) Following the original distribution of annotations per demographic.

(b) Balanced on the distribution of annotations per demographic.

(c) Following the original distribution of annotations per bin of annotators clustered on their ADR.

(d) Balanced on the distribution of annotations per bin of annotators clustered on their ADR.

(e) Following the original distribution of annotations per bin of annotations clustered on their AP.

(f) Balanced on the distribution of annotations per bin of annotations clustered on their AP.

(g) Following the original distribution of annotations per bin of samples clustered on their AS.

(h) Balanced on the distribution of annotations per bin of samples clustered on their AS.

Figure 5.1: Presentation of the constitution of one fold for each dataset resampling. Each resampled dataset contains the same number of annotations.

5.4. Results and discussion

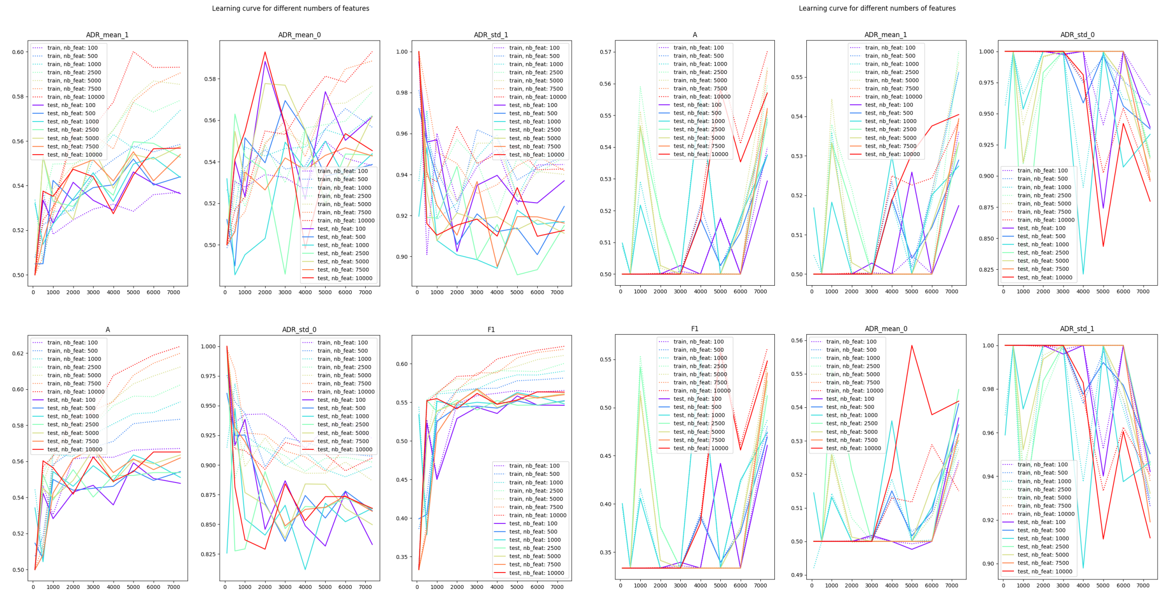
In this section, we present and discuss the results of the experiments.

5.4.1. Training process (H3)

In Fig. 5.2, we show an example of combined feature and learning curves for the two training processes. The learning curve computed on the accuracy shows an increasing trend for the accuracy-based training process, while for the ADR-based training process this evolution is not as linear. On the contrary, the dispersion and general fairness performance measures evolve more linearly for the ADR-based training process than the accuracy-based training process. This is easily explained by the choice of the variable which is optimized in each training process. Both training processes lead to the same conclusions: the more data (7000 data at most here) and the more features (10000 here) are used, the more the performance are maximized.

The significance tests on the performance of the models trained with different training processes are reported in tables such as Table 5.1. These tables comprehend the evaluation of the models on the fairness measures and traditional measures ($\bar{X_1}$, $\bar{X_2}$), the computation of the difference of performance between the models ($\bar{X_1} - \bar{X_2}$), the results of the computation of the experimental test statistic (t_{exp}) and the theoretical test statistic for the 2 values of α retained (t_{th}), and the results of the significance test in the last column (+ indicates that the null hypothesis is rejected and that the two models performance are significantly different).

For all the fairness metrics but not for the accuracy metric there is a significant difference in measures between the models trained with an accuracy- or an ADR- related training process. The dispersion between the performance within each cluster is lower for the ADR-related training process, while the fairness metrics which measure the general performance across clusters show higher performance for the accuracy-related training process. This is because the ADR-based training process takes into account the dispersion contrary to the accuracy-based training process. On the contrary, the accuracy-based training process chooses the hyperparameters whose values increase the accuracy of the classifier, what might simultaneously increase the performance of the models for the data in most of the clusters, and that is why the general performance values are higher for the models trained with the accuracy-based training process.



(a) Learning and feature curves for the accuracy-based training process.

(b) Learning and feature curves for the ADR-based training process.

Figure 5.2: Training of the Logistic Regression with continuous user model, on the balanced dataset along the annotation popularity percentage, evaluated with different performance metrics. The different curves correspond to the different numbers of features from 100 features (purple) to 10000 features (red), and evaluation data (dashed lines for training data and plain lines for test data). The shapes of the curves are inverted for the two training processes.

metric	$\bar{X}_1$	$\bar{X}_2$	$\bar{X}_1 - \bar{X}_2$	t_{exp}	t_{th}	signi.
ADR_discr_0	0.9263	0.9386	**-0.0124**	10.9525	2.201, 3.1058	**(+, +)**
ADR_perf_0	0.6122	0.5679	0.0444	40.4613	2.1314, 2.9467	**(+, +)**
ADR_discr_1	0.9470	0.9551	**-0.0081**	14.2168	2.1448, 2.9768	**(+, +)**
ADR_perf_1	0.6378	0.5939	0.0438	43.6362	2.1448, 2.9768	**(+, +)**
AS_discr_0	0.8811	0.9006	**-0.0195**	20.8404	2.1098, 2.8982	**(+, +)**
AS_perf_0	0.5794	0.5635	0.0159	47.1665	2.1098, 2.8982	**(+, +)**
AS_discr_1	0.8700	0.8883	**-0.0184**	13.8746	2.1448, 2.9768	**(+, +)**
AS_perf_1	0.5828	0.5677	0.0151	31.3883	2.1098, 2.8982	**(+, +)**
AP_discr_0	0.8391	0.8618	**-0.0227**	18.1327	2.1314, 2.9467	**(+, +)**
AP_perf_0	0.5559	0.5452	0.0107	30.5314	2.201, 3.1058	**(+, +)**
AP_discr_1	0.8980	0.9098	**-0.0118**	13.6751	2.1098, 2.8982	**(+, +)**
AP_perf_1	0.6281	0.6051	0.0231	20.8987	2.1098, 2.8982	**(+, +)**
demog_discr_0	0.6446	0.6960	**-0.0514**	9.3540	2.201, 3.1058	**(+, +)**
demog_perf_0	0.6118	0.5925	0.0193	14.3345	2.1098, 2.8982	**(+, +)**
demog_discr_1	0.8483	0.8839	**-0.0356**	8.0386	2.1199, 2.9208	**(+, +)**
demog_perf_1	0.6878	0.6359	0.0519	56.3902	2.1098, 2.8982	**(+, +)**
A	0.7715	0.7714	0.0001	0.0159	2.2281, 3.1693	(-, -)
P	0.8232	0.8065	0.0168	30.8088	2.1098, 2.8982	**(+, +)**
R	0.7715	0.7714	0.0001	0.0159	2.2281, 3.1693	(-, -)
F1	0.7923	0.7866	0.0058	1.5559	2.2281, 3.1693	(-, -)

Table 5.1: Significance tests between model 1 trained with accuracy-based training process, and model 2 trained with ADR-based training process. Computed on all the test data. Models: LR - continuous user model - balanced bin popularity percentage. The performance comparison on the different metrics are reported vertically.
Most metric comparisons show a significant between the performance of the two models.

Consequently, we decide to choose the ADR-based training process for the rest of the experiments. The difference is however not very large and we conclude that ***although hypothesis H3 is verified, the increase of fairness performance is not very large and usual training processes could also be employed. (H3)***

5.4.2. Dataset resamplings (H4)

Here we study the effect of the different dataset resamplings on the performance of the models.

Annotation-popularity dataset resampling

For the AP resampling, the fairness measures based on the demographic categories exhibit much higher performance with the original distribution of the dataset (around 0.2 difference of dispersion for values between

0 and 1), while the measures based on the AS score and the AP score show higher performance with the balanced distribution, with higher values for the dispersion than the general performance. The measures based on the ADR do not exhibit large differences. In the resampling, there are as many annotations which are representative of the majority-vote as annotations which differ a lot from it, what makes the models learn well the highly popular and the less popular annotations. That is why the measure of dispersion based on the AP score is higher with the balanced dataset. The measure based on AS of the samples focuses on measuring the dispersion of performance across samples whose annotations are all similar and samples whose annotations differ a lot. These clusters have a similar distribution in the training dataset as the clusters of balanced AP, so the measure of dispersion based on the ambiguity of the samples is also higher with the balanced dataset.

Concerning the annotator ADR-based fairness measure, the performance are very similar for each model, with slightly higher general performance with the dataset of original distribution. This might be because the clusters on which are based the computations contain as many annotations from annotators who often disagree with the majority vote as annotations from annotators who often agree with the majority vote. These annotators who disagree the most do not disagree for more than 30% of their annotations according to Chapter 3. Consequently, all the clusters contain data with generally high consensus, and therefore training a model on the dataset with original distribution leads to higher performance on this criterion because the dataset contains mostly high consensus data.

On the contrary, the measures based on the demographic categories show an important decrease of performance probably because the distribution of the AP score within each demographic category is similar, with more popular than unpopular annotations. Consequently, learning with more annotations leads to higher performance (and possibly similar performance) within each category, and that is why the dispersion and general performance measures are higher with the dataset of the original distribution.

Thus we conclude that ***the annotation-popularity based balanced resampling enables to increase the performance of the models on two aspects of fairness, the ambiguity score and annotation popularity based fairness measures, but not on the other aspects.***

Sample agreement dataset resampling

The trends between the models trained on the datasets with original or balanced distribution based on the AS are different from the trends observed for the samplings based on the AP. We observe that for all the aspects of fairness, the general performance measures are higher for the models trained on the balanced dataset while the dispersion measures are higher for the models trained on the dataset following the original distribution.

In the training dataset, there are as many annotations about samples which exhibit high consensus than annotations about samples with low consensus. Consequently there are more annotations which are usually representative of the majority vote because the clusters corresponding to the samples of low consensus comprehend both annotations equal and different from the majority vote, while the high consensus samples clusters almost only have majority vote annotations. There must be a more equal number of higher-consensus annotations in each cluster of the different clustering criteria, and consequently the dispersion decreases (lowest-consensus annotations are less "learned" by the model), while the general performance increases since data in most of the clusters are more evenly "learned".

Consequently ***the balanced resampling based on the sample agreement can be used when the general performances of the difference fairness aspects should be maximized.***

Conclusion

We conclude that ***depending on the fairness aspect to maximize, different resampling methods to make the training dataset balanced over one criteria can be chosen. (H4)***

In Fig. 5.3, a comparison of the performance of the models trained on the different resamplings, measured with different metrics, is given. It is useful to select which resampling to use in order to maximize one of the performance metric. For example, if one wishes to maximize the dispersion of the algorithmic fairness metric, one could choose to balance the dataset on an annotation level over the annotation popularity, this is however at the expense of the general performance of the algorithmic fairness metric which is generally lower than for the other models. On the contrary, one could choose to balance the dataset on a sentence level on the sentence ambiguity or on an annotator level on the ADR score or demographic categories in order to achieve a better balance between the two aspects of the algorithmic fairness metric, as well as to reach higher values of the traditional performance metrics. Each of these resamplings present slightly higher or lower performance values on different performance metrics but their general evolution trends are the same.

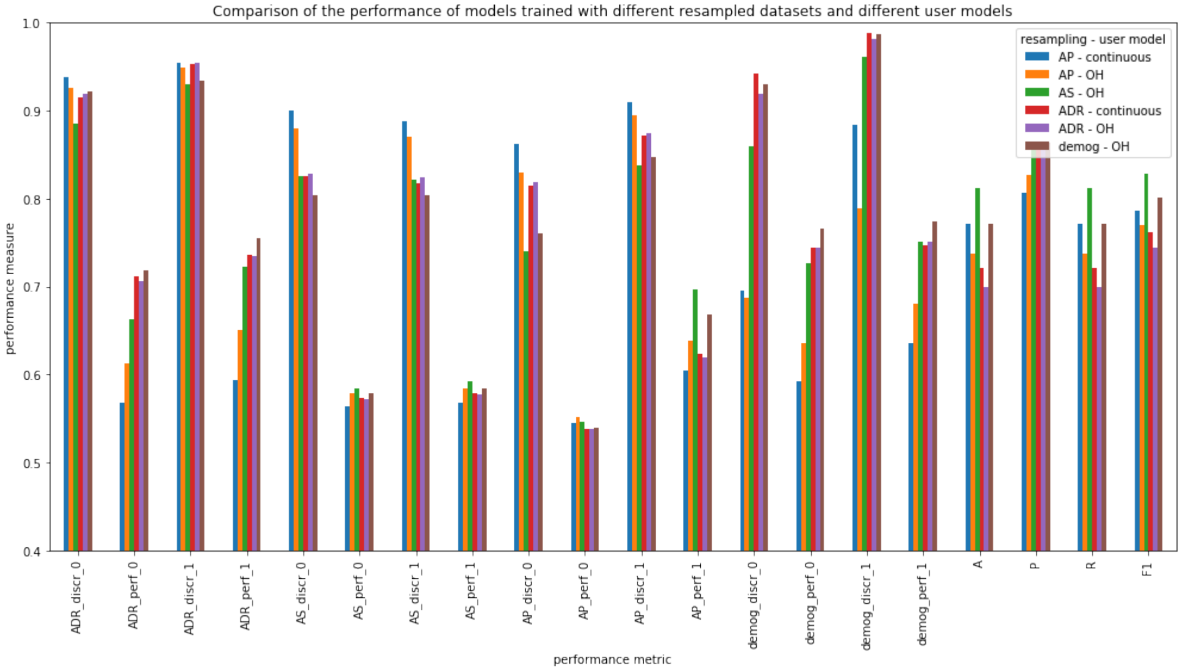


Figure 5.3: Comparison of the performance of the models trained on the different balanced resamplings with different user models. Logistic Regression. The performance trends are the same for the different user models, but not for the different resamplings. The algorithmic fairness performance metrics reported are indicated with "discr" for the dispersion aspect and with "perf" for the general performance, "0" indicates a computation on the whole dataset, while "1" indicates a computation on a smaller subset of the dataset (as explained in Chapter 4).

5.4.3. Machine Learning models (H1, H2)

First, we analyse the results on the balanced dataset over the AP. First, we compare the performance of the models not using any user model, *trained on the aggregated annotations and on all the annotations*. The evaluation on the usual metrics and the dispersion fairness measures exhibit higher performance for the model trained with all the annotations. This is because there are more annotations on which to train the models and consequently the global performance and the performance for each cluster increase. The general performance-based fairness measures however decrease, because the annotators are not differentiated and consequently all the clusters which do not correspond to a total consensus between annotations see their average performance decrease due to the upper limit of accuracy when no distinction is possible.

In Tables 5.2 and 5.3, we report the results of the comparison between the *models trained with the aggregated annotations and with all the annotations with the user model* (continuous and one-hot encoded respectively). The demographic-based fairness measures decrease when using the user models, while most of the other measures, especially the ones focusing on the discrepancy of performance across clusters, increase. We give in Appendix C.5 examples of high and low consensus data samples which are correctly or wrongly classified for each model. We notice that the models with augmented input features learn to predict the annotations of two types of sentences better than the usual models without additional inputs. They are able to predict correctly 1) the uncommon annotations of long sentences which are informative but might also be interpreted as toxic, and 2) the annotations of short non-toxic sentences whose grammar is incorrect.

We additionally analyse the results of the comparison between the models *trained with all the separate annotations, without or with user model.* The general performance fairness measures increase when using the user model, because the models are adapted to learn the users' specificities and distinguish between them. The dispersion-based performance decrease: the very low consensus annotations are not learned by the models while the higher consensus data are learned better than without user model, what contributes to a higher dispersion than without user model. This second batch of observations nuance our previous conclusions. Part of the improvement observed using the individual annotations might only be due to the fact that the datasets of annotations contain more data, and that consequently the models are able to learn better. However, it also enables to conclude that using adapted architectures (here inputs) of the models enables to increase certain aspects of fairness (especially the general performance across clusters), but that it does not make the dispersion performance increase because the very low consensus data are too difficult to learn. Having more data would probably enable the models to learn the low consensus data better.

Table 5.2: Significance tests between model 1 trained with aggregated annotations, and model 2 trained with individual annotations with the continuous user model. Computed on all the test data. Models: LR, trained with balanced sampling of cluster annotation popularity.

metric	$\bar{X_1}$	$\bar{X_2}$	$\bar{X_1} - \bar{X_2}$	t_{exp}	t_{th}	signi.
ADR_discr_0	0.9166	0.9386	**-0.022**	18.4933	2.201, 3.1058	(+, +)
ADR_perf_0	0.6413	0.5679	0.0735	61.9636	2.1448, 2.9768	(+, +)
ADR_discr_1	0.9424	0.9551	**-0.0127**	28.1282	2.1098, 2.8982	(+, +)
ADR_perf_1	0.6858	0.5939	0.0918	80.2558	2.1604, 3.0123	(+, +)
AS_discr_0	0.8524	0.9006	**-0.0483**	55.5514	2.1098, 2.8982	(+, +)
AS_perf_0	0.5664	0.5635	0.0028	8.0738	2.1098, 2.8982	(+, +)
AS_discr_1	0.8465	0.8883	**-0.0418**	33.0378	2.1604, 3.0123	(+, +)
AS_perf_1	0.5727	0.5677	0.005	11.2934	2.1199, 2.9208	(+, +)
AP_discr_0	0.7913	0.8618	**-0.0704**	77.7297	2.1098, 2.8982	(+, +)
AP_perf_0	0.5332	0.5452	**-0.012**	33.2595	2.1788, 3.0545	(+, +)
AP_discr_1	0.8677	0.9098	**-0.0421**	58.1453	2.1199, 2.9208	(+, +)
AP_perf_1	0.6535	0.6051	0.0484	51.5479	2.1098, 2.8982	(+, +)
demog_discr_0	0.9316	0.6960	0.2356	44.3116	2.2622, 3.2498	(+, +)
demog_perf_0	0.6927	0.5925	0.1002	65.2484	2.1199, 2.9208	(+, +)
demog_discr_1	0.9877	0.8839	0.1038	28.6858	2.2622, 3.2498	(+, +)
demog_perf_1	0.7090	0.6359	0.0731	75.5703	2.1098, 2.8982	(+, +)
A	0.7055	0.7714	**-0.0659**	10.9203	2.201, 3.1058	(+, +)
P	0.8387	0.8065	0.0322	65.3002	2.1199, 2.9208	(+, +)
R	0.7055	0.7714	**-0.0659**	10.9203	2.201, 3.1058	(+, +)
F1	0.7478	0.7866	**-0.0387**	9.7651	2.1604, 3.0123	(+, +)

Table 5.3: Significance tests between model 1 trained with aggregated annotations, and model 2 trained with individual annotations with the OH user model. Computed on all the test data. Models: LR, trained with balanced sampling of cluster annotation popularity.

metric	$\bar{X_1}$	$\bar{X_2}$	$\bar{X_1} - \bar{X_2}$	t_{exp}	t_{th}	signi.
ADR_discr_0	0.9166	0.9264	**-0.0098**	5.3677	2.1199, 2.9208	(+, +)
ADR_perf_0	0.6413	0.6127	0.0286	19.8926	2.1098, 2.8982	(+, +)
ADR_discr_1	0.9424	0.9496	**-0.0072**	9.5522	2.1604, 3.0123	(+, +)
ADR_perf_1	0.6858	0.6503	0.0355	27.4301	2.1098, 2.8982	(+, +)
AS_discr_0	0.8524	0.8795	**-0.0271**	29.2343	2.1199, 2.9208	(+, +)
AS_perf_0	0.5664	0.5794	**-0.0131**	36.7404	2.1098, 2.8982	(+, +)
AS_discr_1	0.8465	0.8701	**-0.0236**	23.7801	2.1314, 2.9467	(+, +)
AS_perf_1	0.5727	0.5844	**-0.0118**	27.6040	2.1098, 2.8982	(+, +)
AP_discr_0	0.7913	0.8298	**-0.0385**	25.6621	2.1788, 3.0545	(+, +)
AP_perf_0	0.5332	0.5510	**-0.0178**	62.2921	2.1448, 2.9768	(+, +)
AP_discr_1	0.8677	0.8946	**-0.0269**	29.9545	2.1604, 3.0123	(+, +)
AP_perf_1	0.6535	0.6378	0.0157	12.3177	2.1448, 2.9768	(+, +)
demog_discr_0	0.9316	0.6877	0.2438	84.4621	2.201, 3.1058	(+, +)
demog_perf_0	0.6927	0.6358	0.0569	36.3021	2.1199, 2.9208	(+, +)
demog_discr_1	0.9877	0.7892	0.1984	70.7631	2.2622, 3.2498	(+, +)
demog_perf_1	0.7090	0.6798	0.0292	24.4836	2.1098, 2.8982	(+, +)
A	0.7055	0.7376	**-0.0321**	9.0759	2.1098, 2.8982	(+, +)
P	0.8387	0.8263	0.0124	29.9003	2.1098, 2.8982	(+, +)
R	0.7055	0.7376	**-0.0321**	9.0759	2.1098, 2.8982	(+, +)
F1	0.7478	0.7698	**-0.022**	8.2811	2.1098, 2.8982	(+, +)

In Fig. 5.4, we show on data clustered over different fairness aspects an example of the performance difference of a model trained with no user model and a model trained with a one-hot encoded user model, with or without aggregation of the annotations into the majority vote. As expected, the clusters corresponding to the lowest consensus data receive predictions of lower performance than the clusters constituted of higher consensus data. For the data clustered according to the demographic categories of the annotators, the model which does not distinguish between users exhibit similar performance for each category of population because they all have similar distributions of consensus over the annotations (Appendix A.2). When employing a user model, the performance of the predictions become different across the categories of population because the populations for which there are more training data available see their predictions more adapted than the other populations. This is an indication that the training set would merit being more balanced if this aspect of fairness should be optimized. For the other clustering criteria, the model with a user model perform better on the low-consensus data than the models without user model, because it distinguishes between annotators and consequently make better predictions over data different from the majority vote. On high consensus data however, the model using a user model performs worse than the model without user model training on all the annotations, because it lacks training data to perform as well as without user model on this data while performing better on low-consensus data. It sill performs better than the model without user model and trained on the majority vote, what is an indication that hypothesis H1 is verified.Consequently, employing a user model makes the predictions fairer on our targeted aspects of fairness, but, due to the limited size of the training dataset, it also makes it more discriminative.

Therefore, we conclude that ***hypotheses H1 and H2 are verified: when using the individual annotations instead of the aggregated ones, with adapted models, it is possible to increase the different targeted fairness measures (and the traditional evaluation performance) (H1, H2)***, but it might not be possible to increase the discrimination-oriented fairness due to the limited size of the current training dataset.

We compare the two user property encodings in Table 5.4. The continuous encoding achieves higher fairness performance in terms of dispersion while the one-hot encoding achieves better fairness performance in term of general performance across clusters. ***Consequently, depending on the desired metrics to maximize, one might select one of the two encodings according to these observations. (H2.1, H2.2)***

We perform the same tests on the other resamplings of the dataset. For the dataset constituted of the original distribution of AP, the difference is not large between the models trained without user model and the models trained with a continuous or one-hot encoded user model. However, when there is a difference, it shows a slight improvement of the fairness measures for models including a user model. The performance differences between the continuous and one-hot encoded models are also small. For the dataset constituted of the original distribution of AS, we make the same observations as in the previous case. For the dataset

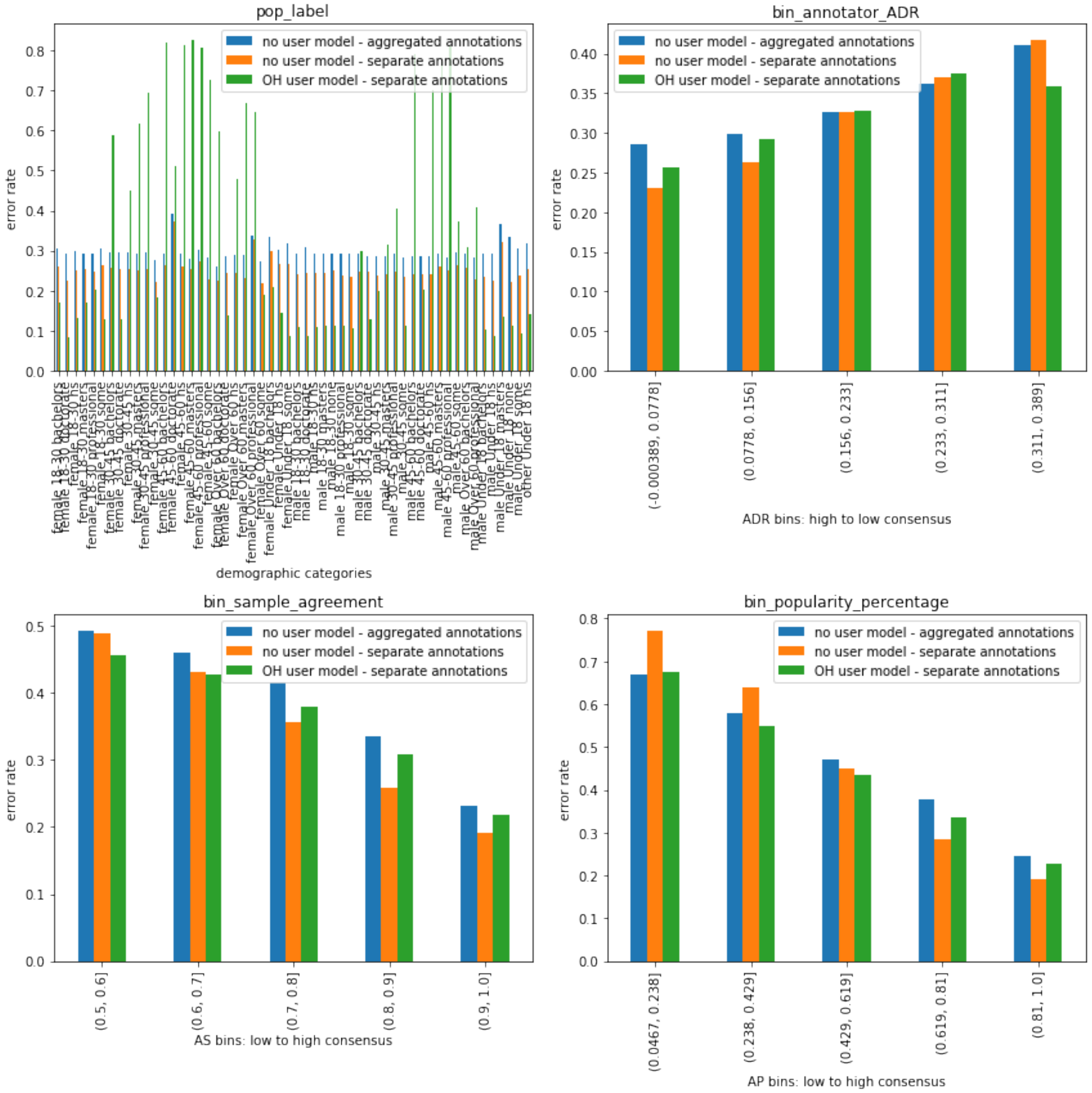


Figure 5.4: Comparison of the error rates of the predictions for the models without and with one-hot encoded user model, trained on the aggregated annotations and all the separate annotations. Logistic Regression trained on the dataset balanced over the annotation popularity.
Low consensus clusters receive predictions of poorer performance than higher consensus clusters. Models with users' model perform better on these low-consensus clusters and approximately equally well on the high-consensus data.

constituted of the balanced distribution of AS, using user models makes a significant difference compared to the performance of the models without user model, especially for the one-hot encoded user model for which the fairness performance difference are important. Generally, the same observations are made with the other resamplings. When the original distribution of the clustering criterion is kept in the dataset, the differences between the models are smaller, maybe because it is not easy for the models to learn both high and low consensus data with these datasets.

5.5. Conclusions, summary

In this chapter, we investigated how to modify current Machine Learning models **(RQ3.1)** made to predict subjective properties of samples in order to make their outputs fairer, and we focused on the task of automatically predicting the toxicity of sentences. We based our hypotheses on findings of the psychology literature. There are several variables which correspond to the personal background of each person, namely the age,

metric	$\bar{X_1}$	$\bar{X_2}$	$\bar{X_1} - \bar{X_2}$	t_{exp}	t_{th}	significance
ADR_discr_0	0.9386	0.9264	0.0122	8.1394	2.2281, 3.1693	(+, +)
ADR_perf_0	0.5679	0.6127	**-0.0448**	38.2222	2.1448, 2.9768	(+, +)
ADR_discr_1	0.9551	0.9496	0.0055	7.5812	2.1788, 3.0545	(+, +)
ADR_perf_1	0.5939	0.6503	**-0.0563**	59.2549	2.1314, 2.9467	(+, +)
AS_discr_0	0.9006	0.8795	0.0211	20.8367	2.1098, 2.8982	(+, +)
AS_perf_0	0.5635	0.5794	**-0.0159**	47.9042	2.1098, 2.8982	(+, +)
AS_discr_1	0.8883	0.8701	0.0183	13.0858	2.1199, 2.9208	(+, +)
AS_perf_1	0.5677	0.5844	**-0.0167**	34.3179	2.1098, 2.8982	(+, +)
AP_discr_0	0.8618	0.8298	0.032	20.9824	2.1604, 3.0123	(+, +)
AP_perf_0	0.5452	0.5510	**-0.0058**	14.1759	2.1199, 2.9208	(+, +)
AP_discr_1	0.9098	0.8946	0.0152	15.4435	2.1199, 2.9208	(+, +)
AP_perf_1	0.6051	0.6378	**-0.0327**	24.8570	2.1314, 2.9467	(+, +)
demog_discr_0	0.6960	0.6877	0.0083	1.4151	2.1604, 3.0123	(-, -)
demog_perf_0	0.5925	0.6358	**-0.0433**	32.9756	2.1098, 2.8982	(+, +)
demog_discr_1	0.8839	0.7892	0.0947	20.7016	2.1199, 2.9208	(+, +)
demog_perf_1	0.6359	0.6798	**-0.0439**	39.8457	2.1314, 2.9467	(+, +)
A	0.7714	0.7376	0.0338	5.4418	2.1604, 3.0123	(+, +)
P	0.8065	0.8263	**-0.0198**	40.5951	2.1199, 2.9208	(+, +)
R	0.7714	0.7376	0.0338	5.4418	2.1604, 3.0123	(+, +)
F1	0.7866	0.7698	0.0168	4.1217	2.1448, 2.9768	(+, +)

Table 5.4: Significance tests between model 1 trained with continuous user model, and model 2 trained with the OH user model. Computed on all the test data. Models: LR, trained with balanced sampling of cluster annotation popularity, with individual annotations.

gender, education level, and ethnicity, which influence how somebody perceives the toxicity of a sentence. Consequently we hypothesized that 1) augmenting the inputs of usual algorithms with these background information, encoded in a continuous or one-hot encoded ways, and training the corresponding models with a 2) dataset which consists of sentences, judgements by different people and individual information of these people instead of sentences and unique labels resulting from the aggregation of multiple annotations make the models fairer. We further hypothesized that 3) optimizing the hyperparameters of the models on the fairness performance instead of the accuracy performance also makes the models fairer. We also proposed to 4) resample the training dataset by balancing it over one fairness aspect to help the model learn.

We verified the hypotheses by comparing the fairness performance of traditional models to the fairness performance of our proposed models trained on the Jigsaw dataset, using significance tests, the models being instantiated with a Logistic Regression. We concluded that our hypotheses are correct: adapting Machine Learning models to take into account the personal background of the person judging a sentence makes the outputs of these models fairer than usual models **(RQ3.2, RQ3.3, RQ3.4)**. Moreover, different resampling methods enable to optimize different aspects of algorithmic fairness **(RQ3.5)**. Even though this might not enable to improve the accuracy of the models greatly, for example it is only improved of 0.02% for the proposed model trained on the annotations in the AP-balanced dataset (0.68) compared to the traditional model trained on the majority votes in the same dataset (0.66), it improves the fairness of their predictions. For people who very often agree with the majority vote (64% of the users of our models), the accuracy is only improved of 3% going from 0.71 to 0.74 ; but for the people who disagree between 30% and 40% of the time with the majority vote (0.38% of the users in our case) and are usually ignored by the models (the accuracy of traditional models is very low, around 0.59), the accuracy of the predictions is improved of 5% going up to 0.64%, what is an important improvement for them.

However, we observed a trade-off between the fairness of the models towards their different users, and the accuracy of their predictions. This limitations are due to several reasons. The accuracy of the predictions is limited by the prediction power of the classifiers used(trade-off between the complexity of the data to learn, the number of features and number of training data available), and investigating more complex models such a Deep Neural Networks could help improve the predictions. Moreover, the trade-off is also due to the fact that the employed dataset does not contain ethnicity information whereas it is claimed to be an important variable in the perception of sentence toxicity, and that even if it would contain this information, this set of available background information is not enough to account for all the different subjectivities of the different users. To overcome this limitation, it would be appropriate to investigate how to use identifiers of the different users without employing a limited set of background information to describe them (Appendix C.2.1).

6 Conclusion

In this chapter, we start by discussing the work done in the thesis and draw conclusions to answer the main research question. Finally we propose future work in consideration with our conclusions.

6.1. Discussion of current work

In this section, we analyse the pipeline of our system to highlight its potential strong points and limitations.

6.1.1. Focus of the work

Machine Learning (ML) is traditionally employed to perform classification tasks along properties which present high-consensus, like the objective task of predicting whether a radiography shows a cancerous tumour or not. Nowadays ML is also more and more used to predict subjective properties of content, such as predicting whether a video segment is perceived as violent, whether an image is perceived as of aesthetic quality, or whether a sentence written on the Web is judged toxic or not. We chose to specifically study the fairness of the predictions of these ML models. To the best of our knowledge, this is a completely new focus in the field of algorithmic fairness, since all other research focus on ML for the classification of people.

Although not all the hypotheses we proposed to answer the research questions and overcome current limitations of the predictions were verified by the experiments, we laid the ground work of the study of algorithmic fairness in the case of subjective properties classification. We consequently identified the areas which require more work to be done and constituted baselines on which to further experiment.

6.1.2. Interdependence of the three entities in the system

The thesis work is organized in three major parts: the dataset collection via crowdsourcing, the algorithmic fairness evaluation method and the adaptation of ML algorithms to the task of predicting subjective properties. We believe our approach combining these three areas is a strong and new point of our project.

Most research tackle solely one of the three areas in order to study or/and mitigate fairness of the predictions of ML models, what only enables to make small progress. These three areas are interdependent and consequently if one of them is of poor quality in the pipeline, the quality of the others is affected. For example, if the dataset collection step provides few or low-quality labels, the evaluation might be inaccurate because there is not enough data to compute significant measures, or the ground truth data might not be correct or might not contain all the possible perceptions of the subjective property. Researchers interested in improving the fairness of the outputs would report on small increases in the accuracy of the predictions of their models, but the models might output predictions which, although they correspond to the available ground truth data, are actually invalid since this ground truth is incorrect or constitutes a biased evaluation set.

On the contrary, we decided to study the three areas together to overcome these shortcomings, with a careful attention to reduce the influence of the possibly low-quality areas. We used a simple classifier whose behaviour is well known in order to investigate new evaluation methods of algorithmic fairness. We chose a state-of-the-art dataset to conduct our experiments, that is assumed to be of high-quality. We analysed it in details in order to identify its possible limitations and improve it. We analysed extensively our proposed evaluation method and the algorithmic fairness of our proposed models applied to the post-processed dataset, with many experiments, visualizations and significance tests on the outcomes of the models and evaluations.

6.1.3. Computation and validity of the fairness metric

The process of evaluating ML models with the new algorithmic fairness evaluation method using cross-validation is very time- and computing power- consuming but necessary when applying significance tests because, contrary to the accuracy, the standard error of the measures can not be directly estimated from the calculation of the metric over one group of data samples. However it would be relevant to investigate whether this standard error (or standard deviation) can be derived statistically from the mathematical definition of the metric, so that we could calculate the value without running the experiments multiple times.

It would also be relevant to investigate whether current fairness evaluation methods could be adapted to our task, and in the case where they could be adapted, whether they would provide similar interpretations about fairness of our multiple models to our measures of algorithmic fairness.

6.2. Conclusions

The predictions of ML models made to classify subjective properties of samples might be unfair for certain people, what can have harmful consequences for these people. That is why we had set up to study how fair these ML models currently are and how to increase their fairness ***(RQ)***. From the analysis of the models, we identified three problems which participate to their unfairness. 1) Current crowdsourcing methods to collect the datasets to train and evaluate the algorithms on create biased datasets which make the trained models unfair (Chapter 3). 2) There is a lack of definition of fairness adapted to our task, and consequently a lack of evaluation methods to evaluate algorithmic fairness (Chapter 4). 3) The architectures and training processes of the algorithms are not adapted to account for users' properties that are related to the subjectivity of the sample property in the domain at stake, but only optimize accuracy, what does not enable to take into account potential unfairness (Chapter 5). We tackled all these limitations in the thesis project.

Contribution 1: literature review. We proposed a comprehensive literature review of the different fields related to the three limitations in order to identify the exact causes of unfairness and formulate new hypotheses. The review enabled us to choose to study algorithms for the prediction of sentence toxicity because psychology showed that sentence toxicity is a subjective property, and this task is very important in order to limit the use of abusive language on the Internet. We selected to focus on dataset biases resulting from the aggregation of the crowdsourced annotations by majority-voting because it is the most common crowdsourcing method to ensure quality of the dataset. We found through literature what influences the perception of a content as toxic on a personal level, namely the age, gender and education level of the person looking at the content. Additionally, we found a state-of-the-art dataset (the Jigsaw dataset) and ML model (the Logistic Regression), which are adapted to study the fairness of toxicity prediction models.

Contribution 2: identification of current limitations of crowdsourcing methods, and of solutions. We analysed the Jigsaw dataset, and critically addressed the problem of dataset bias resulting from crowdsourcing tasks. We showed that the ***aggregation of annotations ignores a majority of people's line of thoughts*** (**RQ1.1**), with 51% of the annotators disagreeing around 15% of the time with the majority vote and 4.5% of annotators disagreeing at least 20% of the time. That justifies that the predictions of the ML models are unfair towards most people if they only represent the majority vote. We searched how to collect valid perceptions of sentence toxicity by crowdsourcing, while mitigating the majority vote bias and keeping the cost of the task low. Concerning the design of the crowdsourcing task, it was shown that ***although general crowdsourcing methods are applicable to the collection of annotations of subjective properties, techniques from psychology could be included to improve the results*** (**RQ1.2**). Concerning the post-processing methods of crowdsourcing results, it was proved that methods proposed in the literature are not appropriate to the current task, and that although the ***methods which compute quality scores of annotators to identify the wrong annotations (CrowdTruth framework) enable to remove spammers*** (which represent around 1.5% of the annotators), ***they do not enable to distinguish between the occasional mistakes of some annotators*** (estimated at 2.5% of the annotators) ***and the valid annotations which reflect the minority opinion*** (estimated at around 10% of the annotations) (**RQ1.2**). Concerning the cost of the annotation collection, we proposed to select the samples to annotate and the annotators by clustering the samples in meaningful ways, but we did not find a proper algorithm to obtain such clusters. It was concluded that more research is needed to collect the different perceptions of a subjective property via crowdsourcing with high-quality and at low-cost (**RQ1.3**). (***RQ1***)

Contribution 3: evaluation method for algorithmic fairness. Existing definitions and evaluation methods of algorithmic fairness are interested in algorithms which classify people and not in algorithms made to classify samples according to the perception people have of the properties of the sample. Consequently, we worked on a ***new definition*** focusing on the problem of accounting for personal perceptions instead of only presenting the perception of the majority. We defined an algorithm as fair when its prediction performance are equal for each user (**RQ2.1**). Also, we proposed a ***method to characterize and evaluate algorithmic fairness*** through clustering of the evaluation dataset into meaningful clusters on an annotation, user or sample level, and verified it in the context of toxicity (**RQ2.2, RQ2.3**). (***RQ2***)

Contribution 4: adaptation of ML models. We then turned our attention to ML models, and showed how to adapt them in order to make them output the different perceptions people have of a sample's property. We proposed to ***train the models with all the annotations instead of the majority vote, and to augment the inputs of current algorithms with the properties of the user*** for who the algorithm is asked to make a prediction (**RQ3.1-3.4**). Although it only increases the global accuracy of the models of 2%, it identifies the users who are usually ignored (because they often disagree with the majority vote) and who consequently usually receive predictions of very low accuracy (59% accurate), and it improves the accuracy of their predictions of 7%. It additionally helps reducing the disparities between the accuracy of the predictions made for users who often agree with the majority vote and the annotators who often disagree, what reduces the unfairness of the predictions: the users' accuracies span between 59% and 72% for traditional models (13% range), against 66% and 74% for our models (8% range). It is concluded that these modifications of the architecture and training process of ML models increase the fairness performance according to our definition. (***RQ3***)

Contribution 5: resampling of the training dataset. Additionally, we listed a choice of different ***resamplings of the training set into balanced datasets over different fairness-related criteria*** (**RQ3.5**) to help the models learn correlations in the data, and consequently ***maximize their performance according to the desired aspects of algorithmic fairness.*** (***RQ1,3***)

To summarize, we proposed a new evaluation metric of algorithmic fairness specifically for algorithms which realize classification of samples according to subjective properties. We concluded that targeted crowdsourcing can be used to collect properly balanced datasets of valid perceptions of subjective properties of samples, possibly at low cost, provided that the task is well-designed and that the results are post-processed appropriately. Finally we showed that ML models can be fairer according to our definition if their architecture and training process are modified possibly using the fairness metric, and if they are trained on these datasets collected via targeted crowdsourcing. Therefore the main hypothesis of the thesis which answers the main research question ***RQ*** is verified. ***Current ways to build datasets using crowdsourcing and to train ML models conduct to unfairness when classifying subjective properties, this unfairness can be mitigated by adapting both the methods to collect the training set and to build the models trained on it.***

A portion of the thesis work was published after the CrowdBias workshop of HCOMP2018 (Appendix E).

6.3. Proposition of future work

In this section, we propose future work in order to make the three parts of our system even more effective.

6.3.1. Application to different use-cases

The complete thesis work is based on systems for sentence toxicity prediction. However we aim at studying the prediction of subjective properties of samples in general. Consequently future work should also address the generalization of our method to other tasks which involve subjective properties. Mainly, the performance of each entity in the system could be investigated in other domains. The creation via crowdsourcing of other datasets, as well as the adaptation of the ML algorithms with features representing the different users' characteristics adapted to the task chosen, for example video segment violence prediction, should be investigated in different domains to check for generalization of the pipeline. The fairness evaluation method should however still be the same, we could simply check for its validity by comparing its expected and real performance.

6.3.2. Creation of a benchmark dataset

A benchmark dataset could be constituted to make the evaluation of the rest of the system independent of the crowdsourcing method used to create the dataset. This is difficult because it is not possible to define "expert annotators" when the property to annotate is subjective, but well-trained and trusted annotators could be asked to provide the annotations. There is no available dataset containing both data samples and annotations of the samples according to one subjective property, as well as information about the annotators. Building such a dataset, possibly large enough to train ML and Deep Learning algorithms, would be a useful contribution for both the Crowdsourcing and ML communities.

Two similar challenges would be to ensure that 1) every perception of the property on one sample and 2) all the different types of annotators with their different lines of thoughts are contained in the dataset. Possibly one could investigate how many annotations per samples, and how many different annotators with what kind of characteristics are necessary in a dataset to train algorithms with sufficient performance.

Additionally, there remains several questions to perform crowdsourcing task to gather data of subjective properties (Chapter 3). For the case of a dataset of sentence toxicity judgements, we should investigate how to make the crowdsourcing task design as clear as possible in order to eliminate all the causes of disagreement among annotators except the subjectivity of each annotator. For example the type of context information to precise for each sample to annotate, the selection of only one category of speech (racism, sexism, ...) and the questions to ask the annotators could be investigated based on methods employed in psychology. We could also collect annotations with more information about their annotators (ethnicity is an important variable according to psychology). For the post-processing of crowdsourcing results, how to filter out invalid annotations from valid but minority annotations remains a research question. For the collection of annotations at low-cost, we could investigate how to cluster the samples and/or annotators meaningfully in order to spread the collected annotations of certain samples or annotators to the rest of the elements of the cluster?

6.3.3. Adaptation of the Machine Learning algorithms

In our task, psychology literature points out variables which influence the perception of the property to annotate. However, in prediction tasks of subjective properties where the variables influencing the judgements are not known, it is not possible to make use of the characteristics of the annotators in order to adapt the predictions to each of them. Consequently, it is worthy to investigate how to adapt the ML algorithms in these cases. It is also useful in cases where these variables are known, but not specified for certain users of the system. To solve these questions, we proposes to investigate hypotheses H5 and H6 formulated in Appendix C.2.1.

Concerning the training process of the ML models, another way to improve fairness could be to modify their training loss function in order to take into account the different fairness metrics. For example the fairness metrics measuring the dispersion of cluster performance and/or the average performance could be added as regularization parameters on which to optimize the models, so that fairness is taken into account in the training process and not only in the selection of their hyperparameters and/or architecture.

Concerning the information used to train the models, we used features extracted from sentence samples and features representing a user's characteristics. However psychology literature highlights the importance of the sentence context in the perception of toxicity. Consequently it would be interesting to evaluate and measure whether modifying the architectures and/or input features of the models to take into account the sample context could improve the accuracy and the fairness of the predictions. This would require to create adapted datasets containing information about the context of the samples next to the samples and annotations.

Finally, we did not investigate in details feature engineering to encode the sentence samples. However extracting additional features which give more information such as the category of speech of the sample might help the training of the algorithms. This could be another direction to look into to increase the performance of the algorithms specifically made for sentence toxicity prediction.

6.3.4. Introduction of an Active Learning process

Active Learning is a particular type of training process used when a large part of the dataset is unlabelled. It consists in requesting annotators to provide labels for selected data samples, possibly selecting to maximize the information that the labelled samples would bring to the training process. Considering that we aim for the ML models to learn the subjectivities of each user, we need more data than the required quantity of data when having one unique label per sample. The data are costly to obtain, and we need to target specific users to collect their individual line of thoughts. Thus, Active Learning is an adapted training process for our task. We could investigate how to adapt current methods for the selection of the samples to annotate to the selection of the pairs of samples and users to obtain annotations on, what would enable to optimize the trade-off between the cost of the crowdsourcing task and the performance of the models.

From Chapter 5, we identified dataset resampling methods to efficiently train the models. We could use this information to proceed to the selection of the samples and annotators to ask for annotations, so that the dataset is not resampled after but during the crowdsourcing task. That might enable to spare annotations which would have been removed and to use these potential annotation power for missing classes.

Finally a last direction of research could be to investigate how this active learning process would perform online. For example we could investigate whether it is feasible in a given time period to collect the needed annotations to train the models with sufficient performance, especially considering that certain annotators' type might be available rarely since they have rare characteristics, such as a person of 15 to 18 years old (age) with a doctoral education (education level).

A

Analysis of the Jigsaw dataset

Here we list example sentences contained in the Jigsaw dataset, ordered according to the percentage of agreement between the annotators (Table A.1).

A.1. Example sentences contained in the Jigsaw dataset

A.1.1. Examples of sentences in the dataset ordered by toxicity annotator agreement

Table A.1: Example sentences ordered by agreement between annotators.

% agreement	Majority-vote	Sentence
50%		"I HAVE A VANDAL BOT AND I AM WILLING TO USE IT IF I HAVE TO! I WILL BAN EVERYONE!" "‘ ==Tesla stuff== Notice how all those “ugly American“ comments are merely part of Irismeister’s rant. ‘" "****Wow, you’re so clever. So smooth. Stop being an ass so we can compromise!" "The article is true, the Israeli policies are killing Arab children." "==Hi!== How dare you mollify moderate modify my comments? I do not modify your comments to pages! I am not a shoe sock puppet, I am a well-intentioned troll! I’m stone half the day, you know. By the way, what’s a sock puppet? Is that an American thing? "
60%	toxic	"After the wasted bit on his sexuality, I haven’t read the above diatribe but one thing for ABSOLUTE certain: Never in the history of this planet has Umberto ever been referred to as Humbert. It is this kind of lunacy that renders Wikipedia as a “fringe“ place for idiots to play. Get serious....DW‘" "Prior to Quickpolls, he would have been perma-blocked by now. Guess I’ll remove the block again. As soon as I get time, which will probably be about November. If you’ve got any brains, you will let that particular troll sleep on. Best."
60%	non-toxic	"::If I pick enough holes in you, will you turn into swiss cheese? ‘Ed" "I will tell you to stop.You think you a famouces.I Know you,I Know you Working to ISIS.I Send to CIA To Capture you." "== Get lost vandal! == Stop removing the notability tag from Steve Rackman. He’s not notable and you know he’s not notable! Stop making excuses and leave the tag alone! Better still, send it to AfD!"
70%	toxic	"Better believe it. At least I earned a Barnstar, Wikipedia’s highest honor, because I’ve been doing nothing but cleaning up your utter nonsense. In a way, I have you to thank, but I’ve gone through hell helping to maintain this site because of you. You drop your pathetic entries and run, never bothering to come back and fix anything. Your intelligence is suspect as is your taste in entertainment. Loser. I STILL dare you to answer me. -" "is the most evil wikipedian in the world I’ve ever seen."
70%	non-toxic	"I fixed the link; I also removed “homeopathy“ as an exampleit’s not anything like a legitimate protoscience, or even half-legit. It’s total pseudoscientific nonsense, and not taken seriously as many protosciences are. I’m willing to tolerate a sympathetic and historical treatment of it on its own page, but pages about real science shouldn’t be littered with frauds." ":::Yeah, it’s kind of silly. Ha, no good deed goes unpunished."
80%	toxic	"“‘Nazi filth“ is impolite‘" "Master Fag Mind your own damn biz those pages had no quarrel with you" "Delete my 200 articles Whhore" "I am a wanker who molests goats"
80%	non-toxic	"Please relate the ozone hole to increases in cancer, and provide figures. Otherwise, this article will be biased toward the environmentalist anti-CFC point of view instead of being neutral. Ed Poor"
90%	toxic	"== renault == you sad little bpy for driving a renault clio which has no vaa voom so there and the mcflurry is made of shit" "This user is the child of a whore."
90%	non-toxic	"This: :One can make an analogy in mathematical terms by envisioning the distribution of opinions in a population as a Gaussian curve. We would then say that the consensus would be a statement that represents the range of opinions within perhaps three standard deviations of the mean opinion. sounds arbitrary and ad hoc. Does it really belong in n encyclopedia article? I don’t see that it adds anything useful. The paragraph that follows seems much more useful. Are there any political theorists out there who can clarify the issues? It seems to me that this is an issue that Locke, Rousseau, de Toqueville, and others must have debated... SR" "== Why oh why... == You removed the trolls ANI section about Drmies.....WHILE I WAS EDITING IT, and I had collected almost 10 pictures from commons to use there. It was going to be such hilarious drama, you owe me my morning laughs....."

100%	toxic	"==You're a motherfucking idiot== You're stupid as all hell and know *nothing*! Butt out of the business of your superiors!" "**Fuck you, you psychotic fanboy retard."
100%	non-toxic	"== Japanese Scene == The largely neoclassical Japanese power metal scene should be mentioned somewhere."

A.1.2. Examples of sentences in the dataset ordered by Unit Quality Score

We give examples of sentences in the dataset ordered by Unit Quality Score computed on the binary labels (Table A.2) and on the full labels (Table A.3).

Table A.2: Example sentences ordered by Unit Quality Score.

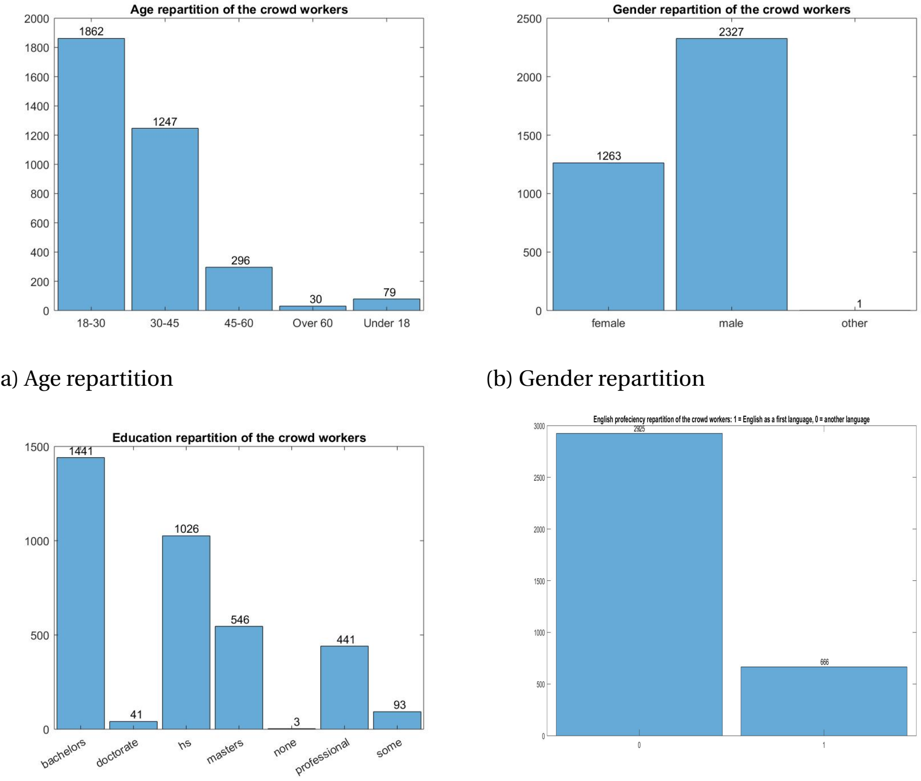
Unit Quality Score	Sentence
0.427	':::Ce type est un sauvage , , ; et stupide ! (Livonian extinct in one article, living in another one.) ( ) .
0.428	lololololololol ur ghey
0.428	YOU WILL BE ASSIMALATED
0.429	== Nlu == Mayne ya knoe Nlu ain't shyt,
0.429	== u smell 2 == its contaygius
0.429	==Hmmmm== Cozy, ain'it? Rumors are you two chumps have go'n yer mits on some treasha, and I wanna piece of dat poie!
s 0.433	== WHY DID YOU EDIT THE ARTICLE ON KURDISTAN? == DID YOU READ THIS DISCUSSIONS BEFORE YOU MADE THE CHANGES? They are adding silly things to the article.
0.433	Serendipodous by definition means...(ha!!) == diff Noticed your increasingly abusive comments at the Planetary Habitiability article, thought I would mention something you missed: There an new article on Serendipodousosity and guess what, it simply means loser. Hey can you be any more of a loser?
1.0	Regardless of how relevant or not relevant climbing the mountain is, that paragraph constitutes important historical background material as to previous occupations on the mountain. If you move it around you are going to severely mess up the flow of the section, for fairly little benefit. Yeah, it focuses much on the summit and upper flanks of Mauna Kea. In this case, there is no escaping it. There are sources to one, and not to the other; moving everything around has so far only messed up flow and made it hard to read; Nowiki's idea of verifiability based on other Wikipedia articles is a confligation of WP:SOURCE (do not use Wikipedia as a source) and would be immediatly removed come the FAC. All in all, the best course of action is to simply agknowledge the biase and move on from there. The section is hardly the most important one in the article anyway.
1.0	I think the structure's fine. What the word means should lead, then the core tenets, then other facets and aspects, then examples.
1.0	== hang in there == As a lurker on the FU pages I appreciate your attempt to bring a rational discussion to the pages. I am very impressed by your ability to withstand the invitation to sniping and to keep your temper. It was more than I could do. thanks.
1.0	'==The Redirect== I'm restoring the article per Talk:Tim Nordwind. Someone claimed Dan has no notability on his own, well, if that's true then the Larry Mullen Jr. & Adam Clayton articles should be redirects to the main U2 article too. '

Table A.3: Example sentences ordered by Unit Quality Score.

Unit Quality Score	Sentence
0.132	==Racial Policies/White Discrimination?== "There has been increasing discrimination against white running backs, defensive backs, and receivers, who have been less and less visible for the last 25 years. In 2005, a minimal majority of offensive linemen are white. Most quarterbacks, punters, and kickers are white, while almost all running backs, wide receivers, defensive backs, defensive linemen, safeties, punt returners, and kickoff returners are black." These sentences are awkward, and fairly innacurate. There are a number of very talented running backs and wide receivers of all ethnicities. There are also many punt returners and kickoff returners of all ethnicities - I'd like to see where this person got their information. Moreover, to say that there is discrimination against one group simply because it's mostly filled by another ethnic group is innacurate. For us to write in an encyclopedic article that there is discrimination, we should have some, you know, evidence of it. *If there are no sources, it is best to remove it.
0.134	==Merry Christmas!== LOL! I'm a dog person myself, but what the heck, we can all get along! . . .and a Happy New Year too!
0.142	== You smartass == You read the Introduction to editing!!!
0.144	If it makes you feel better, we can reword the article to read "Lehi justified acts of terror against their enemies" or somesuch, without specifically using the word "terrorism" in this context. (Btw, I'm puzzled as to how a defunct organization can still justify acts of terror.)
0.145	==Quadell's Sham Nomination== Thank you for having the balls to stand up to Quadell and tell him "No!, we don't want convicted sex offenders to have bureaucrat powers¡' Keep up the great work!
1.0	== media == Isn't there an example in Wikimedia Commons that can be added here?
1.0	This page needs a bot to fix the ISBN numbers.
1.0	== Now do you see why I quit? ==
1.0	::I think it's OK.
1.0	O.K. I forgot myself at Zilina.
1.0	== November 2009 == subst:test1|Iraq national football team -
1.0	REDIRECT Talk:Munir Hussain (cricketer)
1.0	== nonsense == are you a boy or a girl
1.0	people use for simplification reason
1.0	== image === Replace http://en.wikipedia.org/wiki/File:Yohkohimage.gif with http://commons.wikimedia.org/wiki/File:TheSun.png

A.2. General analysis of the Jigsaw dataset

Crowd workers informations In total, 4301 different annotators annotated the samples, from which 3591 have their background informations available. In Fig. A.1, we present the different bins in which the demographics information are classified, and the repartition of the workers into these bins. We plot in Fig. A.2 the workers distribution over the three retained demographics. The demographics categories are not represented equally in the dataset, which is normal since the crowdsourcing population is not representative of the whole population. However, we have to account for this imbalance if we consider that toxicity perception is different within each category.



(a) Age repartition

(b) Gender repartition

(c) Education repartition

(d) English native speakers repartition

Figure A.1: Presentation of the background information of the workers and their distribution in the pool of workers. The different demographic categories are highly unbalanced.

Annotation information In total, 1598289 annotations are available for 159686 unique comments. Most of the comments have 10 annotations, and never less than 8 annotations (Fig. A.3a). The repartition of the number of annotations each worker gave is almost uniform for 0 to 400 annotations per worker, but a large proportion of the workers made more than 400 annotations (Fig. A.3b).

The distribution of the toxicity score and toxicity labels are plotted in Fig. A.4. We observe that these repartitions are very unbalanced, what might be an obstacle to train the Machine Learning models accurately since they would be biased towards the most represented class (non-toxic). We investigate in the next chapters the effect of resampling the dataset to help the algorithms learn.

Analysis of the agreement between workers For each sample, we compute an agreement score considering binary labels (toxic / non-toxic), the score being the largest number of annotations which are of a same label divided by the total number of annotations for the sample. We present in Fig. A.5a the distribution of agreement scores. Most comments have a high agreement rate, the average agreement over all the annotations being 0.91761. However, there are still around 60000 comments with less than 100% agreement. It is for these comments that we want to enable the algorithm predictions to be tuned to each individual crowd worker. We further investigate in Fig. A.5b the agreement rate for the toxic and non-toxic comments separately (the

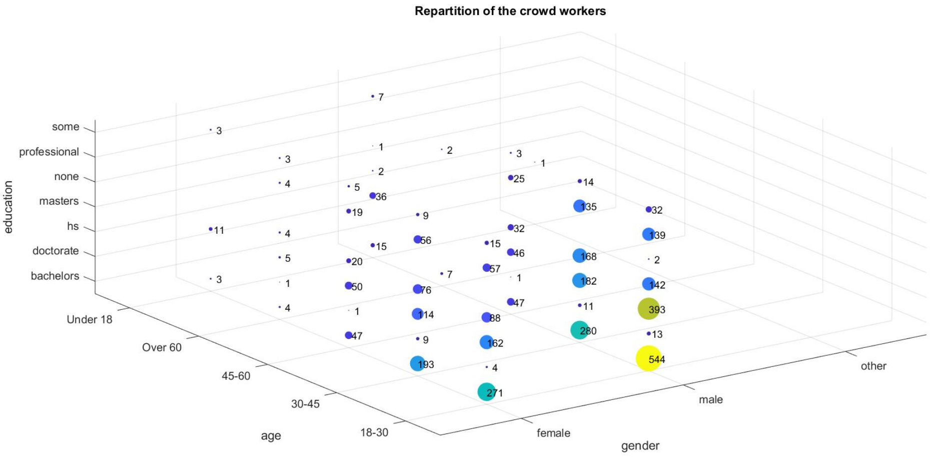
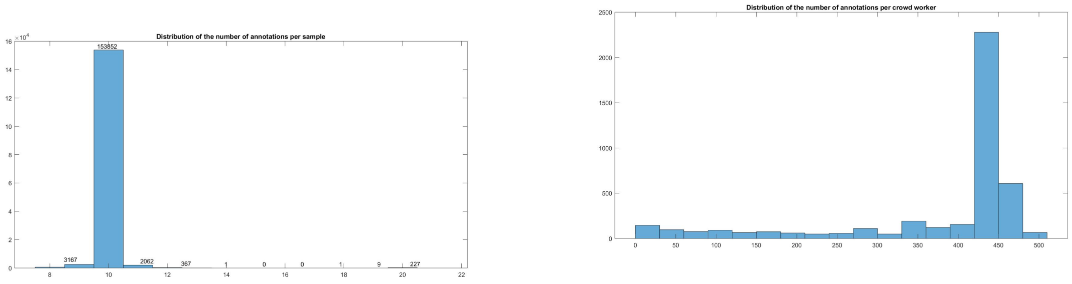


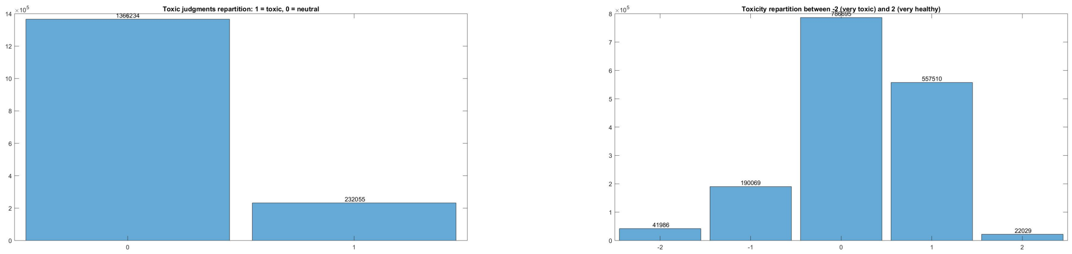
Figure A.2: Multi-dimensional distribution of the crowd workers along their background informations. The different demographic categories are highly unbalanced.



(a) Repartition of the number of annotations per sample

(b) Distribution of the number of annotations per crowd worker

Figure A.3: Analysis of the annotation distribution. Most samples have 10 annotations. Most annotators (around 450) provided a high number of annotations (around 2400).



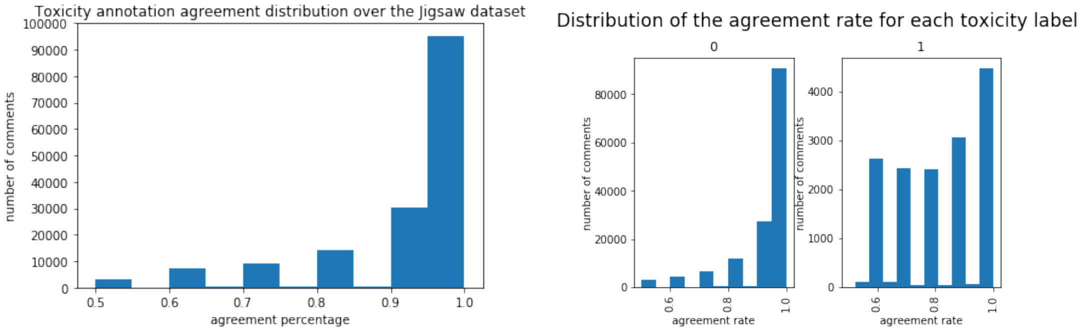
(a) Toxic/non-toxic comment annotations repartition

(b) Toxicity judgements

Figure A.4: Distribution of the toxicity judgements. The classes are highly unbalanced.

toxicity label considered is the label that the majority of annotators voted on, choosing toxic in case of equal number of annotations for each class). The judgements for the comments which are seen as toxic by a majority of people have an agreement rate which is more evenly distributed between 0.5 and 1 than the non-toxic comments. Comments which are not toxic must be less ambiguous or the definition of non-toxic comments might be clearer, while toxicity is a subjective property. Distinguishing between the annotators' perceptions would make the algorithms fairer since it would represent each annotator's point of view.

Examples of sentences contained in the dataset are reported in the Appendix A.1.1, ordered by the percentage of agreement over the toxicity labels. We observe that the more agreement about the toxicity there is, the more insult words are used in the sentences. The more non-toxic a sentence is judged, the longer it is, probably because it contains useful comments.



(a) Agreement percentage distribution over the Jigsaw dataset

(b) Agreement percentage distribution over the Jigsaw dataset, divided among the two toxicity categories

Figure A.5: Analysis of the annotation agreement distribution. Most annotations for each sample have high agreement. However, the annotations for the samples judged in majority toxic are more prone to disagreement than the ones of the non-toxic samples.

We also investigate the agreement rate distribution for each demographics category. First, we intended to plot the distribution of the standard deviations over the annotations of a same sample for each demographics category. However, these results are not significant because the dataset is only constituted of one to three annotations by workers of the same demographics for each sample. Instead, we decided to investigate how close the annotations of the individuals are to the majority voting label for the whole population of the demographics he/she belongs to. We plotted for each demographics the distribution of agreement rate between the annotations and the majority vote. However, this does not bring insights, we only observe that most of each demographics annotations are equal to the majority voting of the annotation -because there are few annotations per sample per demographics.

We finally decided to compute for each worker the average on which his/her annotations differ from the majority vote of the whole population (worker average disagreement rate (ADR)). Then, we plotted on Fig. A.6 the normalized distribution of this average within each demographics (from top to bottom, left to right the most frequent to less frequent demographics). We observe that for most demographic categories, the distributions are similar, with most of the workers always agreeing with the majority, and around 20% of workers disagreeing 10% or more of the time with the majority vote. This shows that no specific demographic category is disagreeing with the majority but part of the workers in each category is. Therefore, only using these demographics as features to predict toxicity perception might not be enough to represent the workers.

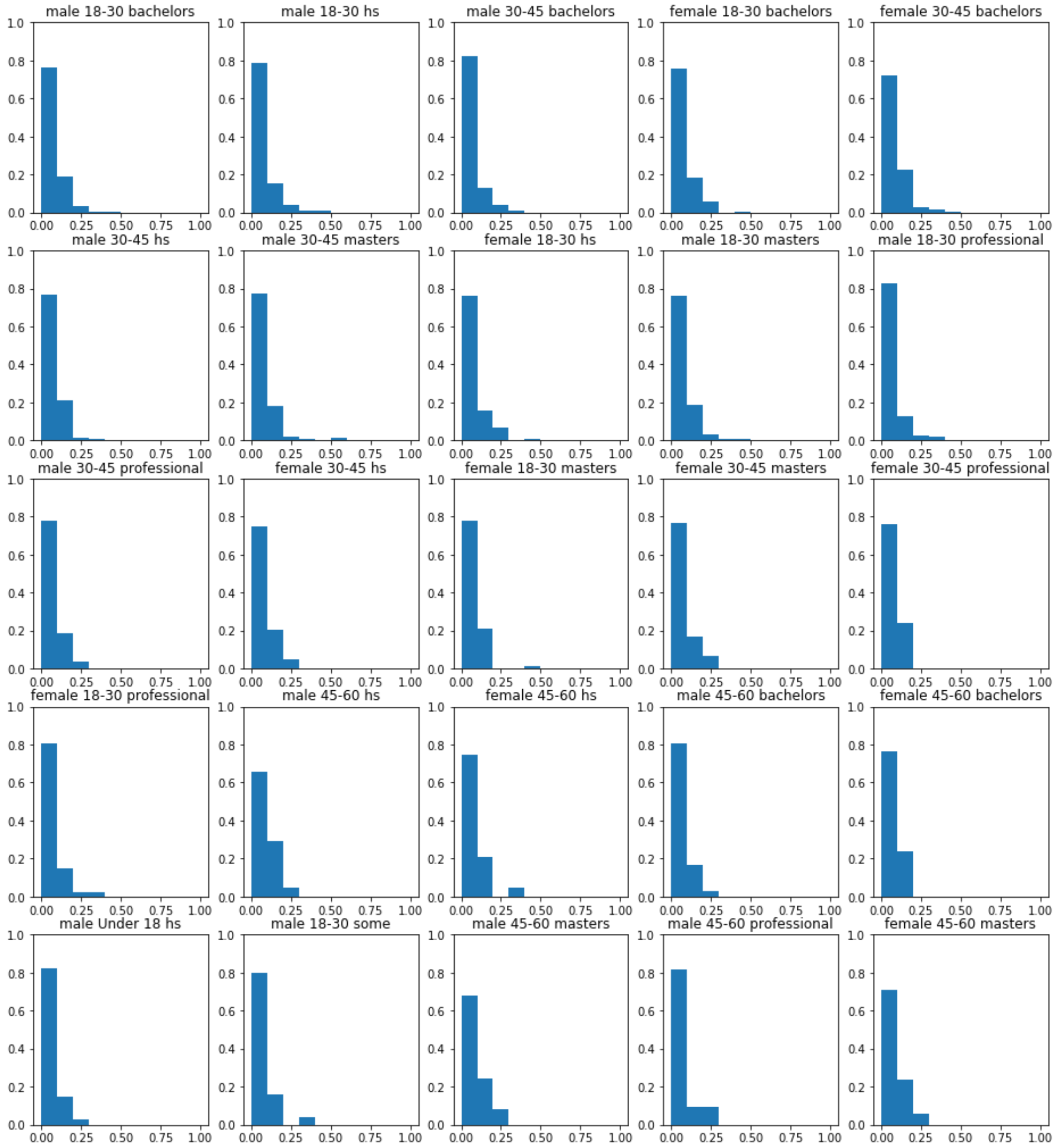


Figure A.6: Normalized distribution for each demographic category of the average disagreement each worker has from the whole population majority voting. The x-axis corresponds to the value of the ADR with the majority-vote, the y-axis corresponds to the proportion of the population within the demographic category for which the annotators have a specific ADR value. Only the 25 most frequent demographic categories are represented, the other ones show similar results or only one bar of full agreement between the annotators. All the demographic categories exhibit a similar distribution of ADR with the majority-vote.

B

Algorithmic fairness evaluation method

B.1. Expected behaviours of the different models

In this section, we list the behaviours related to fairness that we expect to observe on the predictions of the different models.

Behaviours for user-level fairness. We expect the fairness of models (1) and (2) to be low. They do not distinguish between the annotations but are fed with majority vote (MV) labels or the whole dataset (with a majority of MV annotations) and consequently are expected to return the MV label. For annotators who often agree with the MV, the performance of the models will be higher than for the others, what is unfair towards the annotators with opinions belonging to the minority. On the contrary, for models (3) and (4) which aim at distinguishing between users, we expect the algorithms to be better than models (1) and (2) at returning predictions corresponding to each annotation and not to the MV. Since model (3) only uses demographic information and we saw in the previous chapter that it is probably not sufficient information to distinguish between the opinions of the annotators of a same category, there should still remain unfairness. Especially for the annotators whose annotations often differ from the majority vote of each category, the performance will still be low. We expect model (4) to be the most fair for annotators which are known during training since it should return their specific annotations accurately, and the performance are expected to be the same as the performance of models (1) and (2) for the unknown users since no distinction between them is made. The proportion of high-accuracy predictions for each user should increase from models (1) to (4).

Behaviours for sample-level and annotation-level fairness. Sample-level fairness is studied by comparing the prediction performance of the model for each sample's set of annotations. Annotation-level fairness is studied by comparing the prediction performance of the model for each annotation. If we identify on which type of annotation and/or type of sample the model perform high or low, it would also give indications on the causes of unfairness. The expected observations for the user-level, sample-level and annotation-level fairness are similar. For models (1) and (2), we expect the samples for which the agreement between the annotations is high to receive high performance while the ones for which the agreement is low should receive low performance since only one unique label will be outputted by the models whereas several different annotations are expected. If we group the annotations according to their percentage of "popularity" among all the annotations of one sample (rate of the number of equal annotation to the studied annotation among all the annotations for one sample), we expect the two models to be accurate for the high-popularity annotations since they correspond to the opinions of the majority and that is the labels the algorithms are trained on. For models (3) and (4), we expect the performance for each sample to increase, and the performance among the different bins of annotations to become more similar, since the models should be able to distinguish between annotators and predict the individual annotations (including the annotations of the minorities). For model (3), samples for which the annotators disagree the most should obtain higher performance, for model (4) these performance are expected to be even higher for samples on which known users gave annotations. For samples annotated by unknown users, we do not expect change compared to the first two models. The proportion of high-performance predictions for each sample should increase from models (1) to (4).

Behaviours for discrimination-related fairness. A model is considered unfair from a discrimination point of view when its prediction performance are different for different categories of population, for the different demographic categories (by age, gender and education level) in the Jigsaw dataset. For models (1) and

(2), we expect that the performance are similar for most of the demographic categories since it was observed in the previous chapter that almost each demographic category has a similar disagreement distribution with the majority vote. For models (3) and (4), the outputs are expected to be closer to the expected annotations. The demographic categories which have more data in the dataset are expected to have higher performance than the other ones since the algorithms should learn better to distinguish between the annotators. However, for model (3) there should be a maximum-accuracy limit since it is only able to return one label per demographic category, and the difference with models (1) and (2) outputs might not be large since the majority vote of each category might already accurately enough represent the annotations of each individual in the categories. For model (4), we expect the performance for the most frequent demographic categories to be higher than for the other set-ups since the known users should have more accurate predictions. If the training dataset is balanced over demographic categories, each category should have similar performance. Otherwise, the most frequent categories are expected to have higher performance.

B.2. Characterization of algorithmic fairness

In this chapter, we present experimental results and figures we used to investigate the different possible characterizations of the fairness of Machine Learning algorithms.

B.2.1. First batch of experiments

We first investigated general characterizations of algorithmic fairness. These first experimentations enabled us to refine the characterizations in the next subsection.

Scope of the experiments

For the different fairness aspects, we divide the test set according to the different clustering criteria, and we plot the performance of the model on the different clusters for each performance metric usually used to evaluate Machine Learning algorithms. We only make the experiments on the models (1), (2) and (3) because model (4) is not compatible with the Logistic Regression.

Results

The results of the first batch of experiments are grouped into plots similar to Fig. B.1. Since there are many plots, and some are very similar to each other, we only describe the results instead of reporting all the plots. The heatmap plots represent the model performance computed taking the individual annotations as ground truth. Each column corresponds to an evaluation metric on the training or test set. Each heatmap is divided into two parts vertically, on the left the metrics are computed on the annotations whose annotators' demographic information are known and on the right on the annotations whose annotators' information are unknown. The horizontal divisions represent the different clusters on the different clustering criteria.

User-related fairness On Fig. B.1 the clusters correspond to clusters of annotators depending on their average disagreement with the majority vote. The top clusters correspond to high disagreement and the bottom clusters to low disagreement, so the top clusters are expected to get lower performance than the lower clusters. As expected the workers with a higher agreement with the majority vote receive higher performance than the workers with low agreement with the majority vote for each model. The performance using model (3) for the low quality workers increases, what was also expected. Similar observations are made when using the CrowdTruth Worker Quality Score. Therefore, we conclude that these visualizations are meaningful to interpret one possible cause of unfairness of the models. These clustering criteria are "human-interpretable".

Sample-related fairness We proceed to the same experiments with clusters of samples depending on the ambiguity score. The top clusters correspond to low ambiguity and the bottom clusters to high ambiguity (low agreement over the labels of the sample), so the top clusters are expected to get higher performance than the lower clusters. This is the behaviour we observe on the F1-score and the accuracy plots for example. However, the F1-score, the precision, recall and AUC values exhibit very small variations in between the clusters in comparison to the performance computed with the accuracy, and thus these metrics seem to be less adapted to study fairness. Therefore, to study sample-level fairness, we decide to investigate only accuracy (and F1-score only to verify that it is not significant mathematically). The performance's difference observed between the models distinguishing or not between users is very small. It could be that model (3) do not improve fairness regularly according to this clustering criteria or simply that the difference of performance across clusters is low and nothing can be observed. The observations are similar on the UQS clustering criterion.

Discrimination-related fairness Finally we draw the heatmap of clusters of the demographic categories present in the dataset.In general there does not seem to be large differences in performances in-between the clusters, except for a few clusters which differ by 30% on the accuracy. This characterization shows that as expected the discrimination power of the models is low but still existent for a few categories.

Conclusion: choice of a small number of evaluation metrics

First of all, we note that the observations appear clearer on the accuracy evaluations than on the other metrics. For example, the accuracy value in each bin from the high-agreement to low-agreement workers decreases along the bins whereas for the other metrics the evolution of the values is not linear. Therefore, we choose to work on the accuracy in all the following chapters, even though the fairness metrics proposed afterwards could be implementing using any of the metrics above (it depends what the purpose of the person implementing the algorithm is).

Moreover, the plots for the F1-score, precision, recall, and AUC are similar. We do not study the Spearman correlation because it is not meaningful to evaluate binary labels (with a binary ground truth) with it. We decide to retain the F1-score since it is a combination of the precision and recall and thus should give an information on both, and it takes into account class imbalance in its evaluation more than the AUC.

B.2.2. Second batch of experiments

Characterization on a sample-level Fig. B.2 present the performance of the models on a sentence level.

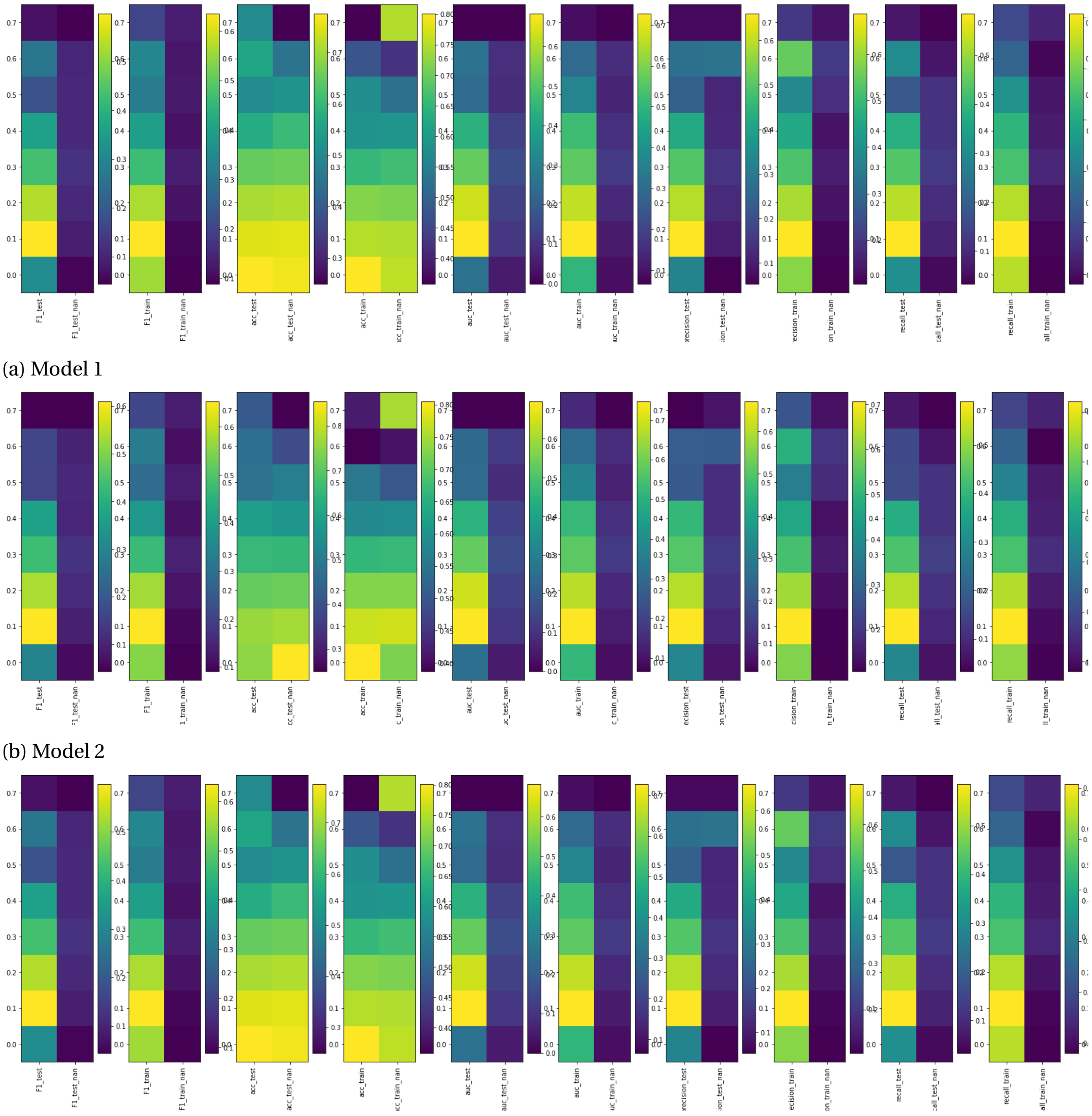
Clustering on the user-level (ADR, WQS, demographic) The characterization based on the average disagreement rate with the majority vote is plotted in Fig. B.3. As expected, low disagreement (bottom of the y-axis) leads to higher performance compared to the higher-disagreement clusters. Both F1-score and accuracy heatmaps show differences between the models: model (3) tends to perform better on middle-disagreement data than model (1).

Concerning the characterization based on clusters formed according to the Worker Quality Score, the workers with a high quality (top of the y-axis) get higher performance compared to the low quality worker, because the quality of the workers is computed in relation with the agreement of their annotations with the other workers' annotations. As expected, both accuracy and F1-score show differences between the models: model (3) performs better than model (1) for all types of workers except the lowest quality clusters.

The expected behaviours are also observed on the demographic-related plots (Fig. B.4). Except on a few of the demographic categories, most of the categories exhibit similar performance, what was expected, so this evaluation method is valid to highlight discrimination-related aspects of unfairness.

Clustering on the annotation-level (AP) As expected, the clusters of the most popular annotations have higher performance than the ones with lower popularity for both the accuracy and F1-score. The F1-score shows improvements on the less popular clusters with model (3), consequently it is a meaningful characterization. Analysing the total accuracy does not exhibit differences between models (1) and (3), contrary to the analysis of the accuracy performance on each class. On the negative class which is dominant in the dataset, there are few differences between the two models, but on the positive class we observe that model (3) has a slightly higher accuracy on both popular and unpopular annotations. We are interested mainly in recognizing the positive class, therefore this clustering criteria seems to be a valid characterization.

Clustering on the sample-level (UQS and AS) We expected high-UQS data to receive higher performance than low-UQS data because the high quality is a sign of high agreement among annotations and therefore of easier labels to learn. However, the performance in each cluster do not follow a linear evolution which means that the UQS might not represent disagreement in a way that we can interpret easily. Concerning the performance evaluation based on the ambiguity score, although we observe a general trend of having higher performances for clearer sentences (sentences with high-agreement on the label), the evolution between the clusters is not exactly linear for low-agreement samples. The observations are similar for both the AS and UQS which are computed similarly, consequently the representation might give valid insights into the potential unfairness, even if they slightly differ from the expected behaviours (it might be that certain annotations on certain samples are less present in the dataset and so harder to learn).

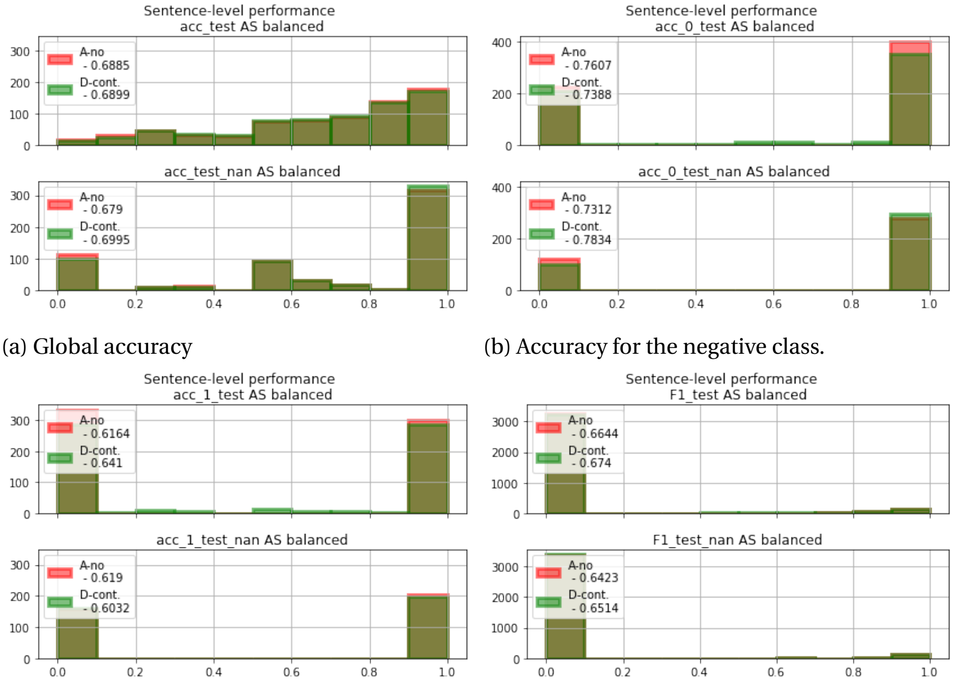


(a) Model 1

(b) Model 2

(c) Model 3

Figure B.1: User-level binning: average disagreement with the majority-vote bins.



(a) Global accuracy

(b) Accuracy for the negative class.

(c) Accuracy for the positive class.

(d) Global F1-score.

Figure B.2: Visualization based on the sentence-level performances. Comparison of models (1) (red) and (3) (green).

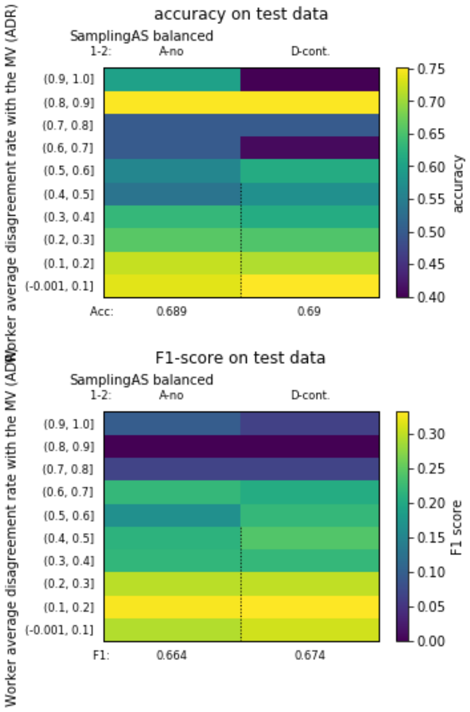


Figure B.3: Visualization of the accuracy and F1-score based on the ADR clustering criteria. Comparison of models (1) (left) and (3) (right).

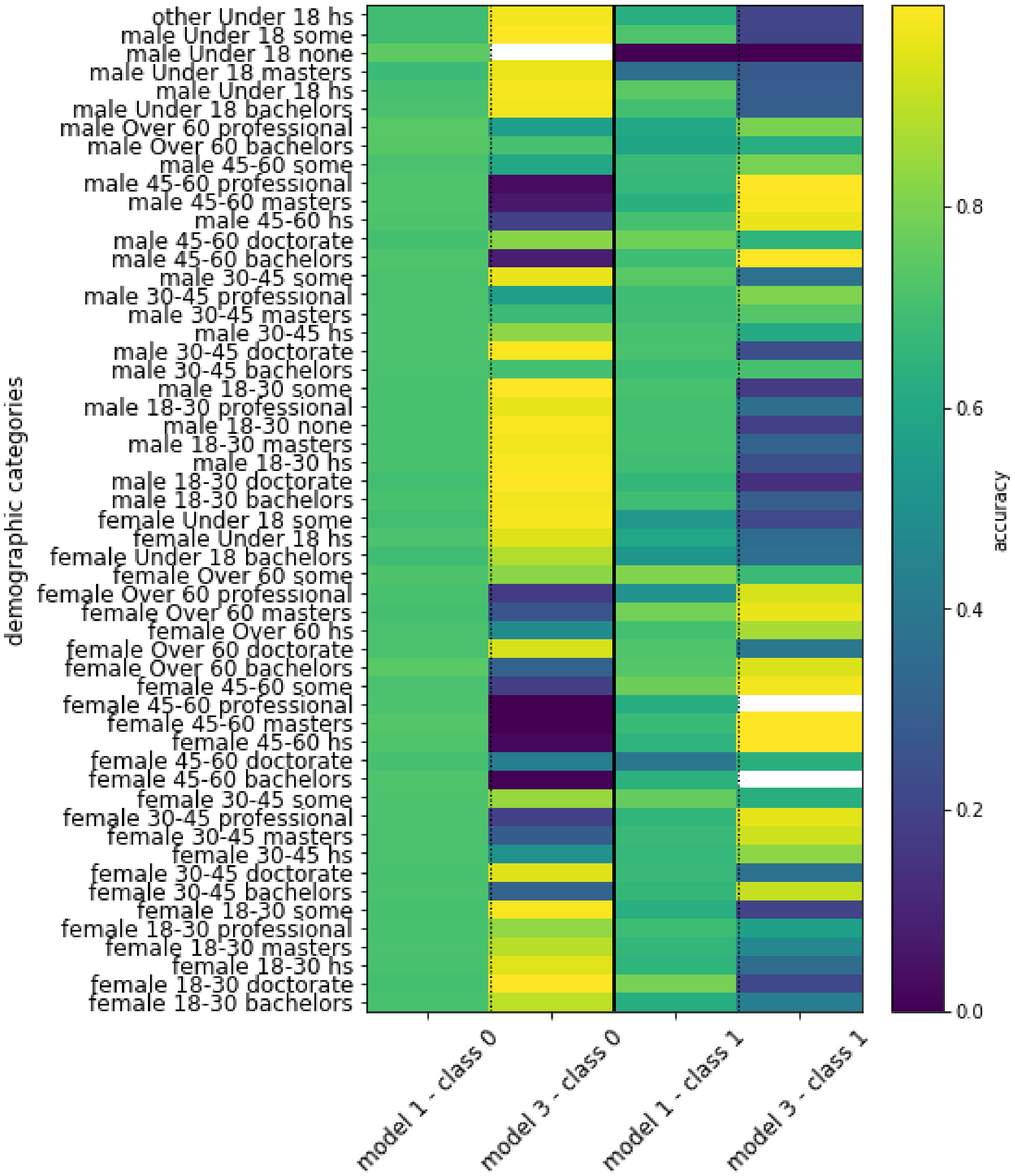
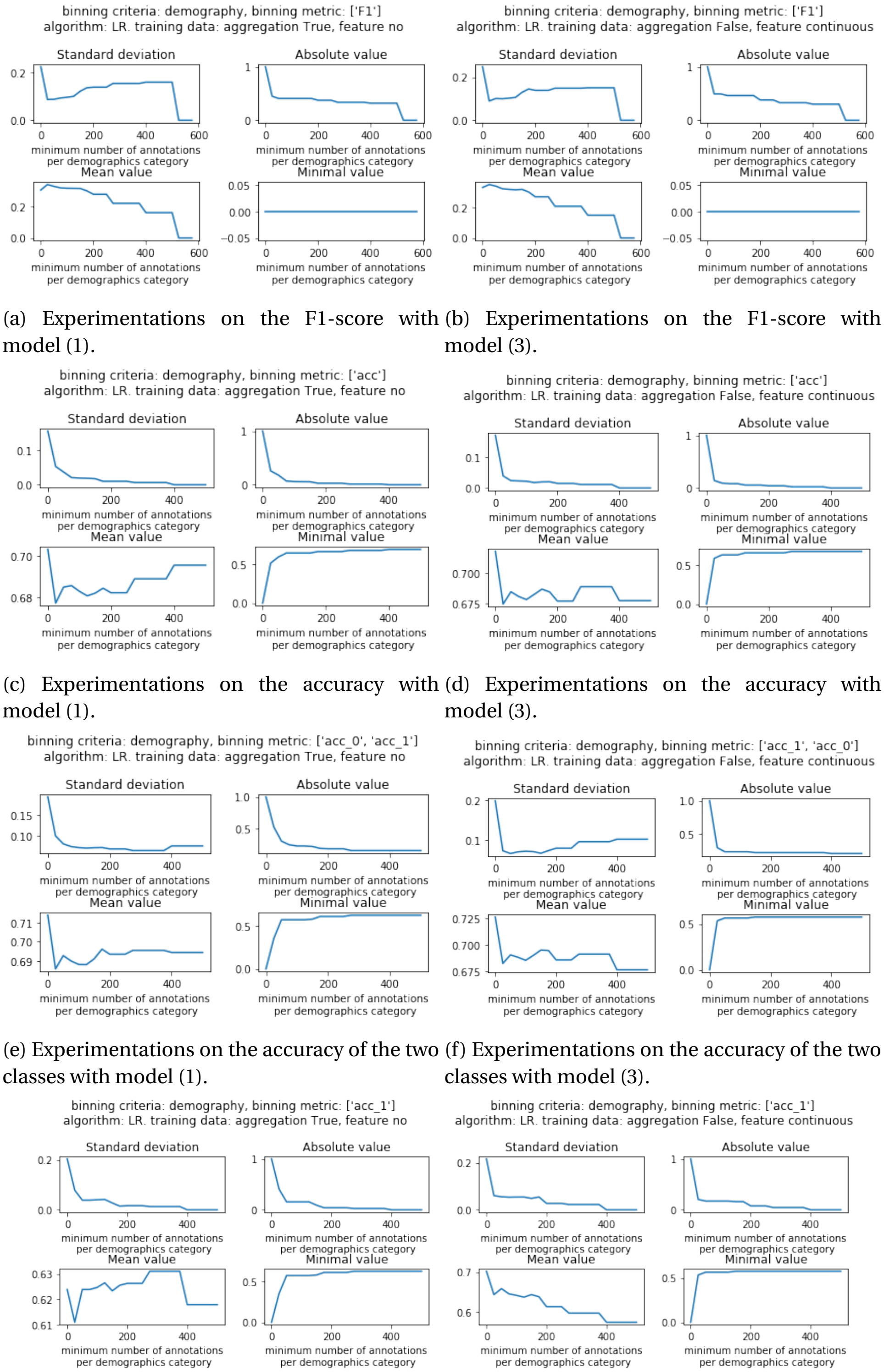


Figure B.4: Visualization of the accuracy on class 0 (non-toxic) and class 1 (toxic) data, based on the demographic clustering criteria. Comparison of models 1 and 3.

B.3. Investigation of the possible fairness metrics

Fig. B.5 shows the results of the experiments on the variables of the discrimination related fairness metric.



(a) Experimentations on the F1-score with model (1).

(b) Experimentations on the F1-score with model (3).

(c) Experimentations on the accuracy with model (1).

(d) Experimentations on the accuracy with model (3).

(e) Experimentations on the accuracy of the two classes with model (1).

(f) Experimentations on the accuracy of the two classes with model (3).

(g) Experimentations on the accuracy of the positive class with model (1).

(h) Experimentations on the accuracy of the positive class with model (3).

Figure B.5: Experimentations on the discrimination-based fairness computed with different evaluation metrics (F1-score, global accuracy, accuracy on the positive class, separated accuracy on the two classes). The x-axis of these plots represents the minimum number of annotations that a demographic category should have to be maintained in the test set.

B.4. Investigation of the significance of the performance of the clusters

Experimental set-up

We compute the significance test over the performance of model (1) since it is the model which should exhibit the most unfair behaviours. The metrics showing the most unfairness are the metrics which should be chosen since model (1) is a baseline not adapted to solve possible unfairness and we aim at highlighting its unfairness. The ADR-related fairness is based on the average performance value for each cluster, the average being computed over the performance of each user of the cluster. We consider that each user's performance is a sample of a population -the population being all the members of a cluster-, and we pose the null hypothesis H0 "two populations from two different clusters are the same". We perform a significance test between the populations in order to verify or refute the null hypothesis. Unfairness corresponds in our case to different performances among the different clusters and consequently to rejection of the null hypothesis. Since the populations are not normally distributed (as shown in Fig. B.6), we choose to use the Kolmogorov-Smirnov test. It usually requires at least around 20 samples per population to compute significant results. We choose the common alpha value of $\alpha = 0.05$.

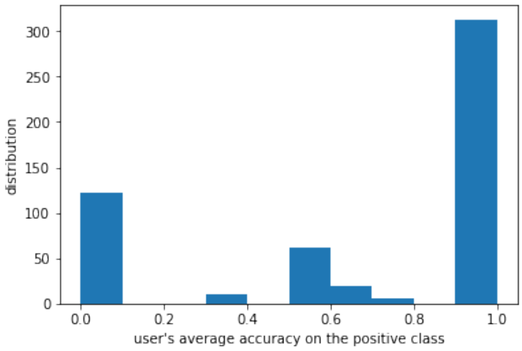


Figure B.6: Histogram of the user's positive class accuracy of the lowest disagreement bin for model (1). These accuracies are not normally distributed.

Results of the significance tests

ADR-related fairness. We divide the dataset into 10 clusters. Since the populations are not large enough for the cluster ranges between 0.5 and 1, we decide to group all of the users in these clusters and redo the computations with these new clusters, using the true positive rate as performance metric of each cluster. The results are reported in Tables B.1 (statistical value) and B.2 (p-value). We see that most of the p-values are under α, what leads us to reject the null hypothesis and to say that for most of the clusters the average accuracy difference is significant. We conclude that it makes sense to compute the dispersion between clusters in order to measure unfairness. Using 10 clusters with the true positive rate gives significant comparisons. We do the same computations on the negative class. All the returned p-values are higher than the α value. We conclude we can not only use the accuracy on this class to study fairness. We replicate the calculations with the total accuracy over the two classes. It leads to the same conclusion as for the accuracy on the positive class. The calculations on the F1-score comprehend more values which do not reject the null hypothesis. Therefore we conclude that we can base our calculations on a combination of the true positive and true negative rates.

Ambiguity score-related fairness. We apply the same significance tests on the ambiguity score clustering criteria, using 5 clusters because one cluster out of 2 has less than 10 samples. Only few of the cluster values are refuting the null hypothesis using the true positive rate. This might be explained because only computing the accuracy on the positive class is not significant when there is only one positive annotation for the sample. For the accuracy over the two classes, all the results refute the null hypothesis.

B.5. Application of the algorithmic fairness metrics to the models

We apply the final fairness metrics to the models (1) to (3) to make a final verification on whether the fairness-related behaviours expected on the different models are exhibited by the metrics. The results are reported in Fig. B.7, B.8.

bins	0.1	0.2	0.3	0.4	0.5	06
0.1	0.	0.0358	0.122	0.2024	0.274	0.373
0.2	0.036	0.	0.097	0.167	0.249	0.348
0.3	0.122	0.097	0.	0.080	0.152	0.252
0.4	0.202	0.167	0.080	0.	0.094	0.193
0.5	0.274	0.249	0.152	0.094	0.	0.107
0.6	0.373	0.348	0.252	0.193	0.107	0.

Table B.1: T-value of the significance test between each ADR bin of configuration (1) on the positive class, on the reorganized bins. NA corresponds to tests where one of the two bins is empty - the computation is not effectuated. The bin values indicated in axis are the upper value of the range (each bin having the size 0.1 average disagreement).

bins	0.1	0.2	0.3	0.4	0.5	06
0.1	1.0	8.761e-1	**1.578e-3**	**7.675e-6**	**3.297e-7**	**2.932e-3**
0.2	8.761e-1	1.0	**2.173e-2**	**4.107e-4**	**4.947e-6**	**6.862e-3**
0.3	**1.578e-3**	**2.173e-2**	1.0	3.238e-1	**2.081e-2**	1.063e-1
0.4	**7.675e-6**	**4.107e-4**	3.238e-1	1.0	4.756e-1	3.831e-1
0.5	**3.297e-7**	**4.947e-6**	**2.081e-2**	4.756e-1	1.0	9.713e-1
0.6	**2.932e-3**	**6.862e-3**	1.063e-1	3.831e-1	9.713e-1	1.0

Table B.2: p-value of the significance test between each ADR bin of configuration (1) on the positive class, on the reorganized bins. The bin values indicated in axis are the upper value of the range (each bin having the size 0.1 average disagreement). In bold are the values which make the null hypothesis rejected.

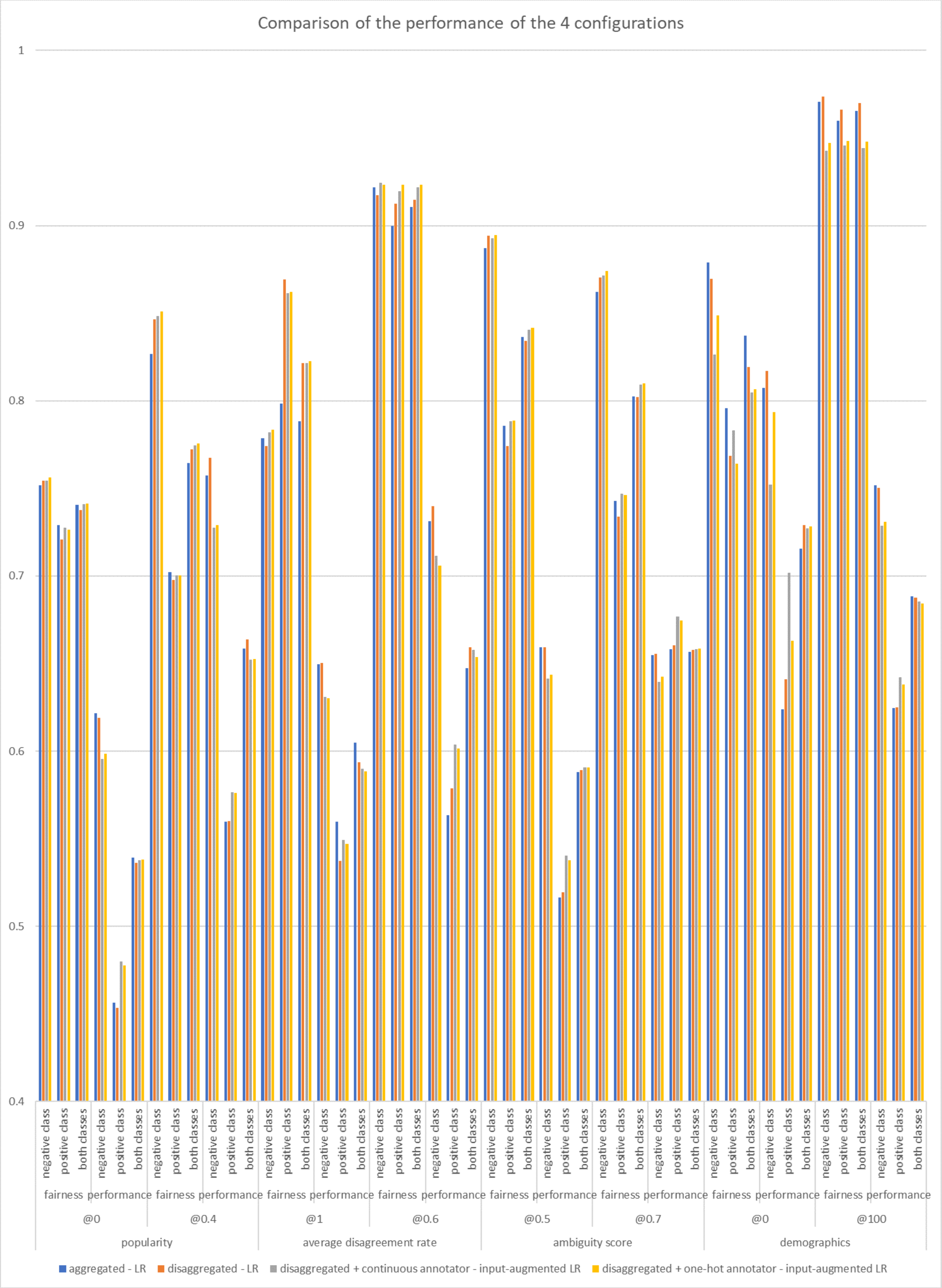


Figure B.7: Comparison of the performance of the 4 models, based on the two separate classes and both classes accuracies. The models using users' information as additional features perform better than the other models for most algorithmic fairness aspects.

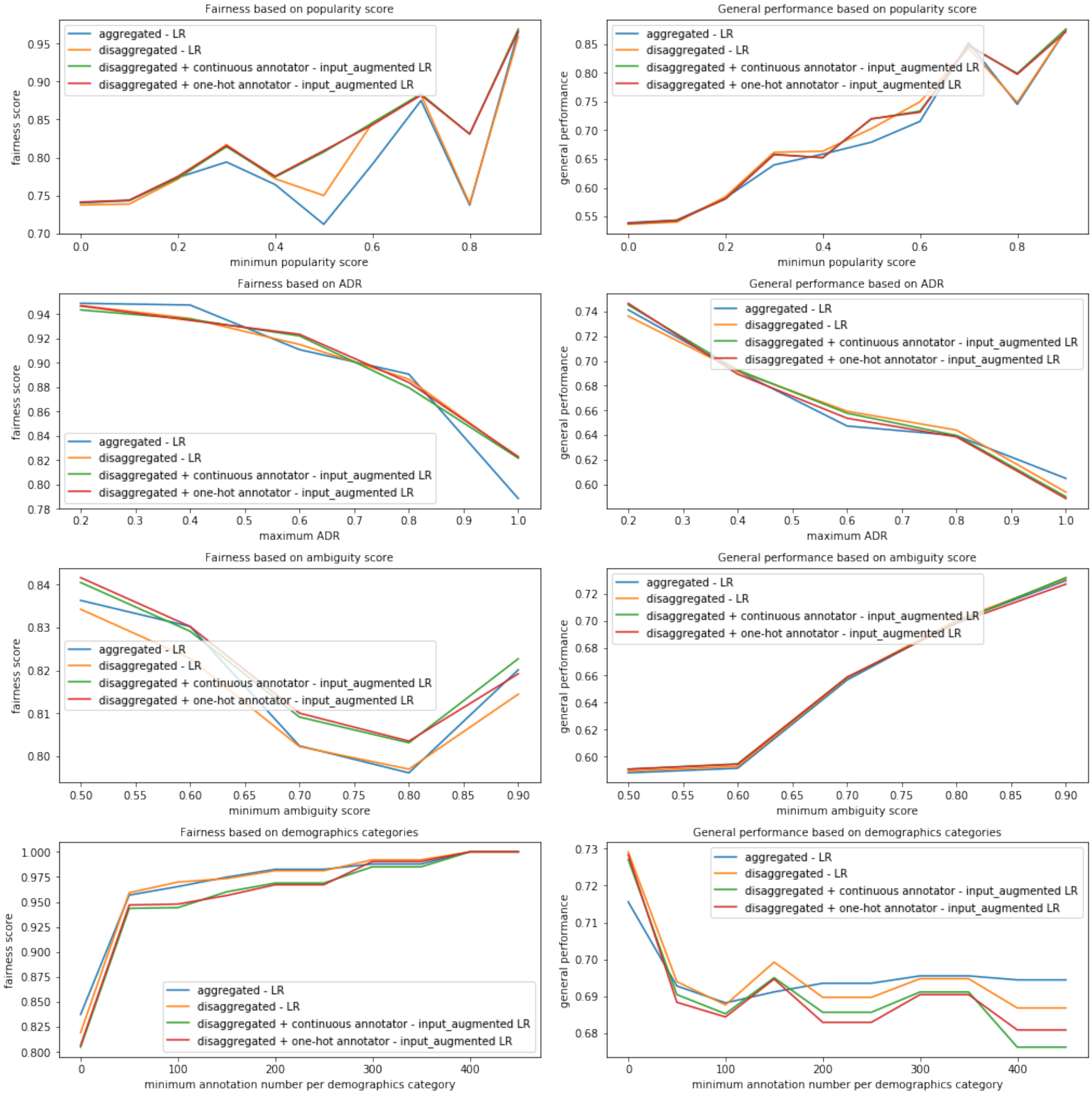


Figure B.8: Comparison of the performance of the 4 models, with variation of the disagreement rate in the test set. (number of demographics categories = [64 17, 13, 8, 5, 5, 3, 3, 2, 2].)
For datasets containing disagreement, the models adapted to the different users perform better on each algorithmic fairness aspect than the models which do not use a users' model.

C

Experimentations on the Machine Learning and Deep Learning algorithms

C.1. Evaluation of the baselines

To verify the efficiency of the algorithms, we compare the results of our implementation on the Jigsaw dataset to the performance given in the literature. We divide the Jigsaw dataset into a training (80% of the data) and test set (20% of the data) and respectively train and test the algorithms on these sets. Similarly to the Jigsaw team [111], we compute for each of the models the standard 2 class area under the receiver operating characteristic curve (AUC), and the Spearman rank correlation. The AUC is evaluated between the models' predicted probability that the comment is toxic and the majority-vote label (MV) of the comment. The Spearman rank correlation is computed between the models' predicted probability of being a toxic comment and the fraction of annotators who considered it as toxic (called ED label in the tables). Additionally, we evaluate the models' performance using traditional metrics (precision, recall, F1-score and accuracy) using the majority-vote labels as ground truth.

C.1.1. Machine Learning

We present the performance of the Logistic Regression classifier in Table C.1 trained with the binary labels aggregated by majority voting.

	$word-embed.$		$char.-embed.$	
	$train$	$test$	$train$	$test$
precision				
$total$	0.9368	0.9198	0.8407	0.8420
$class0$	0.9599	0.9583	0.9430	0.9435
$class1$	0.9368	0.9198	0.8407	0.8420
recall				
$total$	0.6112	0.5906	0.4396	0.4390
$class0$	0.9956	0.9946	0.9911	0.9913
$class1$	0.6112	0.5906	0.4396	0.4390
F1-score				
$total$	0.7398	0.7193	0.5773	0.5771
$class0$	0.9774	0.9761	0.9664	0.9668
$class1$	0.7398	0.7193	0.5773	0.5771
accuracy	0.9585	0.9559	0.9378	0.9385

Table C.1: Logistic Regression classifier performances (metrics in row) on the training and test sets, trained with two different feature vector types (word-embedding and character-embedding). The performances are given on the full data as well as computed on a class level (class0: non-toxic, class1: toxic).

With the Jigsaw toxicity dataset, we draw the same conclusions as the Jigsaw team which reported results on the aggressiveness dataset. The order of magnitude of the performance that we find is the same as the performance reported in the Jigsaw paper [111], while the performance are higher than the ones reported in [50] but they used a different dataset. Besides, we notice that using the character-level features leads to higher performance than the performance with word-level features, which is probably because character-level features have a higher level of details.

C.1.2. Evaluation of other classifiers

We evaluated two additional models identified in the literature review. We trained the Multi-Layer Perceptron and the LSTM Recurrent Neural Network presented in [50] with the Word2Vec data embedding on three different types of training data: majority-vote labels (MV), ED-aggregated labels (ED) and the disaggregated annotations (DA). The ground truth used to evaluate the algorithms is the majority-vote labels (and ED labels to compute the Spearman correlation).

We find higher performance than in the paper [50] but the performance are computed on different datasets. As it was shown for the traditional Machine Learning algorithms by the Jigsaw team, training the algorithms on the ED data increase the performance. On the contrary, training the neural networks on the disaggregated annotations decreases the performance, probably because it "confuses" the algorithm which does not specifically learn the majority-vote label. Training the same neural network on a dataset with balanced classes decreases the performance probably because it reduces the number of training data available too much. Using three labels for training also does not help the algorithm to learn, the performance are lower than for the initial set-up, probably because there are not enough data to learn the three labels accurately.

These two models would merit being investigated in future work because they present higher performance than the Logistic Regression.

C.2. Hypotheses to mitigate unfairness

C.2.1. Additional hypotheses for future work

Transformation of the input samples using the annotators' unique identifiers

As seen in the literature, research is done to learn matrices to transform the inputs of neural networks depending on an additional variable related to the input. In our case we formulate the following hypothesis: ***learning annotator-specific parameters to transform the inputs of traditional algorithms enables the algorithms to output the different opinions on one same sample. (H5)***

Precisely, we propose the following modification of the input. The input is usually a sentence s_i and possibly an information about the person who is annotating the sentence a_j. The sentence is a succession of words $s_i = [w_1, w_2, ..., w_N]$. Each word w_k is represented with a continuous representation (a word embedding is chosen) so that $w_k \in \mathbb{R}^d$, d being the dimension of the word representation. We represent the annotator as a matrix such as $a_j = A_j \in \mathbb{R}^{d \times d}$. As explained in [104], A_j is too large in practice even if $d = 50$ only. Thus, we proceed to the same decomposition as they perform: $A_j = A_{j1} \times A_{j2} + diag(a^*)$ with $A_{j1} \in \mathbb{R}^{d \times r}$, $A_{j2} \in \mathbb{R}^{r \times d}$, $diag(a^*) \in \mathbb{R}^d$ a diagonal matrix common to each annotator, and r a chosen dimension with $r < d$. Finally, we transform each word w_k for the specific annotator a_j with the following calculation in a new word t_{kj} such that $t_{kj} = tanh(A_j \times w_k) = tanh((A_{j1} \times A_{j2} + diag(a^*)) \times w_k)$.

These new inputs are fed to the Deep Learning model presented in the previous section. To learn the matrices A_j, they are initialized with specific values or randomly and they are trained during the training process of the model by backpropagation. We use regularization with the norm of A_{j1}, A_{j2}, $diag(a^*)$ in the loss function. In [104], the hyperparameters are set to the following dimensions: $d = 100$, $r = 3$.

Running the experiments explained in Chapter 5, we quickly run into memory issues when training the model with over 3000 different users. Therefore we propose to group the users into bins and learn matrices specific to these groups instead of to the individual users.

This model should perform better on the known users with many example judgements since it does not require specific information (which might not be enough) to return output adapted to each user. However, for unknown users it is not able to return user-adapted judgements but only makes use of the general diagonal matrix to transform the inputs. Moreover it is more memory- and time- consuming to train it compared to the previously presented algorithms.

Combination of the augmented features and annotators' identifiers

Seeing the drawbacks of the previous two propositions, we propose to combine them to avoid each of their disadvantages by balancing them with the other model advantages. Therefore, we make the following hypothesis: ***combining the two above hypotheses (input-augmentation with users' demographic information***

(H2) and input-transformation with users' identifiers (H5) enables the algorithms to output the different opinions. (H6). This model should enable to generalize to 1) unseen users better than when only using the users' input transformation matrices, and to 2) users without demographic information better than when only using this information. However it remains memory- and time- consuming to train and evaluate.

Regularizing the loss function with the fairness measures

The parameters of the Machine Learning models are optimized by minimizing the value of a loss function. The loss function is parametrized by these parameters, and consists in a computation over the training data samples and labels. For example the Logistic Regression measures the gap between the predicted labels (computed by the model) and expected labels of the data samples in the training set, and the Support Vector Machine makes computations involving both the data samples and labels. During the training process, we could possibly add the weighted sum of the fairness values computed by the model over the training data as a regularization parameter in the loss function. This would enable to take into account fairness when optimizing the parameters of the model, and not only the performance accuracy. As it takes time to implement the new loss function in the different classifiers, this is out of scope of the thesis project but it could be a future work to experiment on.

C.2.2. Summary of the hypotheses and experiments

We summarize in Table C.2 the hypotheses made with the purpose to make Machine Learning models fairer.

Pipeline step	Identifier	Hypothesis
Data aggregation	H1	Using the disaggregated annotations instead of the aggregated labels to train the models make them fairer since they would otherwise not be able to learn the judgements of the different users.
Model architecture	H2	Adding as input to traditional classifiers the annotators' demographic information that psychology literature defines as influencing variables for toxicity perception enables the models to output the judgements of each annotators of the samples. Annotators' demographic information can be encoded with the following methods:
	H2.1	*One-hot encoding of the three variables and concatenation of these 3 representations.*
	H2.2	*Continuous representation (e.g. between [0;1]) of each variable according to the available variable ranges in the dataset and concatenation of these 3 representations.*
	H5	Learning annotator-specific parameters to transform the inputs of traditional algorithms enables the models to output the different judgements on one same sample.
	H6	Combining hypotheses (H2) (input-augmentation with users' demographic information) and (H3) (input-transformation with users' identifiers) enables the models to output the different judgements.
Model training process	H3	Tuning the hyperparameters of the models with grid search using the fairness measures as the performance metric to optimize, and choosing the number of data features and training data according to the learning and feature curves plotted using the fairness measures increase the fairness of the models compared to models whose hyperparameters and training data are chosen using the classical performance metrics.
Dataset	H4	Balancing the training dataset over one of the clustering criteria used to study the multiple fairness aspects increases the fairness performance of the models trained with this resampled dataset.

Table C.2: Summary of the hypotheses to make the models fairer

There are many experiments to run considering that we selected several Machine Learning algorithms to investigate at the beginning of the chapter. We summarize in Table C.3 the experiments that we should proceed with.

Dataset distribution		Annotation aggregation	Machine Learning model		Grid search metrics
clustering criteria	*resampling*		*classifier*	*input type*	
ADR	balanced	aggregated with majority-vote	Logistic Regression	sample features	accuracy
AS	original	disaggregated	Support Vector Machine	sample features + one-hot demographic	combined fairness
AP			Multi-Layer Perceptron	sample features + continuous demographic	
demographic			neural network	sample features + annotators' identifiers	

Table C.3: Theoretical list of experiments.

C.3. Determination of the grid search performance metric

To get an idea of how the different performance metrics available (accuracy, F1-score, precision, recall, dispersion and general performance with the four fairness aspects) would influence the choice of the hyperparameters during grid search, we plotted in Fig. C.1 the performance measured during grid search of the hyperparameters of the Logistic Regression trained on disaggregated data with additional one-hot encoded demographic inputs. We noticed that accuracy, F1-score, precision and recall point out to similar hyperparameters ; the dispersion side of the fairness metrics show similar performance trends and the general performance side of the fairness metrics generally exhibits opposite trends. These similarities lead us to only study the accuracy as a baseline against the hypothesis 3, and to only select the fairness measures based on the annotators' Average Disagreement Rate (ADR) with the majority-vote as a base for H3's computations (it is also the main aspect of fairness).

The dispersion and general performance results found with the annotators' ADR would lead to very different choices of hyperparameters. We aim at increasing the fairness of the models but the models' accuracy should not be close to zero because they would not have any use. Consequently we combine both measures to perform the selection of hyperparameters. There are different ways to compute the mean of two values:

- Average mean.
- Harmonic mean.
- Average of the differences of the measured performance to the average performance over the grid. This could enable to quantify how much improvement or decrease compared to the average each hyperparameter enables.
- The dispersion and general performance metrics do not represent the same notions although they are both in range [0;1]. Consequently, it might not be meaningful to combine their values by using the mean. In order to get values independent of the notion represented by the metric, we propose to transform the values in each grid with the following computations. After computing the average mean and standard deviation of the grid measures, we calculate for each grid value the number of standard deviation in the absolute value difference between the value and the mean ($\frac{x_{value}-\mu}{\sigma}$ with x_{value} the grid value, μ the grid mean and σ the grid standard deviation). The number of standard deviation is independent of the studied metric.
  We investigate the following variants to combine the transformed values:
  - Average of the two values. Since the values are numbers without units, adding them together gives meaningful results.
  - Normalization of the values by dividing them with the maximum (computed with absolute values) number of standard deviation in the grid, and average of the values. The normalization might make the dispersion and general performance values more comparable in case they generally have very different standard deviations.
  - Weighted average of the normalized number of standard deviation. Several weights can be tested to give more importance to the dispersion or general performance aspect of the metric.

We show in Fig. C.2 the results obtained with these different combinations of the dispersion and general performance measures. We observe that the results of the first three propositions exhibit the same trends but that the values computed with the harmonic mean exhibit larger ranges of variations than the others. The variants of the last proposition also show similar trends except when the weights of the weighted average are too close to 0 or 1 because only one of the two metric is influencing the results. It is hard to define what the best combination is because it depends mainly on which aspect (dispersion or general performance) we prioritize. We decide to use the last proposed combination of values with a weight of $\frac{2}{5}$ for the dispersion and $\frac{3}{5}$ for the general performance because further experiments proved these weights adapted not to choose hyperparameters which return models with high fairness but very low accuracy.

We additionally plotted separately the grid search results for the combination of the two fairness measures on the four different aspects of fairness. The hyperparameters selected with the four different aspects are all different. Consequently, as future work we could investigate the combination of several aspects of fairness to select the hyperparameters during cross-validation, in order to take into account all these aspects by order of preference.

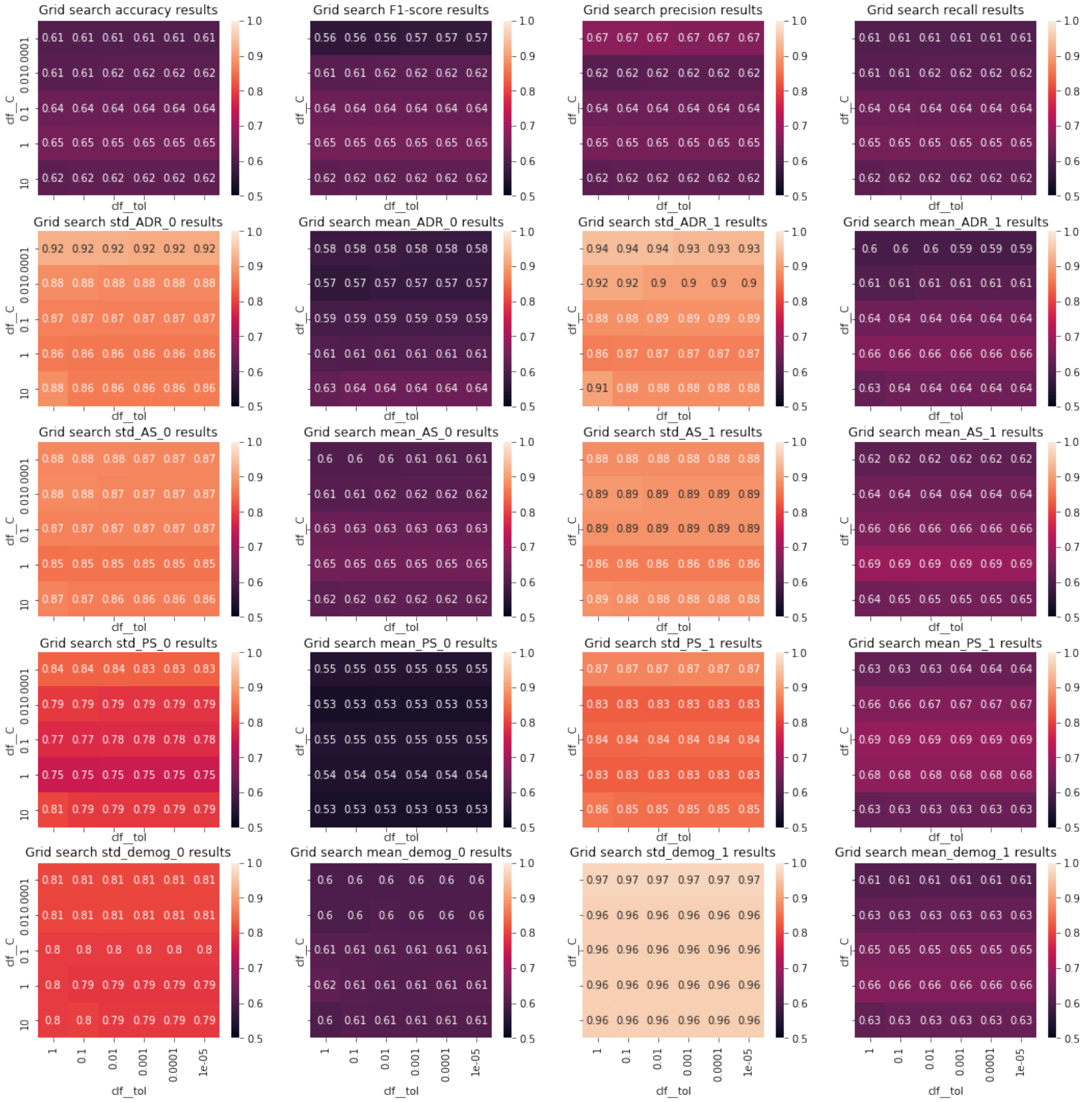


Figure C.1: Grid search over the hyperparameters of the Logistic Regression, model evaluated with the usual performance metrics. The different metrics conduct to the selection of very different hyperparameters. However, the influence of the hyperparameters on the performance of the models is shown later to be small.

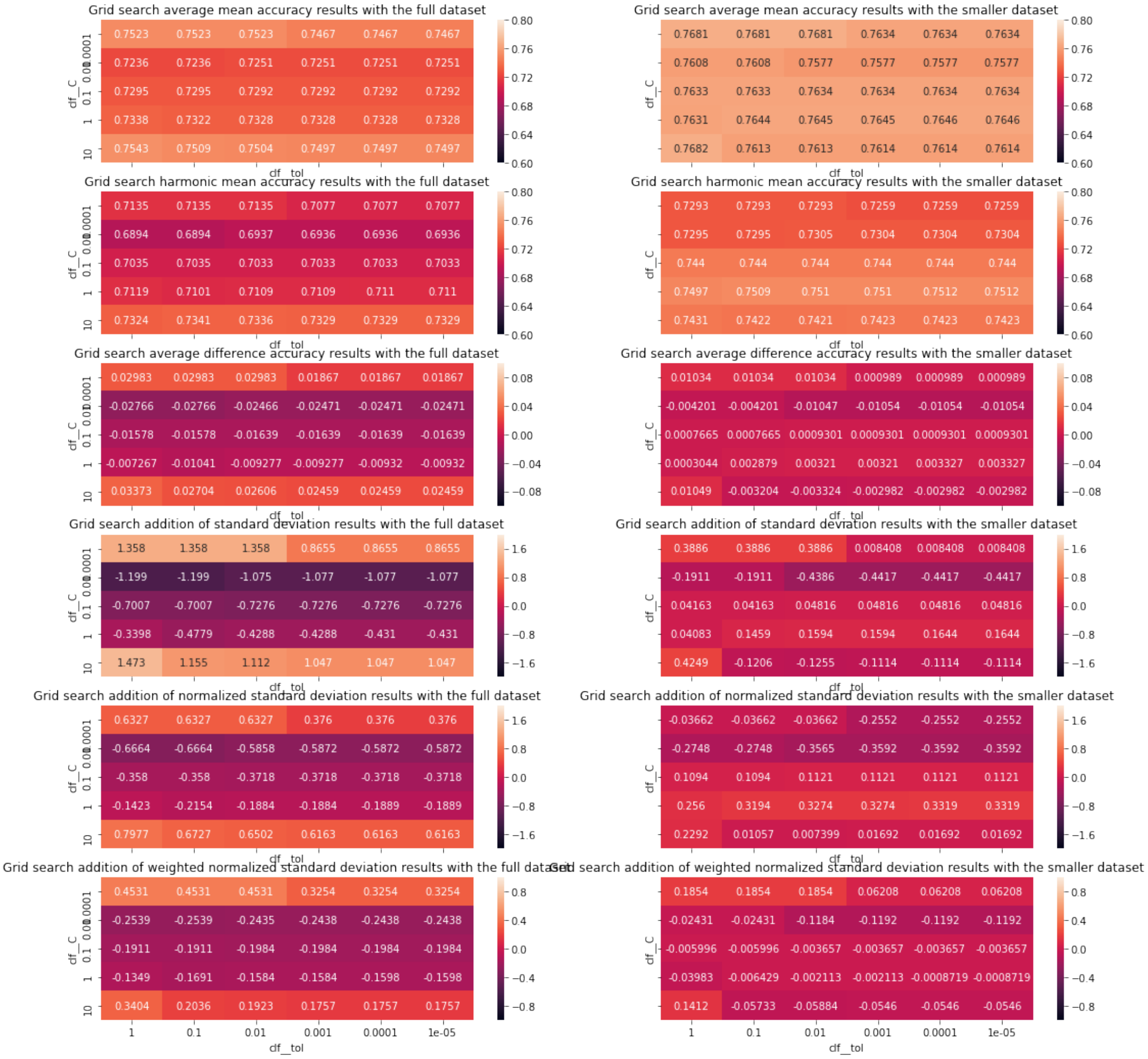


Figure C.2: Combinations of the dispersion and general performance values based on the annotators' Average Disagreement Rate with the majority-vote, on the results of the grid search over the hyperparameters of the Logistic Regression.
The different combinations lead to the selection of very different sets of hyperparameters.

C.4. Method to resample the dataset

The pseudo-code is given in Algorithm C.1. First we define the total number of annotations N_a that each fold should contain, by maximizing this number considering the annotations and their associated clustering criteria values available. Some demographic categories have very few annotations, we decide to remove the less frequent categories in order to obtain a large enough dataset and we keep the 39 largest demographic populations (populations for which the number of toxic and non-toxic annotations is above 500). We choose $N_a = 1029$ because we estimate that this should enable to constitute large enough training and test sets since 9 times this quantity of annotations are used in the training set. Then we resample the datasets according to the different hypotheses so as to have N_a annotations per new dataset.

Algorithm C.1 Dataset resampling

```
function COMPUTE_DISTRIBUTION($dataset, clustering\_criterion$)
    if clustering_criterion = demographic then
        $dataset$ ← remove less frequent demographic populations.
        $dataset\_clusters$ ← cluster dataset's annotations according to the annotator's demographic.
    else
        $dataset\_clusters$ ← cluster dataset's annotations on $cluster\_criterion$ into 5 bins.
    end if
    $dataset\_clusters$ ← divide bins of $dataset\_clusters$ into 2 bins for the 2 classes.
    return $distribution$ ← compute the distribution of the bins in $dataset\_clusters$.
end function

function COMPUTE_FOLD_SIZE($dataset$)
    $min\_bin\_size\_list$ ← empty list.
    for $clustering\_criterion$ in list of clustering criteria do
        $distribution$ ← COMPUTE_DISTRIBUTION($dataset, clustering\_criterion$).
        $min\_bin\_size\_list$ ← append minimum bin size of $distribution$ among all the bins.
    end for
    return $\frac{minimum(min\_bin\_size\_list)}{10}$
end function

function ORIGINAL_DISTRIBUTION_RESAMPLING($dataset, clustering\_criterion, fold\_size$)
    $distribution$ ← COMPUTE_DISTRIBUTION($dataset, clustering\_criterion$).
    $total\_available\_data$ ← sum values of $distribution$.
    $list\_data\_fold$ ← empty list.
    for $cell$ in $distribution$ do
        $number\_data\_fold \leftarrow size(cell) * \frac{fold\_size}{total\_available\_data}$.
        for $fold\_id$ in range(10) do
            $data\_fold$ ← get $number\_data\_fold$ from $cell$.
            $cell$ ← remove $data\_fold$ from $cell$.
            $list\_data\_fold$ ← append $data\_fold$.
        end for
    end for
    return $list\_data\_fold$.
end function

function BALANCED_DISTRIBUTION_RESAMPLING($dataset, clustering\_criterion, fold\_size$)
    $distribution$ ← COMPUTE_DISTRIBUTION($dataset, clustering\_criterion$).
    if clustering_criterion = demographic then
        $number\_data\_fold \leftarrow \frac{fold\_size}{number\_demographic*2}$
    else
        $number\_data\_fold \leftarrow \frac{fold\_size}{10}$
    end if
    $list\_data\_fold$ ← empty list.
    for $cell$ in $distribution$ do
        for $fold\_id$ in range(10) do
            $data\_fold$ ← get $number\_data\_fold$ from $cell$.
            $cell$ ← remove $data\_fold$ from $cell$.
            $list\_data\_fold$ ← append $data\_fold$.
        end for
    end for
    return $list\_data\_fold$.
end function
```

C.5. Examples of correct and incorrect classifications

We give examples (annotation: 0 = non-toxic, 1 = toxic) of high-consensus data which are:

- well predicted by the model without user model (M1) but not by the model which uses the one-hot encoded users' model (M2):

comment	annotation
added tags files help editors know whats going thanks	0
good call thanks	0

- well predicted by M2 but not M1:

comment	annotation
note need member project order edit page whether edit minor per wikipedia bold et al	0
fairly certain henrik swedish thus worth exposing parents however	0
clarification zundark right checked wikipedia bugs page first bug code makes wikipedia work means line code may error small extra space analogous simplified way trying make something bold html forgetting put end see something like words bold instead words bold like virus code somebody deliberately wrote order infect computer damage files go around jhk	0
elected electoral jhk	0
template inuse check templates template inuse use top articles working avert speedies edit conflicts	0

- incorrectly predicted by M1 and M2:

comment	annotation
really think much historic gnosticism	0

- well predicted by M1 and M2:

comment	annotation
september 2010 please stop adding unreferenced controversial content biographical articles content nature could regarded defamatory violation wikipedia policy continue blocked editing wikipedia	0
clarification zundark right checked wikipedia bugs page first bug code makes wikipedia work means line code may error small extra space analogous simplified way trying make something bold html forgetting put end see something like words bold instead words bold like virus code somebody deliberately wrote order infect computer damage files go around jhk	0

We give examples of low-consensus data which are:

- well predicted by the model without user model M1 but not by the OH model M2:

comment	annotation
hey please stop vandalizing stop	1
please stop continue vandalize wikipedia blocked editing shared ip address make unconstructive edits consider creating account avoid irrelevant warnings	1
warning honesty warning person like get	1
apology necessary probably dislike	1
knew better would know bullshit gets published people like try delete truth	1

- well predicted by M2 but not M1:

comment	annotation
like said countless times today mulatto movement internet mulatto movement lol lack better phrase draws esteem counterproductive denigration black people nothing else put black quotes united states mulatto black necessarily mutually exclusive concepts go far say power self identifying mulattos hope always relation black people mulattos fail realize existence promotes race mixing encouraging 6 8 mandingo date marry impregnate 5 3 mary sue whites matter much cozy try delink blacks innately reject perennially frown upon sure black mulatto men would take advantage mulatto movement sleep many white women possible white women accessible interestingly people helm mulatto movement likely turn quadroons octoroons fascinating hierarchical paradigm followed funny imagine hexadecaroon forming quadroon movement denigrated black people mulatto people well gotta love stuff lol	1
certainly better world service called simple negro months ago sources exist material nollen dr wright	1
fyrom live toghether ethnical point view albanians turks slavs makedonji armanji national point view fyromian macedonians fyromians change flag another sun fyromian sun wich reprezent freedom liberty action makedonji armanji regaine official right use ancient symbol vergina sun sun use makedonji armanji everywhere planet must tell taht still dont know somebody use yhese sun moon another planet dont know universal symbol sun official northern greece ask karolos papulias makedon arman used epir thessaly kavala makedonia thrakia regions live 1 5 milion makedonji armanji speak greek makedon arman nodody planet use vergina sun makedonji armanji also called vlasi valaq cingari cobani aromani pomak maybe tommorow chinese political interest misterjapa dont understand wy dont know cancelling every day ethimology makedonji armanji vergina sun balkans houndred books find names officcccccccciiiiiiiaaaaaaallllliiiii add bibliography someone delete every day respect wiki principles tell mister adyjapan makedonji armânjiljii ancient macedonians romanized millenary distinguished language romanian greeks ethnical point view truth 2 thousands years commun greeks romanians nothing masive imigrantions explosion moscopole gramostea still continue mizerably propaganda stupid tendence assimilation still wikipedia delete native names sincerly tell seleting action ilegal project must stop accept native names symbol go anyway add native names every day	1
trying explain neil armstrong landing moon fictitious santa claus appearing xmas 007 submarine cars dare pollute cyberspace outrageous lies suggest otherwise deadly diseased celebrity liars like armstrong aldrin nixon telling truth young impressionable children looked apollo wikipedia see nothing deadly mistaken rehashing late 60 hoaxes presented scientific achievements	1
dont think equally bad muslims rioting destroying property islam religion peace tolerance acts violence equally reprehensible islam cartoons realize world watching act like muslims like undisciplined individuals	1

- well predicted by M1 and M2:

comment	annotation
bother evidently feel examples disgusting attitude towards others nothing harassment offending individual	1
april 2008 please stop continue vandalize wikipedia blocked editing ip using similar another used time vandalise page	1
vandalislisilisilism gee thanks	1
oh sorry must watch straaaangle death omid djalili mummy 1999	1
game year wolf need stop changing blood moon year wolf studio album compilation album game said many interviews album 4 tracks game thanks dont tell cite source cited shit either	0

- incorrectly predicted by M1 and M2:

comment	annotation
arg shit credible	0
ok please explain page newspaper called antelope valley press notable page newspaper called av news publications cover area los angeles county allowing stay pasted deleted content first attempt create page deleted thought might format issue copied entire av press page template replaced text av press text av news website link thinking would resolve issue use outrage page deleted almost posted worded better work av news asked see getting page others think people looking wiki aware media outlets area av press sure references need bloody website perhaps could head ass attitude give example two lord god almighty bwilkins finds acceptable question wiki getting information playing favorites much av press donate wonderous exclusivity simply see reasonable explanation av news keeps getting deleted	1
email sent admin blocked ip permanently without explanation claimed ip going wild trying avoid scrutiny nonsense first done anything could remotely considered going wild edint articles making badly needed edits even tried identify edits considered objectionable made attacking statement expect others believe second certainly trying aovid scrutiny signing everyone certainly knows ip using means lots people watching third claim ani uncivil fourth claim edit warring edit warring took place made new edits revert one two blind reverts changes made people clearly ignoring talk page discussion putting back wanted edit comments explaining disagreed reverts went reverted would ones edit warring revert certainly use one revert without abusive edit warring fifth claim ani page log make edits well ok rules saying anyone sign sign edit blocked ip address use edit ip blocks account well given justification block therefore expect take block would nice apologize flagrant violation several wikipedia policies	1
assume good faith editor clearly intentionally causing trouble thanks waste time	1
problem mate professional registered website true story ronald ryan almost completed purrum able contribute trash lies allegations accusations opinions views ryan case purrum infamous promotional book profits hanged man pages 221 222 confirms purrum compulsive manipulative liar fact book states discrepencies eyewitnesses evidence substancial wide ranging fourteen eyewitnesses testified different accounts saw eyewitnesses testified seeing ryan east hodson eyewitnesses testified seeing ryan west hodson eyewitnesses testified seeing smoke coming ryan rifle although established expert forensic ballistics senior sergeant colin letherbarrow testified cartridges used smokeless variety eleven eyewitnesses testified saw ryan armed rifle fourteen eyewitnesses testified hearing one single shot hodson fell ground significantly four eyewitnesses testified actually seeing ryan fire shot contest evidence eyewitness contradictory purrum infamous book goes describe discrepencies prosecution case eg lack scientific forensic evidence vital missing pieces evidence downward trajectory angle fatal bullet prison officer paterson testifying fired single shot heard eyewitnesses thank good luck ongoing lies	1

D

Ethical debate related to the computational automation of hate speech detection frameworks

This chapter is an adapted version of the final essay written for the course UD2010 Critical Reflection on Technology (course part of the TU Delft Honours Program), adapted to the thesis report.

Abstract

Recently, Machine Learning and the study of social Big Data have been brought together by computer scientists to discover correlations and make inferences on new human-related data such as sentiment classification of comments posted on the Web, ... Among these research, we are interested here in algorithms performing automatic classification of Web sentences into classes like hateful or not hateful, or offensive or not offensive. Even though human workers are already performing such tasks to filter posts on social media and comments on forums, the new development of these computational systems sees many criticisms arising. We discuss the arguments of the opponents to this technology in two steps: first the issues related to the justifiability of the technology implementation and its possible limits are investigated, and then the potential negative consequences of applying such a technology.We argue that this technology is not harmful by itself, but the applications that people could find to it could be, and therefore it should be adopted only under certain conditions that are drafted in the last section.

D.1. Introduction

In Computer Science, two fields of research have recently been brought together: Machine Learning -the development of algorithms to automatically classify data samples-, and the study of social big data -the large amount of social media data produced everyday by people's interactions with the Internet and now available to the company workers. Combining these two areas, the aim of researchers is to discover correlations in the data and classify automatically previously unseen data, such as sentiment classification of comments posted on the Web, recognition of objects in social media post pictures, ... One of these technologies is the automatized classification of Web sentences into different classes related to human judgment: hateful or not hateful, or toxic or not toxic, or offensive or not offensive (these classes have similar definitions that the researchers do not differentiate clearly) [94]. Several studies show the desirability of detecting hate speech on the Internet [105]. Mainly with the increasing use of the Internet and websites where comments are enabled, the quantity of posted messages is too large for human moderators alone to filter them. Thus, it becomes more and more important to be able to detect hateful comments automatically.

The development of this technology is very recent, but already criticized by many. First, its opponents claim that sentence toxicity classification can not be justified mainly because the technology would not accurately represent people's opinion since several persons may have different views on one sentence. Second, even if we consider that the technology classifies in an acceptable way, its negative consequences would outweigh the positive ones: it would be used to filter Web's content and thus the accessible information would be biased towards certain opinions considered as morally correct by a majority of people or by the creators of the algorithms. We argue conversely that this technology is not harmful by itself, but the applications that people could find for it could be, and therefore it should be adopted only under certain conditions. First, we investigate the possible objections to the technology by looking at how it is developed, after explaining how it works. Then, we reflect on its intended purpose. Finally, we propose rules to make the technology less controversial.

D.2. Description of the technology

The technology is a system which takes as input a sentence, and returns a judgment over the toxicity, offensiveness or hatefulness of the sentence (to simplify, we mention toxicity in the rest of the essay but we mean any of these categories). Two main versions of this technology are considered. The first one is the currently most accurate one: the output is a binary label saying whether the sample is toxic or not, or a percentage estimating for what proportion of the population the sentence would be considered toxic. However, a newly emerging research proposition (version 2) is to adapt these systems to each specific reader of the sentence: instead of returning a label corresponding to the majority's opinion, the output judgment would be different for each reader.

In order to build current systems, it is first required to build a dataset consisting in example sentences and their toxicity judgment or judgment percentage. Then, a mathematical classification model is designed and implemented. This model is then trained to solve its intended task by being fed with the dataset multiple times and adapting its parameters to it.

If the technology is adapted to each individual, the dataset contains example sentences and their corresponding judgments by multiple persons. Then, the model is trained by inputting the sentences, their judgments, the identifiers of the individuals, and possibly features describing them such as their age, gender, education level, ethnicity. These features are investigated to help the classification by the algorithms because the psychology literature [32] about offensiveness and hatefulness perception declares them as major variables in the judgments. It is hypothesized that the more properties characterizing the individuals are available, the more accurate the algorithms would be at outputting the individuals' judgments.

D.3. On the ethics of the system pipeline

In this section, we investigate what are the possible implementation's aspects which could be considered harmful by looking at each step of the technology pipeline, and reflect on whether they are avoidable.

D.3.1. On the possible breach of privacy

There are several issues with the dataset, the first one being privacy. Most researchers create their dataset by scraping sentences on the Web from social media posts, chat histories, users' flagged posts as undesirable content... Besides, certain researchers link the sentences' users to their social media profile to collect additional information about them like their gender, nationality, etc... Certain persons argue that storing their data is an invasion of privacy. However, these posts and profiles are available publicly on the Internet because the users previously decided to publish them. Therefore, it is debatable whether this is truly an infringement of privacy. When creating crowdsourcing tasks to collect toxicity annotations on the sentences, the researchers may ask the workers to give personal information such as their age, gender, ... This is sometimes considered an infringement of privacy whereas the workers are not forced to give this information.

Moreover, it depends on how the data are used. If they are made anonymous, the users are not identifiable, what is an additional reason not to consider it as breach of privacy. Researchers could also store the data only for the time spent exploiting them and then delete them, so that there is no permanent trace of the users. Finally, if the users wish to benefit from the individual-level tuned algorithms, they would be made aware of the necessity of accessing their personal data and asked beforehand to share them, consequently the data of the unwilling users would remain unused. If not sufficient, an additional acceptation could be required from the social media users (in addition to the usually ignored terms of use of the social media platforms) to make them more conscious of their public data being accessed.

D.3.2. On the justifiability of sentence hatefulness classification

The second issue is related to the actual content of the data and its potential discriminative power. As highlighted in the new Machine Learning conference FAT* [1] (Fairness, Accountability, and Transparency), datasets may contain implicit biases which turn the algorithms into discriminative systems [**?**]. If the system is adapted to each individual but these individuals' opinions are not represented equally in the dataset, the algorithm may perform more accurately for certain people than others, and thus it could be considered fairer regarding specific individuals. Is this inequality an issue? The system cannot be perfect and errors may have more or less consequences, but claiming the system itself or its creators are discriminative sounds absurd as long as it is not intended. Besides, the more data available, the more these issues can be avoided.

Moreover, the second version of the system aims at returning outputs tuned to each individual by generalizing over data describing only a group of individuals. That questions the conception of the human mind that researchers implicitly have when designing these systems: every individual can be assimilated to a group of representative persons in the dataset, and can be described by a few features. However, if researchers did not make this first assumption, they would not be able to create a system serving their goal, so this is unavoidable. The second assumption that a group of features would enable to learn individuals' preferences is debatable. For example, more personal variables also have an influence in the offensiveness judgments. Thus, only using the first group of properties can be considered discriminatory since people are seen simply as members of specific demographics groups. A solution is to explain that the algorithms are made to classify toxicity for categories of users instead of individual users, or to evaluate the algorithms and display warnings about the possible inaccuracies. Moreover, if researchers manage to create a system with high accuracy, their method would be proved legitimate, but it implies that they first have to test their assumption.

Certain persons could object there is no objective rule to define when the system is effective enough to be used. There is no criterion to choose an accuracy threshold to reach to use the system. It is also impossible to claim that the data contain every kind of opinion, and thus that the algorithm is evaluated against every possible configuration because sentence toxicity perception depends on many parameters of the sentence context -not only who judges it but also what was the aim of the person writing it, to whom it was addressed... However, at least the most frequent opinions are represented in the dataset. If the limits and characteristics of the dataset used are explained to the users of the algorithm, they would be aware of where errors are possible and the limitations of the algorithm.

Finally, many claim that the decision process is not explicit, so the outputs are not verifiable and consequently should not be trusted and used. However, even for human thinking not every decision can be explained rationally, as Daniel Dennett says "We also don't know how we take decisions". Thus this objection does not hold, and the rational could be claimed here to be the statistics behind the system.

We presented the objections related to the practical implementation of the technology, and showed that with certain conditions on the development and use of the technology they do not hold.

D.4. On the justification of the technology applications

After explaining the advantages to build such a technology, we tackle the opposing arguments related to its purpose by examining the different stakeholders' views (Internet users, websites' companies, and researchers and engineers who create and build the technologies).

D.4.1. On the usefulness of the technology

The advantages are directed towards the users. The first application is the filtering of human content on the Web. Hateful speech filtering is an important task according to the regulations established in certain countries, for example in Germany hate speech was spread over the Internet to influence the elections' outcome [2] and regulations are enforced to have companies remove these posts [3]. Posts on the Internet (on forums or social media such as Facebook) are already filtered by humans reading each post and emitting a judgment according to criteria defined by people responsible for the application. However, facing the increasing number of Internet users, human filterers are not able anymore to process all the posts, which makes automa-

[1] https://fatconference.org/

[2] http://www.spiegel.de/international/germany/trolls-in-germany-right-wing-extremists-stir-internet-hate-a-1166778.html

[3] http://www.bbc.com/news/technology-42510868

tized classification systems useful since they can be used to quickly identify the offensive posts, as Instagram started recently [4].

Moreover, a system can be argued to provide impartial judgments since it always has the same behavior, whereas human judgments are subjective even if the filtering criteria is made as explicit as possible -several human judges may have different views and even one unique judge perception may vary over time. Additionally, the content can be filtered based on different criteria such as differentiating content adapted to children or to different Internet communities of readers for example with different political views.

Another similar use is to advise Internet users: when they write a post, the system can be used to warn them whether their text may be perceived offensive and for which part of the population.

Finally, the filtering could also serve as pre-processing of Web social data for the training of further computational systems. Certain systems like Microsoft chatbot Tay [82] are trained by collecting and feeding them Internet posts. If the content is not filtered, the systems may return undesirable outputs such as Tay which became racist in less than one day.

Thus, the algorithms have many advantages but one could argue they cannot be totally accurate and might not be useful. We object that they can be designed to make more errors on false negatives or false positives. When tuned to minimize the false negatives, the errors are not harmful: if too few posts are removed, the users can simply ignore the toxic comments. When tuned to minimize the false positives, human judges could be required to read and filter only the positive outputs of the algorithm, what would considerably reduce their amount of work and enable not to remove non-toxic statements.

D.4.2. On the criticisms of the technology applications

Now that we have stated the foreseen uses of the technology, we examine the criticisms about the purpose of the algorithm. We consider a system which would work perfectly in order not to confuse with implementation issues. Although data filtering appears to be an advantage, its morality merits to be investigated. For websites' users, the reduced access to information and the selection of the accessible information are two main issues, even though human filtering brings the same debate.

A first question is whether abusive content should be filtered? Abusive speech has been used constantly for example in political campaigns far before the Internet was invented, but has not always been forbidden. So, why are people suddenly interested in filtering it on the Web? Possibly, because of their anonymity on the Web people are more carefree and abusive what hinders peaceful use of the Web as a mean of information access and communication, and that would legitimate setting up barriers to prevent abuse. However, should freedom of expression be limited? For certain persons, netizens should be able to express themselves without using abusive language and although there is freedom of expression, people are free up until a certain extent, and therefore should not be offensive. That is what supports the laws against racist and negationist statements in the public debate in several countries and on the Web with the Council of Europe cybercrime convention. For others, freedom of expression is more important than offending others and therefore these laws are not justified and no comment should be deleted -these persons could simply not use the filter. From the point of view of the persons posting the data, removing their content is also equal to negating their freedom of expression if we do not believe the laws are justified. Having the polemic statements available on the websites with the possibility for users to hide them could be a solution: the posts would remain there, but would only not be read by the people who decide not to have them displayed.

Second, filtering the available content on the Internet could have unwanted consequences. It may decrease the amount of information and opinions Internet users are exposed to, reducing their reflection and producing one unique way of thinking (filter bubble phenomenon). People would become more close-minded. However, this highly depends on the exact filtering criterion: if only comments using abusive language, hateful speech are filtered, then the Internet users who are able to expose their ideas without diminishing others -even though their ideas go against the majority ideas- would be kept. Consequently, the only issue is when the posts both give information and use hateful speech, there only the use of the algorithm is questionable and comes back to the question of free expression.

Additionally, information selection might lead to a potential hidden censorship: the algorithms' conceptors or their influencers (companies or possibly other stakeholders) could use specific filtering criteria which

[4] https://instagram-press.com/blog/2018/05/01/protecting-our-community-from-bullying-comments-2/

would direct users' opinions toward certain ideologies; but this is already the case with other media like television, radio, ... Moreover, if the algorithm's criteria are made public, there should not be such a danger.

We saw that the filter bubble and censorship resulting from misusing the technology are the two main dangers opposed to its development and use, along a possible breach of free expression. However, we contradicted these arguments by proposing to disclose the technology implementation.

D.5. Possible regulations to make the use of the system more controlled

From the previous sections, it is concluded that the technology itself would not be harmful to its users as long as some rules would be set up to control its inaccuracies and applications. We now explain these regulations.

The doubts concerning the justifiability of sentence toxicity classification and the possible censorship deviation are solvable by making the algorithm pipeline and performances transparent, for people to have the possibility to become aware of and alert about the limits and misuse of the technology they use. Shifting the choice of the filtering criterion from the companies to organizations, by making the criterion universal, would enable to make the algorithms unattainable by possible influencers.

Second, we identified issues concerning the morality of Web content filtering, especially if the system was used over the whole Internet by default -what would be a large scale compared to only implementing it on some websites. We suggest that instead of automatically filtering posts, a mask should be proposed that users could choose to activate, so that they become involved in the decision process and decide consciously to what information they are exposed. For example, it could be used for children but not for their parents if they consider that children should not learn abusive language. Besides, in order not to have users forget the filter and become less critical, a warning should be displayed to remind them about it.

Laws currently require companies to remove hateful content within a few days but certain persons are against this for free speech reasons. Thus the rules could possibly be alleviated and the filtering could be made mandatory only for certain people characterized as "sensitive" such as children, and available as a choice for the others. However the question of defining these sensitive people would be controversial.

Finally, we explained previously that the perception of toxicity is subjective. Having websites apply the algorithm automatically could tend toward a general-judging system not adapted to each individual preferences and thus would ignore the minority's opinions when filtering the data. Thus we believe that if the system was not activated by the application providers but installed by each user who would choose their own filtering parameters, then it could become inclusive of the different opinions.

D.6. Conclusion

In conclusion, although automatic filtering of toxic content on the Web is a useful technology considering the increasing number of possibly abusive Internet users, it is contested by many because it does not seem justifiable to develop an individually-tuned system from general data and it could have several negative consequences. However, we showed that if they want to create such a technology, researchers have no other possibility than to test their algorithm design but also clearly make the users aware of the limitations. Moreover, we argued that the objections to the technology's adoption do not hold if certain regulations are set up to make users employ it consciously.

E
Scientific publication

Following is the paper that was published for the CrowdBias workshop co-located with the sixth AAAI Conference on Human Computation and Crowdsourcing (HCOMP2018).

Characterising and Mitigating Aggregation-Bias in Crowdsourced Toxicity Annotations

Agathe Balayn[1,2], Panagiotis Mavridis[1], Alessandro Bozzon[1], Benjamin Timmermans[2], and Zoltán Szlávik[2]

[1] TU Delft, Web Information Systems
a.m.a.balayn@student.tudelft.nl,{p.mavridis,a.bozzon}@tudelft.nl
[2] IBM Netherlands, Center for Advanced Studies
{b.timmermans,zoltan.szlavik}@nl.ibm.com

Abstract. Training machine learning (ML) models for natural language processing usually requires large amount of data, often acquired through crowdsourcing. The way this data is collected and aggregated can have an effect on the outputs of the trained model such as ignoring the labels which differ from the majority. In this paper we investigate how label aggregation can bias the ML results towards certain data samples and propose a methodology to highlight and mitigate this bias. Although our work is applicable to any kind of label aggregation for data subject to multiple interpretations, we focus on the effects of the bias introduced by majority voting on toxicity prediction over sentences. Our preliminary results point out that we can mitigate the majority-bias and get increased prediction accuracy for the minority opinions if we take into account the different labels from annotators when training adapted models, rather than rely on the aggregated labels.

Keywords: dataset bias · Machine Learning fairness · crowdsourcing · annotation aggregation.

1 Introduction

When using crowdsourcing to gather training data for Machine Learning (ML) algorithms, several workers work with the same input samples and the annotations are aggregated into a unique one like the majority vote (MV) to ensure its correctness (elimination of annotation mistakes and spammers mainly). Although this data collection method is designed to get high-quality data, we expect that certain tasks involving subjectivity such as image aesthetic prediction, hate speech detection, detection of violent video segments, sentence sentiment analysis, cannot be tackled this way: samples should not be described with unique labels only since they are interpretable differently by different persons.

The use of hate/toxic speech has increased with the growth of the Internet [5]. Predicting whether a sentence is toxic is highly subjective because of its multitude of possible interpretations. The sentence "I agree with that and the fact that the article needs cleaning. Some of these paragraphs [..] seem like they were written by 5 year olds." is judged negative or positive by different readers, but this perceptions' diversity is ignored when selecting one unique label as done in recent research [4]. [3] studied the existence of identity term biases resulting from the imbalance of a toxicity dataset content, we show with the example of MV-aggregation that crowdsourcing processing methods on the same dataset also create an algorithmic bias here towards the majority opinion. When annotations differ but are all valid for certain annotators, aggregation loses information and leads to decrease of accuracy and unfairness in ML results, thus we hypothesize that the bias can be mitigated by using disaggregated data. In this study, we first exhibit the presence of the majority-bias and its consequences, then we propose a methodology to expose and counter its algorithmic effects.

2 Majority-biased dataset and consequences

We show on the toxicity dataset [6] that in usual crowdsourcing aggregations of annotations, certain worker contributions are ignored for the majority and that it affects the fairness of ML algorithms' results. The dataset consists of 159686 Wikipedia page comments for which 10 annotations per sample are available. A large number of annotators (4301) that we have their personal information rate the phrases with 5 labels of toxicity ranging from -2 (very toxic) to 2 (very healthy) with 0 being neutral.

Subjectivities in the dataset. For each worker, we compute the average disagreement rate (ADR) with the ground truth (percentage of annotations different from the MV here), and plot the distribution over the dataset after removing the annotations of the lowest quality workers (spammers) (fig. 1). The quality score for each worker (WQS) is computed with the CrowdTruth framework [1] using binary labels ([-2;-1]:toxic, [0;2]:non-toxic), along a unit quality score (UQS) to represent the clarity of each sentence. Without removing low-quality workers, the proportion of high agreement is high because most spammers constantly use one positive label and the dataset is unbalanced with more samples with non-toxic MV. The more possible spammers are removed, the more the disagreement increases until the distributions stabilize. Only 0.09% of the workers always agree with the MV for 50 spammers removed: MV-aggregation is not representative of most individuals but only of a sentence-level common opinion.

Algorithmic effect of the bias. We consider the task of predicting binary labels. Training traditional algorithms to predict the MV, annotations of only maximum 0.09% of annotators would be entirely correct: the majority-bias is not consistent with the worker's individual opinions. We evaluate traditional models (sec. 3) trained and tested on aggregated and disaggregated labels (table 1). In both cases accuracy is higher when measured on aggregated data, what shows that classical input data's treatment makes usual models' predictions biased towards one type of opinion, here the majority opinion, instead of representing each subjectivity.

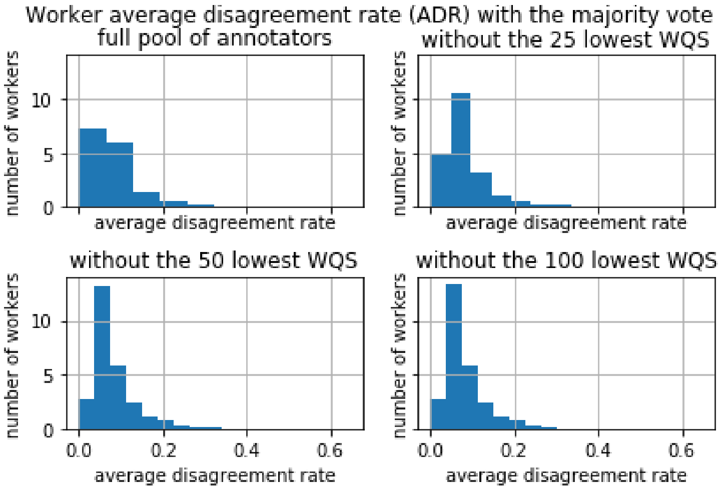


Fig. 1. Normalized distribution comparison of the ADR with the MV with and without low quality worker filtering.

Table 1. Accuracy performances of the model on the ambiguity balanced dataset.

	agg. testing	disagg. testing
agg. training	0.76	0.70
disagg. training	0.77	0.71
disagg. training with user	0.77	0.70

3 Method to measure and mitigate the bias

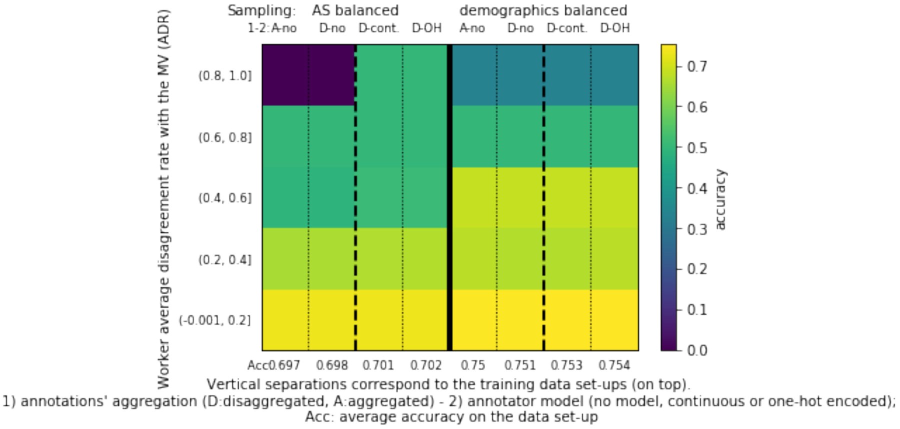
We claim that a fairer algorithm should return different outputs for a same sample depending on its reader. Here, we propose measures of the majority-bias' algorithmic effect and a method to counter its unfairness.

Bias measure. Global metrics are usually used to optimize the algorithms' parameters and evaluate them. However, they do not inform on the bias' effects since most samples' labels have a high-agreement: the slight improvement when training on disaggregated data hints only lightly at label disaggregation (table 1, fig. 2). To identify the effects, we propose to measure sentence-level and worker-level accuracies on the annotations spread in the following bins: we divide the sentences along their ambiguity score (AS) (percentage of agreement in annotations) or UQS, the workers with their ADR, WQS or demographics categories; and also plot histograms of the per-user and per-sentence errors to identify potential unfairness among all workers or sentences.

Bias mitigation: ML. To account for the full range of valid opinions, we propose to modify the inputs to the ML models. After removing low-quality workers, instead of the aggregated labels we feed them with the annotations augmented with the available worker demographics (age, gender, education, with a continuous or one-hot encoded representation) that psychology literature [2] gives as the most influencing factors of offensiveness perception (along with ethnicity not available here). Each (sentence, demographics, annotation) tuple is considered as one data sample. We employ the Logistic Regression (LR) classifier, and encode sentences with term frequency-inverse document frequency (tf-idf). The optimal hyperparameters for each set-up are chosen by performing a grid search.

Bias mitigation: dataset balancing. We define 4 data set-ups to help the algorithms learn the individual annotations. Sentence AS and MV-toxicity are computed, and we resample the dataset following the original distribution or balancing the distribution on these 2 criteria, to obtain a dataset whose majority-bias is decreased by equally representing samples with high and low agreement between workers. We also resample the annotations along the MV-toxicity and demographics categories (removing the least frequent ones) into one dataset following the distributions and a balanced one, to foster performance fairness in-between populations.

Results. Binned metrics like the user-level ADR-binned accuracy (fig. 2 with bins along the y-axis) enable to show that models are more suited to workers who agree with the MV (bottom of the y-axis), and highlight the benefit of using disaggregated data with adapted ML models. On the AS-balanced dataset (left part of the x-axis), the user representation increases accuracy for workers with a high disagreement with the majority over using aggregated data or no user-model. The resampling choice also helps understanding and mitigating bias' effects: balancing on demographics neither clearly shows the performance gap between minority and high-ADR workers nor improves accuracy with the user representation, contrary to the AS dataset in which MV-consensus' presence is reduced.



Vertical separations correspond to the training data set-ups (on top).
1) annotations' aggregation (D:disaggregated, A:aggregated) - 2) annotator model (no model, continuous or one-hot encoded);
Acc: average accuracy on the data set-up

Fig. 2. Average and ADR-binned accuracies for two resamplings of the dataset.

4 Conclusion and Discussion

Disaggregating the annotations decreases the majority-bias' effects with adapted ML models' inputs and dataset resamplings. Binning the evaluation metrics enables to understand and verify the existence of these effects. We only reported results using the LR classifier but we now investigate adaptations of Deep Learning algorithm's architectures which are better suited to the large dataset (10 times more annotations than labels) and to the size of the ML inputs.

Acknowledgements

This research is supported by the Capture Bias project [3], part of the VWData Research Programme funded by the Startimpuls programme of the Dutch National Research Agenda, route "Value Creation through Responsible Access to and use of Big Data" (NWO 400.17.605/4174).

References

1. Aroyo, L., Welty, C.: Crowd truth: Harnessing disagreement in crowdsourcing a relation extraction gold standard. WebSci2013. ACM **2013** (2013)
2. Cowan, G., Hodge, C.: Judgments of hate speech: The effects of target group, publicness, and behavioral responses of the target. Journal of Applied Social Psychology **26**(4), 355–374 (1996)
3. Dixon, L., Li, J., Sorensen, J., Thain, N., Vasserman, L.: Measuring and mitigating unintended bias in text classification (2017)
4. Schmidt, A., Wiegand, M.: A survey on hate speech detection using natural language processing. In: Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media. pp. 1–10 (2017)
5. Tsesis, A.: Hate in cyberspace: Regulating hate speech on the internet. San Diego L. Rev. **38**, 817 (2001)
6. Wulczyn, E., Thain, N., Dixon, L.: Ex machina: Personal attacks seen at scale. In: Proceedings of the 26th International Conference on World Wide Web. pp. 1391–1399. International World Wide Web Conferences Steering Committee (2017)

[3] https://capturebias.eu/

Bibliography

[1] Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. *arXiv preprint arXiv:1803.02453*, 2018.

[2] Cecilia Ovesdotter Alm. Subjective natural language problems: Motivations, applications, characterizations, and implications. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 107–112. Association for Computational Linguistics, 2011.

[3] Hector Martinez Alonso, Anders Johannsen, and Barbara Plank. Supersense tagging with inter-annotator disagreement. In *Linguistic Annotation Workshop 2016*, pages 43–48, 2016.

[4] Omar Alonso and Ricardo Baeza-Yates. Design and implementation of relevance assessments using crowdsourcing. In *European Conference on Information Retrieval*, pages 153–164. Springer, 2011.

[5] Julia Angwin, Jeff Larson, Surya Mattu, and Lauren Kirchner. Machine bias: There's software used across the country to predict future criminals. *And it's biased against blacks. ProPublica*, 2016.

[6] David Archard. Insults, free speech and offensiveness. *Journal of Applied Philosophy*, 31(2):127–141, 2014.

[7] Lora Aroyo and Chris Welty. Crowd truth: Harnessing disagreement in crowdsourcing a relation extraction gold standard. *WebSci2013. ACM*, 2013, 2013.

[8] Lora Aroyo and Chris Welty. The three sides of crowdtruth. *Journal of Human Computation*, 1:31–34, 2014.

[9] Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. Deep learning for hate speech detection in tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pages 759–760. International World Wide Web Conferences Steering Committee, 2017.

[10] Beata Beigman Klebanov and Eyal Beigman. From annotator agreement to noise models. *Computational Linguistics*, 35(4):495–503, 2009.

[11] Simone Bianco, Luigi Celona, Paolo Napoletano, and Raimondo Schettini. Predicting image aesthetics with deep learning. In *International Conference on Advanced Concepts for Intelligent Vision Systems*, pages 117–125. Springer, 2016.

[12] Reuben Binns. Fairness in machine learning: Lessons from political philosophy. In *Conference on Fairness, Accountability and Transparency*, pages 149–159, 2018.

[13] Reuben Binns, Michael Veale, Max Van Kleek, and Nigel Shadbolt. Like trainer, like bot? inheritance of bias in algorithmic content moderation. In *International Conference on Social Informatics*, pages 405–415. Springer, 2017.

[14] Robert J Boeckmann and Jeffrey Liew. Hate speech: Asian american students' justice judgments and psychological responses. *Journal of Social Issues*, 58(2):363–381, 2002.

[15] Engin Bozdag and Jeroen van den Hoven. Breaking the filter bubble: democracy and design. *Ethics and Information Technology*, 17(4):249–265, 2015.

[16] Alessandro Bozzon, Marco Brambilla, Stefano Ceri, Matteo Silvestri, and Giuliano Vesci. Choosing the right crowd: expert finding in social networks. In *Proceedings of the 16th International Conference on Extending Database Technology*, pages 637–648. ACM, 2013.

[17] Anthony Brew, Derek Greene, and Pádraig Cunningham. Using crowdsourcing and active learning to track sentiment in online media. In *ECAI*, pages 145–150, 2010.

[18] Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Conference on Fairness, Accountability and Transparency*, pages 77–91, 2018.

[19] Peter Burnap and Matthew Leighton Williams. Hate speech, machine classification and statistical modelling of information flows on twitter: Interpretation and communication for policy decision making. 2014.

[20] Eshwar Chandrasekharan, Mattia Samory, Anirudh Srinivasan, and Eric Gilbert. The bag of communities: Identifying abusive behavior online with preexisting internet data. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 3175–3187. ACM, 2017.

[21] Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Emiliano De Cristofaro, Gianluca Stringhini, and Athena Vakali. Mean birds: Detecting aggression and bullying on twitter. In *Proceedings of the 2017 ACM on Web Science Conference*, pages 13–22. ACM, 2017.

[22] Nitesh V Chawla. Data mining for imbalanced datasets: An overview. In *Data mining and knowledge discovery handbook*, pages 875–886. Springer, 2009.

[23] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[24] Hao Chen, Susan Mckeever, and Sarah Jane Delany. Harnessing the power of text mining for the detection of abusive content in social media. In *Advances in Computational Intelligence Systems*, pages 187–205. Springer, 2017.

[25] Hao Chen, Susan Mckeever, and Sarah Jane Delany. Presenting a labelled dataset for real-time detection of abusive user posts. In *Proceedings of the International Conference on Web Intelligence*, pages 884–890. ACM, 2017.

[26] Kuan-Ta Chen, Chen-Chi Wu, Yu-Chun Chang, and Chin-Laung Lei. A crowdsourceable qoe evaluation framework for multimedia content. In *Proceedings of the 17th ACM international conference on Multimedia*, pages 491–500. ACM, 2009.

[27] Ying Chen, Yilu Zhou, Sencun Zhu, and Heng Xu. Detecting offensive language in social media to protect adolescent online safety. In *Privacy, Security, Risk and Trust (PASSAT), 2012 International Conference on and 2012 International Confernece on Social Computing (SocialCom)*, pages 71–80. IEEE, 2012.

[28] Alexandra Chouldechova and Max G'Sell. Fairer and more accurate, but for whom? *arXiv preprint arXiv:1707.00046*, 2017.

[29] Theodora Chu, Kylie Jue, and Max Wang. Comment abuse classification with deep learning.

[30] Sam Corbett-Davies, Emma Pierson, Avi Feller, Sharad Goel, and Aziz Huq. Algorithmic decision making and the cost of fairness. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 797–806. ACM, 2017.

[31] Paul Covington, Jay Adams, and Emre Sargin. Deep neural networks for youtube recommendations. In *Proceedings of the 10th ACM Conference on Recommender Systems*, pages 191–198. ACM, 2016.

[32] Gloria Cowan and Cyndi Hodge. Judgments of hate speech: The effects of target group, publicness, and behavioral responses of the target. *Journal of Applied Social Psychology*, 26(4):355–374, 1996.

[33] Gloria Cowan and Désirée Khatchadourian. Empathy, ways of knowing, and interdependence as mediators of gender differences in attitudes toward hate speech and freedom of speech. *Psychology of Women Quarterly*, 27(4):300–308, 2003.

[34] Gloria Cowan and Jon Mettrick. The effects of target variables and setting on perceptions of hate speech1. *Journal of Applied Social Psychology*, 32(2):277–299, 2002.

[35] George B Cunningham, Mauricio Ferreira, and Janet S Fink. Reactions to prejudicial statements: The influence of statement content and characteristics of the commenter. *Group Dynamics: Theory, Research, and Practice*, 13(1):59, 2009.

[36] Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. Automated hate speech detection and the problem of offensive language. *arXiv preprint arXiv:1703.04009*, 2017.

[37] Karen R Dickson. All prejudices are not created equal: Different responses to subtle versus blatant expressions of prejudice. 2012.

[38] Karthik Dinakar, Roi Reichart, and Henry Lieberman. Modeling the detection of textual cyberbullying. *The Social Mobile Web*, 11(02), 2011.

[39] Nemanja Djuric, Jing Zhou, Robin Morris, Mihajlo Grbovic, Vladan Radosavljevic, and Narayan Bhamidipati. Hate speech detection with comment embeddings. In *Proceedings of the 24th International Conference on World Wide Web*, pages 29–30. ACM, 2015.

[40] Daniel M Downs and Gloria Cowan. Predicting the importance of freedom of speech and the perceived harm of hate speech. *Journal of applied social psychology*, 42(6):1353–1375, 2012.

[41] Anca Dumitrache, Oana Inel, Benjamin Timmermans, and Lora Aroyo. Crowdsourcing ambiguity-aware ground truth.

[42] Anca Dumitrache, Lora Aroyo, and Chris Welty. Achieving expert-level annotation quality with crowdtruth. In *Proc. of BDM2I Workshop, ISWC*, 2015.

[43] Anca Dumitrache, Lora Aroyo, and Chris Welty. Crowdtruth measures for language ambiguity. In *Proc. of LD4IE Workshop, ISWC*, 2015.

[44] Anca Dumitrache, Oana Inel, Lora Aroyo, and Chris Welty. Metrics for capturing ambiguity in crowdsourcing by interlinking workers, annotations and input data. 2018.

[45] Cynthia Dwork, Nicole Immorlica, Adam Tauman Kalai, and Mark DM Leiserson. Decoupled classifiers for group-fair and efficient machine learning. In *Conference on Fairness, Accountability and Transparency*, pages 119–133, 2018.

[46] Ziv Epstein, Blakeley H Payne, Judy Hanwen Shen, Abhimanyu Dubey, Bjarke Felbo, Matthew Groh, Nick Obradovich, Manuel Cebrian, and Iyad Rahwan. Closing the ai knowledge gap. *arXiv preprint arXiv:1803.07233*, 2018.

[47] Óscar Figuerola Salas, Velibor Adzic, Akash Shah, and Hari Kalva. Assessing internet video quality using crowdsourcing. In *Proceedings of the 2nd ACM international workshop on Crowdsourcing for Multimedia*, pages 23–28. ACM, 2013.

[48] Michael J Franklin, Donald Kossmann, Tim Kraska, Sukriti Ramesh, and Reynold Xin. Crowddb: answering queries with crowdsourcing. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pages 61–72. ACM, 2011.

[49] Björn Gambäck and Utpal Kumar Sikdar. Using convolutional neural networks to classify hate-speech. In *Proceedings of the First Workshop on Abusive Language Online*, pages 85–90, 2017.

[50] Lei Gao and Ruihong Huang. Detecting online hate speech using context aware models. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 260–266, 2017.

[51] Lei Gao, Alexis Kuppersmith, and Ruihong Huang. Recognizing explicit and implicit hate speech using a weakly supervised two-path bootstrapping approach. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, volume 1, pages 774–782, 2017.

[52] Eva García-Martín and Niklas Lavesson. Is it ethical to avoid error analysis? In *2017 Workshop on Fairness, Accountability, and Transparency in Machine Learning (FAT/ML 2017)*. arXiv, 2017.

[53] Deepti Ghadiyaram and Alan C Bovik. Massive online crowdsourced study of subjective and objective picture quality. *IEEE Transactions on Image Processing*, 25(1):372–387, 2016.

[54] Joshua Guberman and Libby Hemphill. Challenges in modifying existing scales for detecting harassment in individual tweets. In *Proceedings of the 50th Hawaii International Conference on System Sciences*, 2017.

[55] Moritz Hardt, Eric Price, Nati Srebro, et al. Equality of opportunity in supervised learning. In *Advances in neural information processing systems*, pages 3315–3323, 2016.

[56] Jeffrey Heer and Michael Bostock. Crowdsourcing graphical perception: using mechanical turk to assess visualization design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 203–212. ACM, 2010.

[57] PJ Henry, Sarah E Butler, and Mark J Brandt. The influence of target group status on the perception of the offensiveness of group-based slurs. *Journal of Experimental Social Psychology*, 53:185–192, 2014.

[58] Tobias Hoβfeld, Raimund Schatz, and Sebastian Egger. Sos: The mos is not enough! In *Quality of Multimedia Experience (QoMEX), 2011 Third International Workshop on*, pages 131–136. IEEE, 2011.

[59] Tobias Hoßfeld, Michael Seufert, Matthias Hirth, Thomas Zinner, Phuoc Tran-Gia, and Raimund Schatz. Quantification of youtube qoe via crowdsourcing. In *Multimedia (ISM), 2011 IEEE International Symposium on*, pages 494–499. IEEE, 2011.

[60] Tobias Hoßfeld, Matthias Hirth, Pavel Korshunov, Philippe Hanhart, Bruno Gardlo, Christian Keimel, and Christian Timmerer. Survey of web-based crowdsourcing frameworks for subjective quality assessment. In *Multimedia Signal Processing (MMSP), 2014 IEEE 16th International Workshop on*, pages 1–6. IEEE, 2014.

[61] Jeff Howe. The rise of crowdsourcing. *Wired magazine*, 14(6):1–4, 2006.

[62] Pei-Yun Hsueh, Prem Melville, and Vikas Sindhwani. Data quality from crowdsourcing: a study of annotation selection criteria. In *Proceedings of the NAACL HLT 2009 workshop on active learning for natural language processing*, pages 27–35. Association for Computational Linguistics, 2009.

[63] Panagiotis G Ipeirotis and Evgeniy Gabrilovich. Quizz: targeted crowdsourcing with a billion (potential) users. In *Proceedings of the 23rd international conference on World wide web*, pages 143–154. ACM, 2014.

[64] Emily Jamison and Iryna Gurevych. Noise or additional information? leveraging crowdsource annotation item agreement for natural language tasks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 291–297, 2015.

[65] Chaitanya K Joshi, Fei Mi, and Boi Faltings. Personalization in goal-oriented dialog. *arXiv preprint arXiv:1706.07503*, 2017.

[66] Toshihiro Kamishima, Shotaro Akaho, Hideki Asoh, and Jun Sakuma. Fairness-aware classifier with prejudice remover regularizer. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 35–50. Springer, 2012.

[67] Christian Keimel, Julian Habigt, Clemens Horch, and Klaus Diepold. Qualitycrowd—a framework for crowd-based quality evaluation. In *Picture Coding Symposium (PCS), 2012*, pages 245–248. IEEE, 2012.

[68] Aniket Kittur, Ed H Chi, and Bongwon Suh. Crowdsourcing user studies with mechanical turk. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 453–456. ACM, 2008.

[69] Quoc Le and Tomas Mikolov. Distributed representations of sentences and documents. In *International Conference on Machine Learning*, pages 1188–1196, 2014.

[70] Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 994–1003, 2016.

[71] Revathy Amadera Lingam and Norizah Aripin. Comments on fire! classifying flaming comments on youtube videos in malaysia. *Jurnal Komunikasi, Malaysian Journal of Communication*, 33(4), 2017.

[72] Bingquan Liu, Zhen Xu, Chengjie Sun, Baoxun Wang, Xiaolong Wang, Derek F Wong, and Min Zhang. Content-oriented user modeling for personalized response ranking in chatbots. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 2017.

[73] Amrita Mangaonkar, Allenoush Hayrapetian, and Rajeev Raje. Collaborative detection of cyberbullying behavior in twitter data. In *Electro/Information Technology (EIT), 2015 IEEE International Conference on*, pages 611–616. IEEE, 2015.

[74] Panagiotis Mavridis, David Gross-Amblard, and Zoltán Miklós. Using hierarchical skills for optimized task assignment in knowledge-intensive crowdsourcing. In *Proceedings of the 25th International Conference on World Wide Web*, pages 843–853. International World Wide Web Conferences Steering Committee, 2016.

[75] Maw Maw and Vimala Balakrishnan. An analysis on the hateful contents detection techniques on social media. 2016.

[76] Andrés Montoyo, Patricio MartíNez-Barco, and Alexandra Balahur. Subjectivity and sentiment analysis: An overview of the current state of the area and envisaged developments, 2012.

[77] Chikashi Nobata, Joel Tetreault, Achint Thomas, Yashar Mehdad, and Yi Chang. Abusive language detection in online user content. In *Proceedings of the 25th International Conference on World Wide Web*, pages 145–153. International World Wide Web Conferences Steering Committee, 2016.

[78] Ika Nurfarida and Laudetta Dianne Fitri. Mapping and defining hate speech in instagram's comments: A study of language use in social media.

[79] Conor J O'Dea, Stuart S Miller, Emma B Andres, Madelyn H Ray, Derrick F Till, and Donald A Saucier. Out of bounds: factors affecting the perceived offensiveness of racial slurs. *Language Sciences*, 52:155–164, 2015.

[80] Cathy O'Neil. *Weapons of math destruction: How big data increases inequality and threatens democracy*. Broadway Books, 2016.

[81] John Pavlopoulos, Prodromos Malakasiotis, and Ion Androutsopoulos. Deep learning for user comment moderation. *ACL 2017*, page 25, 2017.

[82] Rob Price. Microsoft is deleting its ai chatbot's incredibly racist tweets. *Business Insider*, 2016.

[83] Benjamin Rainer, Markus Waltl, and Christian Timmerer. A web based subjective evaluation platform. In *Quality of Multimedia Experience (QoMEX), 2013 Fifth International Workshop on*, pages 24–25. IEEE, 2013.

[84] Judith Alice Redi, Tobias Hoßfeld, Pavel Korshunov, Filippo Mazza, Isabel Povoa, and Christian Keimel. Crowdsourcing-based multimedia subjective evaluations: a case study on image recognizability and aesthetic appeal. In *Proceedings of the 2nd ACM international workshop on Crowdsourcing for multimedia*, pages 29–34. ACM, 2013.

[85] Dennis Reidsma. Annotations and subjective machines of annotators, embodied agents, users, and other humans. 2008.

[86] Dennis Reidsma et al. Exploiting'subjective'annotations. In *Proceedings of the Workshop on Human Judgements in Computational Linguistics*, pages 8–16. Association for Computational Linguistics, 2008.

[87] Kelly Reynolds, April Kontostathis, and Lynne Edwards. Using machine learning to detect cyberbullying. In *Machine learning and applications and workshops (ICMLA), 2011 10th International Conference on*, volume 2, pages 241–244. IEEE, 2011.

[88] Flávio Ribeiro, Dinei Florencio, and Vítor Nascimento. Crowdsourcing subjective image quality evaluation. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 3097–3100. IEEE, 2011.

[89] Sergio Rojas-Galeano. On obstructing obscenity obfuscation. *ACM Transactions on the Web (TWEB)*, 11(2):12, 2017.

[90] Andrea Romei and Salvatore Ruggieri. A multidisciplinary survey on discrimination analysis. *The Knowledge Engineering Review*, 29(5):582–638, 2014.

[91] Marta Sabou, Kalina Bontcheva, Leon Derczynski, and Arno Scharl. Corpus annotation through crowdsourcing: Towards best practice guidelines. In *LREC*, pages 859–866, 2014.

[92] Sasha Sax. Flame wars: Automatic insult detection, 2016.

[93] Markus Schedi, Mats Sjöberg, Ionuţ Mironică, Bogdan Ionescu, Vu Lam Quang, Yu-Gang Jiang, and Claire-Hélène Demarty. Vsd2014: a dataset for violent scenes detection in hollywood movies and web videos. In *Content-Based Multimedia Indexing (CBMI), 2015 13th International Workshop on*, pages 1–6. IEEE, 2015.

[94] Anna Schmidt and Michael Wiegand. A survey on hate speech detection using natural language processing. In *Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media. Association for Computational Linguistics, Valencia, Spain*, pages 1–10, 2017.

[95] Leandro Araújo Silva, Mainack Mondal, Denzil Correa, Fabrício Benevenuto, and Ingmar Weber. Analyzing the targets of hate in online social media. In *ICWSM*, pages 687–690, 2016.

[96] Michael Skirpan and Micha Gorelick. The authority of" fair" in machine learning. *arXiv preprint arXiv:1706.09976*, 2017.

[97] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y Ng. Cheap and fast—but is it good?: evaluating non-expert annotations for natural language tasks. In *Proceedings of the conference on empirical methods in natural language processing*, pages 254–263. Association for Computational Linguistics, 2008.

[98] Sara Sood, Judd Antin, and Elizabeth Churchill. Profanity use in online communities. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1481–1490. ACM, 2012.

[99] Sara Owsley Sood, Judd Antin, and Elizabeth F Churchill. Using crowdsourcing to improve profanity detection. In *AAAI Spring Symposium: Wisdom of the Crowd*, volume 12, page 06, 2012.

[100] Sara Owsley Sood, Elizabeth F Churchill, and Judd Antin. Automatic identification of personal insults on social news sites. *Journal of the Association for Information Science and Technology*, 63(2):270–285, 2012.

[101] Jacquelin A Speck, Erik M Schmidt, Brandon G Morton, and Youngmoo E Kim. A comparative study of collaborative vs. traditional musical mood annotation. In *ISMIR*, pages 549–554, 2011.

[102] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015.

[103] Pawel Swietojanski and Steve Renals. Learning hidden unit contributions for unsupervised speaker adaptation of neural network acoustic models. In *Spoken Language Technology Workshop (SLT), 2014 IEEE*, pages 171–176. IEEE, 2014.

[104] Duyu Tang, Bing Qin, Ting Liu, and Yuekui Yang. User modeling with neural network for review rating prediction. In *IJCAI*, pages 1340–1346, 2015.

[105] Alexander Tsesis. Hate in cyberspace: Regulating hate speech on the internet. *San Diego L. Rev.*, 38: 817, 2001.

[106] Michael Veale and Reuben Binns. Fairer machine learning in the real world: Mitigating discrimination without collecting sensitive data. *Big Data & Society*, 4(2):2053951717743530, 2017.

[107] William Warner and Julia Hirschberg. Detecting hate speech on the world wide web. In *Proceedings of the Second Workshop on Language in Social Media*, pages 19–26. Association for Computational Linguistics, 2012.

[108] Zeerak Waseem and Dirk Hovy. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *SRW@ HLT-NAACL*, pages 88–93, 2016.

[109] Amanda Williams, Clio Oliver, Katherine Aumer, and Chanel Meyers. Racial microaggressions and perceptions of internet memes. *Computers in Human Behavior*, 63:424–432, 2016.

[110] Ou Wu, Yunfei Chen, Bing Li, and Weiming Hu. Evaluating the visual quality of web pages using a computational aesthetic approach. In *Proceedings of the fourth ACM international conference on Web search and data mining*, pages 337–346. ACM, 2011.

[111] Ellery Wulczyn, Nithum Thain, and Lucas Dixon. Ex machina: Personal attacks seen at scale. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1391–1399. International World Wide Web Conferences Steering Committee, 2017.

[112] Jun-Ming Xu, Kwang-Sung Jun, Xiaojin Zhu, and Amy Bellmore. Learning from bullying traces in social media. In *Proceedings of the 2012 conference of the North American chapter of the association for computational linguistics: Human language technologies*, pages 656–666. Association for Computational Linguistics, 2012.

[113] Zhi Xu and Sencun Zhu. Filtering offensive language in online communities using grammatical relations. In *Proceedings of the Seventh Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference*, pages 1–10, 2010.

[114] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rodriguez, and Krishna P Gummadi. Fairness beyond disparate treatment & disparate impact: Learning classification without disparate mistreatment. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1171–1180. International World Wide Web Conferences Steering Committee, 2017.

[115] Muhammad Bilal Zafar, Isabel Valera, Manuel Rodriguez, Krishna Gummadi, and Adrian Weller. From parity to preference-based notions of fairness in classification. In *Advances in Neural Information Processing Systems*, pages 228–238, 2017.

[116] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. Learning fair representations. In *International Conference on Machine Learning*, pages 325–333, 2013.

[117] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. Personalizing dialogue agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243*, 2018.

[118] Zhe Zhang and Daniel B Neill. Identifying significant predictive bias in classifiers. *arXiv preprint arXiv:1611.08292*, 2016.

[119] Ziqi Zhang, David Robinson, and Jonathan Tepper. Detecting hate speech on twitter using a convolution-gru based deep neural network. In *Lecture notes in computer science*. Springer Verlag, 2018.

[120] Indre Zliobaite. A survey on measuring indirect discrimination in machine learning. *arXiv preprint arXiv:1511.00148*, 2015.

[121] Indrė Žliobaitė. Measuring discrimination in algorithmic decision making. *Data Mining and Knowledge Discovery*, 31(4):1060–1089, 2017.