Counterfactually Fair Automatic Speech Recognition

Leda Sarı, Student Member, IEEE, Mark Hasegawa-Johnson, Fellow, IEEE, and Chang D. Yoo, Senior Member, IEEE

Abstract—Widely used automatic speech recognition (ASR) systems have been empirically demonstrated in various studies to be unfair, having higher error rates for some groups of users than others. One way to define fairness in ASR is to require that changing the demographic group affiliation of any individual (e.g., changing their gender, age, education or race) should not change the probability distribution across possible speech-to-text transcriptions. In the paradigm of counterfactual fairness, all variables independent of group affiliation (e.g., the text being read by the speaker) remain unchanged, while variables dependent on group affiliation (e.g., the speaker’s voice) are counterfactually modified. Hence, we approach the fairness of ASR by training the ASR to minimize change in its outcome probabilities despite a counterfactual change in the individual’s demographic attributes. Starting from the individualized counterfactual equal odds criterion, we provide relaxations to it and compare their performances. On the Corpus of Regional African American Language (CORAAL) and LibriSpeech dataset to accommodate for differences due to gender, age, education, and race. We show that with counterfactual training, we can reduce average character error rates while achieving lower performance gap between demographic groups, and lower error standard deviation among individuals.

Index Terms—Automatic speech recognition, speaker adaptation, fairness in machine learning, counterfactual fairness.

I. INTRODUCTION

MACHINE learning techniques have recently been utilized in decision making for critical settings such as financial services and hiring for a job [1]–[3]. Evaluation of these models has shown that optimizing for commonly-used measures such as accuracy can sometimes be unfair in the sense that individuals belonging to certain groups based on their gender, age, race, etc. may be disadvantaged, e.g., having lower accuracy than a majority group. These concerns have led researchers to focus on fairness in machine learning.

There are several mutually exclusive definitions of the word “fair” [2]. Bickel, Hammel and O’Connell [4] studied a dataset demonstrating significantly lower rates of college admission for women than for men. They argued that the data showed no unfairness, because women were applying to different departments than men; when department was entered as a factor in the statistical model, the apparent unfairness vanished. Srivastava et al. [5], however, present data suggesting that definitions of fairness involving more than two interacting variables are irrelevant to public perception: the users of an algorithm rate its fairness based on the demographic parity of its outcomes.

Pearl [6] re-analyzed the data of Bickel et al. using a causal model, in which all of the variables of interest are explicitly represented. The framework of causal reasoning treats problems of fairness as open discussions about the causal links among variables [7]. For example, a job hiring decision might depend on physical height, habitual clothing, or choice of undergraduate major. All of these variables correlate with gender; any of them might be considered to be either a fair or unfair basis for discrimination, depending on the job and the observer. The framework of counterfactual fairness [8] requires the designer of an algorithm to explicitly state which variables are permitted to change the outcome probabilities, and which variables are protected attributes that are not allowed to change the outcomes. In the original proposal [8], every variable that is causally dependent on a protected attribute must then be regressed against the protected attribute in order to infer a latent residual explanatory variable, independent of the protected attribute, on the basis of which decisions may be made. Many recent models in computer vision [9], [10] and natural language processing [11], [12] train statistical models to infer the residual variables implicitly, based on training using counterfactual training data. In a counterfactual training dataset, the protected attributes of individuals are counterfactually modified, and all variables dependent on the protected attribute are subject to possible consequent dependent modifications. A machine learning algorithm forced to ignore such counterfactual modifications is thereby forced to implicitly learn the residual information contained in each such variable after removing its dependence on the protected attribute.

As an important machine learning application, automatic speech recognition (ASR) is also subject to fairness concerns. Various studies have shown concerns regarding the performance gap between male and female speakers [13] as well as black and white speakers [14]. Because of the power of modern speaker adaptation methods [15]–[17], the unfairness of ASR is usually cast as a problem of unfair training corpora, e.g., [18] describes the under-performance of ASR for female speakers as a natural consequence of the under-representation of women in ASR training corpora. Counterfactual fairness provides a complementary approach that can provide some benefit even when the dataset is biased by focusing on fair treatment of every individual. Counterfactual fairness requires
that if the demographic group affiliation of any given speaker (e.g., gender, race, age, education) were changed, the accuracy of the ASR should not change.

In this work, we propose counterfactually fair automatic speech recognition. We train the ASR so that it generates equivalent output label distributions for counterfactual speakers whose voices have been resampled with different protected attributes, but are otherwise identical in every respect that is not causally dependent on the protected attribute. We can summarize the main contributions of this study as follows:

- Differences in ASR error rate between groups, and accuracy standard deviation among individuals, are both reduced by the use of counterfactual training data.
- Counterfactual training data is applied in two ways: (1) using data augmentation, i.e., training the ASR to accurately transcribe it, and (2) using counterfactual equalized odds [19], i.e., forcing the ASR to ignore the difference between factual and counterfactual data. Of these two, data augmentation is empirically demonstrated to reduce the average error rate of the ASR, but increase unfairness, while the method of counterfactual equalized odds reduces both the average error rate and the unfairness of the recognizer.
- The method of counterfactual equalized odds is made applicable to connectionist temporal classification (CTC) [20] by the invention of three different sequence training criteria, of which only one is observed to be empirically successful.

The rest of the article is organized as follows: Section II reviews CTC and fairness in machine learning. Section III derives our proposed counterfactually fair training criteria for ASR. Section IV presents the experimental settings, and Section V presents the results. Section VI discusses the scale and source of the error rate reductions. We conclude the paper with a summary in Section VII.

II. BACKGROUND

In the following sections, we will use the following notation: scalar values will be denoted by small-case normal font \( x \), vectors will be denoted by small-case bold letters \( \mathbf{x} \), sequence of vectors (or matrices) will be denoted by small-case bold letters with bar \( \bar{\mathbf{x}} \), sets will be denoted by calligraphic symbols \( \mathcal{X} \). For random variables we will use a similar notation except that we will replace small-case letters with capitals \( X, \bar{X}, \mathbf{X} \), respectively.

A. End-to-end ASR and CTC Loss

In recent years, the availability of very large speech corpora and increased computation power led speech researchers to investigate end-to-end approaches for ASR. An end-to-end ASR is viewed as a transducer which maps from an acoustic vector sequence \( \mathbf{x} = [x_1, \ldots, x_T] \), \( x_t \in \mathcal{X} \) where \( m \) is the dimension of the vectors, to a sequence of characters \( y = [y_1, \ldots, y_S] \), \( y_s \in \mathcal{Y} \), where we assume that \( S \leq T \). Although there are several paradigms for end-to-end ASR, including CTC-based models [21], RNN transducers [22], purely attention-based transducers [23], and joint models that combine CTC with encoder-decoder models [24], we will focus on CTC-based models in this study, which we will review next.

Neural network training using CTC loss [20], which was proposed for sequence-to-sequence labeling tasks, has become one of the major approaches for end-to-end ASR systems [21], [25]. Since \( S \leq T \), CTC decomposes the probability \( P(Y|X) \) into a sum over alignment paths between input and output. The alignment path is a length-\( T \) sequence \( \pi = [\pi_1, \ldots, \pi_T] \), \( \pi_t \in \mathcal{Y} \cup \{-\} \), where the blank character \( - \) is added to the label inventory in order to assist in the definition of a surjective mapping \( I : \mathcal{Y} \cup \{-\} \) \( \rightarrow Y^S \) from any alignment to its corresponding label sequence. Suppose that a neural network generates per-frame softmax outputs \( P(\Pi_t = \pi_t|X = \bar{x}) \) for the input \( \bar{x} \), then

\[
P(Y = y|X = \bar{x}) = \sum_{\pi|l(\pi) = y} \prod_{t=1}^{T} P(\Pi_t = \pi_t|X = \bar{x}).
\]

(1)

The network that generates the softmax outputs is trained by minimizing the negative log-likelihood, denoted by \( \mathcal{L}_{\text{CTC}} \), as

\[
\mathcal{L}_{\text{CTC}}(\bar{x}, y) = -\log \sum_{\pi:|l(\pi) = y} \exp \sum_{t=1}^{T} \log P(\Pi_t = \pi_t|X = \bar{x}).
\]

(2)

As shown in [20], these probabilities can be computed efficiently using a forward-backward algorithm, in which all transitions satisfying the constraint \( l(\pi) = y \) have equal transition probability, and in which the observation probability is the softmax output of the neural network. The algorithm first starts by augmenting the original sequence with a special blank symbol \( - \) and produces a new augmented sequence \( y' = [-, y_1, -, y_2, -, \ldots, -, y_S, -] = [y'_1, \ldots, y'_{2S+1}] \) of length \( 2S + 1 \). Using capital letters to denote random variables and small letters to denote their instance values, CTC defines forward and backward probabilities as \( \alpha_t(z) = P(Z_t = z, Y'_1 = y'_1, \ldots, Y'_z = y'_z|X = \mathbf{x}) \) and \( \beta_t(z) = P(Y'_z+1 = y'_z+1, \ldots, Y'_{2S+1} = y'_{2S+1}|Z_t = z, \mathbf{X} = \mathbf{x}) \), where \( z_t \in \{1, \ldots, 2S+1\} \) is an index into the sequence \( y' \). The total probability of the observed sequence \( y \) can then be written for any \( t \) as

\[
P(Y = y|X = \mathbf{x}) = \sum_{z_t \in \{1, \ldots, 2S+1\}} \alpha_t(z) \beta_t(z),
\]

(3)

or specifically for the last time index \( T \) as

\[
P(Y = y|X = \mathbf{x}) = \alpha_T(|y'|) - 1 + \alpha_T(|y'|),
\]

(4)

where \( |y'| = 2S + 1 \) denotes the length of \( y' \).

B. Fairness in Machine Learning

It has been observed that machine learning models are liable in making unfair decisions [26] due to various reasons including data bias, missing data from certain groups and algorithmic bias. To evaluate the unfairness in machine learning, several fairness criteria have been defined [27]. Since it is impossible to simultaneously satisfy all possible definitions of fairness [2], recent papers emphasize the need for algorithm designers to
specify a causal graph relating the variables of interest, in a manner that permits open discussion of the presumed causal dependencies among them [28].

According to [29], equalized odds is defined as the condition in which a predictor \( \hat{Y} \in \mathcal{Y} \) of an outcome \( Y \in \mathcal{Y} \) is conditionally independent, given \( Y = y \), of the protected attribute \( A \in \mathcal{A} \):

\[
P(\hat{Y} = \hat{y} | A = a, Y = y) = P(\hat{Y} = \hat{y} | A = \tilde{a}, Y = y)
\]

\[\forall a, \tilde{a} \in \mathcal{A}, y, \hat{y} \in \mathcal{Y}.\] (5)

Causal analyses of fairness require the algorithm designer to specify a protected attribute, \( A \), with respect to which the outcome \( \hat{Y} \) should be fair. The observed variables, \( X \), are then divided into descendants \( (X_d) \) and non-descendants \( (X_{nd}) \) of \( A \). Each descendant variable may then be hypothesized to be the deterministic result of interaction between the protected attribute and a latent variable, \( U \), which is independent of \( A \).

In [8], the notion of counterfactual fairness is introduced. A counterfactually fair algorithm is defined to be an algorithm whose outcome probability distribution is unchanged even if the value of the protected attribute is changed. A test for counterfactual fairness involves three steps: abduction, action, and prediction [30]. The abduction step computes the distribution \( P(U | X = x, A = a) \) of the latent variables. The action step then modifies the protected attribute, \( A \leftarrow \tilde{a} \). Finally, the prediction step computes the distribution of the observed descendants and the outcome variable, averaged over the abducted distribution of \( U \); we write this weighted average outcome distribution as

\[
P(\hat{Y}_{A\leftarrow\tilde{a}} | X = x, A = a) = \sum_a P(\hat{Y} | A = \tilde{a}, X_{nd} = x_{nd}, U = u) \times P(U = u | X = x, A = a).\] (6)

The criterion of counterfactual fairness requires that the outcome distribution should be unchanged by the operations of intervention and prediction, i.e.,

\[
P(\hat{Y}_{A\leftarrow\tilde{a}} | X = x, A = a) = P(\hat{Y}_{A\leftarrow\tilde{a}} | X = x, A = a).\] (7)

Many recent papers use deep generative models, such as variational auto-encoders (VAEs) or generative adversarial networks (GANs) [31], [32], to perform the abduction and action steps in a counterfactual fairness training paradigm. A VAE infers \( U \) from the variational posterior distribution, \( P(U = u | x, a) | A = a, X = x) \), then infers counterfactual observations \( x_{A\leftarrow\tilde{a}} \) from the generative distribution \( P(X | U = u, x, a) | A = \tilde{a}) \), resulting in a one-term approximation of Eq. (6). A GAN infers \( U \) in order to maximize the joint distribution \( P(U = u | x, a) | A = a, X = x) \), then infers \( x_{A\leftarrow\tilde{a}} \) from the generative distribution \( P(X | U = u, x, a) | A = \tilde{a}) \).

In the case of ASR, we can consider the variables in the counterfactual framework as the following: \( A \) represents the protected attribute of the speaker (gender, age, education level, etc.). Outcome represents the output unit which could be character or phoneme. In our experiments, we used character models so our outcomes are characters. Predictor is the estimate of these outcomes, e.g., character whose probability distribution is given by the softmax output layer. The latent variable, \( U \), is the set of all information about the speech signal that is independent of a speaker’s protected attribute. Since we do not know this information, we estimate it implicitly. A voice conversion system modifies the speech signal to have a different value of the protected attribute (e.g., male→female, old→young, little education→much education, etc.). The ASR never explicitly calculates the latent variable, but it assumes that the original and modified speech signals have the same value of the latent variable; Eq. (7) is therefore enforced by requiring that the original and modified speech signals should result in the same ASR outcomes.

C. Counterfactual Equalized Odds

In [19], counterfactual fairness is combined with equalized odds in order to introduce counterfactual equalized odds for all \( x \in \mathcal{X}, y \in \mathcal{Y} \) and \( a \in \mathcal{A} \):

\[
P(\hat{Y}_{A\leftarrow a} | X = x, Y_{A\leftarrow a} = y, A = a) = P(\hat{Y}_{A\leftarrow a} | X = x, Y_{A\leftarrow a} = y, A = a).\] (8)

This equation implies that the counterfactual and factual data are required to have matching outcomes only if their resampled ground truth labels match.

Enforcing Eq. (8) is simplified, in [19], by assuming that \( Y \) is binary. If \( Y \) is binary, then equating the factual and counterfactual posteriors is performed by simply equating the probability of \( \hat{Y} = 1 \) (see Appendix A for details), e.g., by minimizing the squared logit difference \( \Delta \sigma^{-1} \) between factual and counterfactual tokens:

\[
\Delta \sigma^{-1} = \sigma^{-1}(\phi(x_{A\leftarrow a, \tilde{a}})) - \sigma^{-1}(\phi(x, a)),\] (9)

where \( \phi(x, a) = P(\hat{Y} = 1 | X = x, A = a) \) is the classifier output given observation \( x \) and protected attribute \( a \), and \( \sigma^{-1}(\phi) = \ln(\phi/(1 - \phi)) \) is the logit function. Generated counterfactual training examples are also used to augment the training dataset by the use of a counterfactual data augmentation criterion:

\[
\mathcal{L}_{CF} = \sum_{a} 1[a \neq \tilde{a}] J(\phi(x_{A\leftarrow a, \tilde{a}}, y_{A\leftarrow a}), y_{A\leftarrow a}).\] (10)

where \( J \) denotes binary cross-entropy, and \( 1 \) is the indicator function. The logit-pairing and counterfactual cross-entropy training criteria are balanced with the target cross-entropy in a multi-task training framework with task weights \( \lambda_{CF} \) and \( \lambda_{CLM} \), thus the overall training metric is:

\[
\mathcal{L}_{CF+CLM} = J(\phi(x, a), y) + \lambda_{CF} \mathcal{L}_{CF} + \lambda_{CLM} \sum_{\tilde{a}} 1[a \neq \tilde{a}] 1[y = y_{A\leftarrow a}] (\Delta \sigma^{-1})^2.\] (11)

III. COUNTERFACTUAL TRAINING FOR ASR

This paper proposes counterfactually fair ASR. Counterfactual fairness is enforced using a data augmentation scheme based on [19], but extended to sequence data. Extension to sequence data requires us to make assumptions about independence or conditional independence between the protected attribute (\( A \)), the label sequence (\( Y \)), the time-aligned label sequence (\( \Pi \)), and the alignment sequence (\( Z \)). Fig. 1 shows
Models differ in the relative importance of the time-aligned education, while keeping the spoken content the same. In as if the speaker were from a different gender, age, race or hypothetical world. Hence, we need to generate utterances sequence and the protected attribute. the assumed dependence or independence between the label latent variable depends on the text of the utterance (Y). The schematic graphical model for counterfactual data generation is given in Eq. 1a. Details of the auto-encoder architecture are provided in Section IV.

B. Counterfactual CTC Matching

In [19], the label variable, Y, was binary, therefore the label distributions resulting from factual and counterfactual data could be matched by simply matching their logits. The closest analogue in ASR, perhaps, is the CTC loss itself, the metric \( \mathcal{L}_{CTC} \) shown in Eq. (2). Whereas the logit of a binary classifier is \( \ln P(\hat{Y} = 1) - \ln P(\hat{Y} = 0) \), the CTC loss is the negative log probability of the known correct answer, \( \mathcal{L}_{CTC} = - \ln P(\hat{Y} = y) \). Using a similar reasoning as Eq. (8), we provide a relaxed version of equalized counterfactual odds which we call counterfactual CTC matching for ASR:

\[
P(\hat{Y}_{A\rightarrow\tilde{a}} = y | \hat{X} = \tilde{x}, Y_{A\rightarrow\tilde{a}} = y, A = a) = P(\hat{Y}_{A\rightarrow\tilde{a}} = y | \hat{X} = \tilde{x}, Y_{A\rightarrow\tilde{a}} = y, A = a).
\]

Eq. (12) can be interpreted as only requiring similarity between the probabilities of correct outcomes (predicted outcome matches ground truth) given factual and counterfactual individuals. The left-hand side of Eq. (12) is the original CTC loss, \( \mathcal{L}_{CTC}(\hat{x}, y) \). The right-hand side is the CTC loss \( \mathcal{L}_{CTC}(\tilde{x}_{A\rightarrow\tilde{a}}, y) \) computed using a counterfactual speech signal generated from the distribution \( P(\hat{X}|A = \tilde{a}, \hat{U} = \tilde{u}(\tilde{x}, a)) \). Eq. (12) therefore instantiates the causal graph shown in Fig. 1b: the transcription should depend only on the information that is shared in common between \( \tilde{x} \) and \( \tilde{x}_{A\rightarrow\tilde{a}} \), which is the latent variable \( \tilde{u} = \tilde{u}(\tilde{x}, a) \) that the voice conversion system used to compute \( \tilde{x}_{A\rightarrow\tilde{a}} \) from \( \tilde{x} \). If, in Eq. (11), we replace \( \mathcal{L}_{CTC} \) by the CTC loss of counterfactual data, and if we replace the logit pairing term with the difference of the CTC losses between the factual individual, \( \tilde{x} \), and the counterfactual individual, \( \tilde{x}_{A\rightarrow\tilde{a}} \), we arrive at the loss function for counterfactual CTC matching:

\[
\mathcal{L}_{CCM} = \mathcal{L}_{CTC}(\hat{x}, y) + \lambda_{CF} \mathcal{L}_{CTC}(\tilde{x}_{A\rightarrow\tilde{a}}, y) + \lambda_{CCM} \sum_{a \neq \tilde{a}} 1[y = y_{A\rightarrow\tilde{a}}] [\Delta \mathcal{L}_{CTC}]^2,
\]

\[
\Delta \mathcal{L}_{CTC} = \mathcal{L}_{CTC}(\tilde{x}_{A\rightarrow\tilde{a}}, y) - \mathcal{L}_{CTC}(\tilde{x}, y),
\]

where \( \lambda_{CF} \) and \( \lambda_{CCM} \) are hyper-parameters denoting the importance of counterfactual data augmentation and counterfactual CTC loss matching, respectively.

C. Counterfactual Log Probability Matching

Intuitively, the latent variable, \( U \), should contain some information about the time alignment of labels to the speech spectrogram. It is possible to force such an alignment by representing the alignment variable, \( \Pi = [\Pi_1, \ldots, \Pi_T] \), explicitly in the causal graph, as shown in Fig. 1c. This graph depicts the assumption that the label sequence \( Y \), alignment sequence \( \Pi \), and latent variable \( U \) should all be independent of \( A \). The assumption can be enforced by requiring the ASR to learn a

\[\text{(a) Causal graph assumed by the counterfactual feature generation step.}\]

\[\text{(b) Causal graph assumed by the counterfactual CTC matching algorithm.}\]

\[\text{(c) Causal graph assumed by the counterfactual log probability matching algorithm.}\]

\[\text{(d) Causal graph assumed by the counterfactual log posterior matching algorithm.}\]
latent representation such that \( P(\Pi|\bar{U}, A) = P(\Pi|\tilde{U}) \), i.e., \( \Pi \) is conditionally independent of \( A \) given \( \bar{U} \).

Define \( \phi_{kt}(\bar{x}) = P(\Pi_t = k|X = \bar{x}) \), the \( k \)th output of the softmax layer at time \( t \). The requirement that \( \Pi \) is conditionally independent of \( A \) given \( \bar{U} \) can be enforced by requiring that \( \Delta \sigma_{kt}^{-1} = 0 \) for all \( k \) and \( t \), where

\[
\Delta \sigma_{kt}^{-1} = \sigma^{-1}(\phi_{kt}(\bar{x}_{A\rightarrow\bar{a}})) - \sigma^{-1}(\phi_{kt}(\bar{x})).
\]  

(15)

Extending Eq. (11) to include all logits, at all frames, we obtain:

\[
\mathcal{L}_{CLM} = \mathcal{L}_{CTC}(\bar{x}, y) + \lambda_{CF} \mathcal{L}_{CTC}(\bar{x}_{A\rightarrow\bar{a}}, y) + \lambda_{CLM} \sum_t \sum_k \sum_{\bar{a}} \sum_{\bar{a}'} \mathbb{1}[\bar{a} \neq \bar{a}'](\Delta \sigma_{kt}^{-1})^2.
\]  

(16)

The neural network in ASR has a multi-class output layer, hence it is convenient to use the log probability rather than the binary logit function: replace \( \sigma^{-1}(\phi_{kt}(\bar{x})) \) by \( \log \phi_{kt}(\bar{x}) \). Logits and log-probability give the same result in the limit of zero training corpus error (Appendix A).

**D. Counterfactual Log Posterior Matching**

In [19], the label, \( Y \), is assumed to depend on the protected attribute. A comparable assumption in the case of ASR is shown in Fig. 1d. Here, the variable \( Z = [Z_1, \ldots, Z_T] \) is a sequence of time-aligned label indices, describing the time alignment of the augmented label string \( y' = [-, y_1, -, y_2, -, \ldots, - y_T, -] \), where \( Z_t = z \) means that the \( z \)th character in the string is aligned with spectrogram frame \( x_t \). Notice that \( \Pi_t \) is a deterministic function of \( Z_t \), but not vice versa: for example, \( Z_t = 1 \) means that \( \Pi_t = - \), but knowledge that \( \Pi_t = - \) is insufficient to determine \( Z_t \). The assumption shown in Fig. 1d is not compatible with independence between \( Z \) and \( A \), because there are two causal chains connecting \( Z \) to \( A \). Instead, as in [19], counterfactual fairness is achieved by breaking both causal chains, i.e., by finding a latent variable \( U \) such that \( P(Z|U, Y, A) = P(Z|U, Y) \).

As in the previous two sections, the latent variable is unknown; all we know is that the factual speech signal, \( \bar{x} \), and the counterfactual speech signal, \( \bar{x}_{A\rightarrow\bar{a}} \), shared the same latent variable, therefore we require that the conditional distribution \( P(Z|X, Y) \) should be unchanged if \( \bar{x} \) is counterfactually modified to \( \bar{x}_{A\rightarrow\bar{a}} \), i.e.,

\[
P(Z_t = z|X = \bar{x}_{A\rightarrow\bar{a}}, Y = y) = P(Z_t = z|X = \bar{x}_{A\rightarrow\bar{a}}, Y = y).
\]  

(17)

In other words, similar to Eq. (8), the goal is to match the posterior probability of the outcome labels \( (Y \) in Eq. (8), \( Z \) in Eq. (17)) after observing the target outcome \( Y \). With the above assumption, this corresponds to the probability of characters at the softmax layer after observing the true ground truth sequence. In the standard formulation of CTC, this probability is written as \( \gamma_{kt}(\bar{x}, y) = P(Z_t = k|X = \bar{x}, Y = y) \), which is given in terms of the forward variables \( \alpha_t(z) \) and backward variables \( \beta_t(z) \) of CTC loss computation as

\[
\gamma_{kt}(\bar{x}, y) = \frac{\alpha_t(k)\beta_t(k)}{\sum_z \alpha_t(z')\beta_t(z')}.
\]  

(18)

**TABLE I: Duration of male and female speech in hours depending on the data split.**

| Subset   | Male     | Female   | Total |
|----------|----------|----------|-------|
| Train    | 9.8      | 11.2     | 21    |
| Dev      | 2.5      | 2.2      | 4.7   |
| Test     | 2.8      | 1.9      | 4.7   |

**Fig. 2: The adversarial auto-encoder which is used to generate counterfactual examples**

The counterfactual fairness criterion shown in Eq. (17) can therefore be enforced in a multi-task framework by minimizing

\[
\mathcal{L}_{CFPM} = \mathcal{L}_{CTC}(\bar{x}, y) + \lambda_{CT} \mathcal{L}_{CTC}(\bar{x}_{A\rightarrow\bar{a}}, y) + \lambda_{CFPM} \sum_t \sum_k \sum_{\bar{a}} \sum_{\bar{a}'} \mathbb{1}[\bar{a} \neq \bar{a}'](\Delta \log \gamma_{kt})^2
\]  

(19)

where

\[
\Delta \log \gamma_{kt} = \log \gamma_{kt}(\bar{x}_{A\rightarrow\bar{a}}, y) - \log \gamma_{kt}(\bar{x}, y).
\]  

(20)

**IV. EXPERIMENTAL SETTINGS**

We performed our experiments on the Corpus of Regional African American Language (CORAL) [33]. The dataset is split into train, development, and test sets based on the speakers. All speakers in the datasets are alphabetically sorted and the utterances belonging to the first 64 male and 64 female speakers are used for training. From the remaining set, 8 male and 8 female speakers are used in development set and the remaining 14 male and 6 female speakers’ utterances are used for test purposes. Table I summarizes the total duration per gender in terms of hours.

The baseline system is a DeepSpeech2 model [34] trained on the CORAL dataset with CTC loss. Input features are log magnitude spectrograms extracted from 20ms windows with 10ms skip, Hamming window is used for shaping the time-domain data. Our network outputs are English alphabet characters along with blank, apostrophe and the end-of-sentence token. The baseline DeepSpeech2 model has 2 convolutional
layers, each with batch normalization and \text{tanh} activation. The convolution kernel sizes are $41 \times 11$ and $21 \times 11$ respectively. These layers are followed by 5 batch-normalized bidirectional LSTM layers with 768 cells, whose output is fed into a fully connected layer. The baseline model is trained for 30 epochs with Adam optimization, batch-size 16 and learning rate of 0.001. All models are implemented using PyTorch [35] and each one ran on a single Nvidia Tesla V100 GPU.

In order to generate the counterfactual inputs for training, we use an LSTM based adversarial auto-encoder as shown in Fig. 2. This model takes the input audio features ($\tilde{x}$), and encodes them in hidden (latent) vectors denoted by $\tilde{U}$. The adversary is trained to compute an estimated protected attribute ($\hat{A}$) based on $\tilde{U}$, with adversary weights trained to minimize the binary cross-entropy $J(\hat{A}, A)$; the gradient of binary cross-entropy is back-propagated through a gradient reversal layer [36] to remove information about $A$ from the latent vector $\tilde{U}$. The autoencoder appends, to $\tilde{U}$, a dense vector $\mu_a$ representing the protected attribute $A = a$, then passes these concatenated vectors through an LSTM-based decoder which generates a minimum-MSE estimate ($\hat{X}$) of the input feature sequence. In our implementation, the vector $\mu_a$ is computed by taking the average spectrum over all frames of all training utterances belonging to the attribute group $A = a$. The $\hat{A}$ layer, on the other hand, is a one-hot encoding of the attribute group, computed once per utterance by passing the time-averaged LSTM activations of the encoder to the adversary. In the gender case, there are 2 output nodes in the $\hat{A}$ layer: one for male and one for female speakers. Each vector in the input feature sequence $\tilde{x}$ is 161 dimensional, therefore the attribute embedding vectors $\mu_a$ are also 161 dimensional. The encoder consists of two LSTM layers each with 128 cells. The two LSTMs in the decoder have 256 cells and the fully-connected layer generating $\hat{X}$ has 161 output units. The adversarial network that generates $\hat{A}$ consists of a linear layer of size 64, followed by ReLU nonlinearity and another linear layer whose number of nodes depends on the experiment (2 for gender, 10 for age, 43 for education level). This network is trained using batch-size 16, learning rate 0.001, with Adam optimization for 50 epochs on the CORAAL training set. During training matched $\tilde{X} = \tilde{x}$ and true attribute $A = a$ are used. Once this model is learned, counterfactual examples are generated by computing $\hat{X}$ from the abducted latent variable $\tilde{U} = \tilde{u}$ and counterfactual $\mu_a$.

The proposed ASR models are then trained on both factual and counterfactual data. This model is a DeepSpeech2 model similar to the baseline but trained from scratch with the proposed objectives instead of just CTC. In order to compute the counterfactual loss shown in Eq. (10), we pair each factual utterance $\tilde{x}$ with one counterfactual utterance $\tilde{x}_{A \rightarrow \hat{a}}$, where the counterfactual attribute $\hat{a}$ is chosen uniformly at random from the set $\{a: \hat{a} \neq a\}$. The delta terms ($\Delta L_{\text{CTC}}, \Delta \sigma^{-1}, \Delta \log \gamma$) are computed between the reconstructed original features and the counterfactually reconstructed features.

We compared the performances of the ASR systems based on the overall character error rate (CER) on the test data, the CER difference between male and female speakers and also the standard deviation of CER across all test speakers. We tested the significance of the CER difference between models using NIST’s SCLITE toolkit, MAPSSWE method [37] and reported models with significant change at p-value of 0.001 when applicable.

V. Results

In the first set of experiments, our aim is to determine if the middle term in Eq. (16), i.e., the CTC loss due to the counterfactual input ($L_{\text{CF}}$) is crucial. The counterfactual log probability matching model (Eq. (16)) is trained under two conditions, $\lambda_{\text{CF}} \in \{0, 1\}$, each while sweeping the log probability matching weight ($\lambda_{\text{CLM}}$) from 0.01 to 500. Fig. 3 shows the CERs and Fig. 4 shows the CER difference between males and females (M-F), and the standard deviation of CER across test speakers (Stdev). As we observe from Fig. 4,
including the loss term due to the counterfactual input (denoted as $\lambda_{CF} = 1$ in the legend) has a larger gap between gender groups and also a larger standard deviation even though it leads to lower CER in Fig. 3 for $\lambda_{CLM} \geq 100$. Since we are after fair outcomes, we set $\lambda_{CF} = 0$ in the subsequent experiments. Interestingly, irrespective of $\lambda_{CF}$, for most values of $\lambda_{CLM}$ that we have tested, the gap and the standard deviation were higher as compared to the baseline model which was only trained with CTC loss on the original input features. Reduced male-female gap was only observed when we increased $\lambda_{CLM}$ over 100. When we increase $\lambda$ further (to 500 or 1000), then fairness continues to improve, but the average CER becomes worse than the baseline. As we prefer to improve fairness without harming the average CER, our conclusion is that we should choose $\lambda_{CLM} = 200$, a particular setting for which we achieve fairer outcomes than the baseline without hurting the average CER.

In the second set of experiments, our goal is to compare the unadapted baseline, counterfactual log probability matching (CF-LogProb), counterfactual CTC loss matching (CF-CTC) and the log character posterior matching (CF-LogPost) models. As mentioned above, here we set $\lambda_{CF} = 0$ and only sweep the $\lambda$ corresponding to the last term in Eqs. (16), (13) and (19) (respectively $\lambda_{CLM}$, $\lambda_{CCM}$, or $\lambda_{CPM}$). Figure 5 compares the average CER of the four models as a function of $\lambda$. Figure 6 shows the CER gap between male and female and the inter-speaker standard deviation of CER.

In terms of the overall CER (Fig. 5), only the log probability matching approach achieves significantly lower CER than the baseline for most values of $\lambda$. The log posterior matching performs similar to the baseline for small values of $\lambda$ but when $\lambda$ reaches 100, it performs significantly worse than the baseline. On the other hand, the CTC loss matching approach results in a large increase in the CER as we increase the weight of the counterfactual fairness term $\lambda$.

In Fig. 6, we compare the CER gap between males and females as well as the standard deviation of CER over all test speakers. Although the CTC matching approach has a lower gap (fairer) than the other two approaches, it has much higher overall CER (lower accuracy) as we show in Fig. 5. The log probability and log posterior matching approaches have similar performance which are only better than the baseline when $\lambda \geq 100$.

In Fig. 6, we also compare the standard deviation of CERs from different models. As expected, the curves have downward slopes because as we increase the weight of fairness $\lambda$, we should achieve fairer outcomes, which tends to correlate with lower inter-speaker standard deviation. When $\lambda < 1$ all three approaches have higher standard deviation as compared to the baseline and the log posterior matching approach has the smallest deviation. When $\lambda = 100$, CTC and log posterior matching approaches perform better than the baseline but they come at the expense of having higher CER as shown in Fig. 5. Although the log probability matching method has lower male-female gap in Fig. 6 at $\lambda = 100$, we see that the standard deviation of CER is not lower than the baseline. Still, as we discussed above, $\lambda = 200$ is the optimal point for the LogProb setting.

In the experiments described above, the protected attribute was always gender. However, CORAAL dataset comes with speaker metadata including their age and education groups which can also be considered as protected attributes. In the sequel, we investigate the cases where the protected attribute is age or education group rather than gender. In these experiments, we still use the auto-encoder model described in Fig. 2 except that the number of possible attributes changes depending on the experiment. For example, in the age group experiments, we have 10 classes as there are 5 age groups from two genders. Having 10 classes instead of 5 allows us...
to keep the gender attribute the same while generating the counterfactual data from a different age group.

As we can see from Table II, when $\lambda_{CLM} = 200$, we operate at an equal or lower CER level than the baseline while reducing the inter-speaker standard deviation of the CER and the gap between male and female speakers. Although the ASR was trained using age as the protected attribute, the CER difference among age groups is the only measure of fairness that did not improve: inter-speaker standard deviation has improved, and the male-female gap has improved, but inter-age-group differences did not. We speculate that this is a data sparsity issue, caused by the small number of speakers in each age group in the test set (4-5 male and 0-4 female speakers in each age group).

Next, we will investigate the case where the protected attribute is the education level of the speaker. According to the results shown in the lower part of Table II, when we have $\lambda_{CLM} = 200$, we operate at a lower CER than the baseline while having lower inter-speaker standard deviation. For comparison, we also include the male/female and age group statistics in each case. As we can see, when $\lambda_{CLM} = 200$, we are able to reduce the gender gap, as well as the CERs per age group. Since there are around ten education categories, those statistics are not provided in the table. However, if we look at the data, we also observe some decrease in CER for each education category. We will provide further discussion in the next section.

It has been shown that speech data augmentation by text-to-speech models helps reduce error rates (e.g. [38]). Hence, it might be argued that counterfactual training should be performed by simply generating counterfactual training data, and then training the ASR using the augmented dataset; in our notation, this corresponds to setting $\lambda_{CF} = 1$ without any counterfactual matching ($\lambda_{CLM} = 0$). Table III compares two experimental conditions: counterfactual data augmentation ($\lambda_{CF} = 1$, $\lambda_{CLM} = 0$) and counterfactual regularization ($\lambda_{CF} = 0$, $\lambda_{CLM} = 200$). As shown, counterfactual augmentation improves the average CER, but harms fairness (increases both inter-speaker standard deviation and male-female gap). This is because counterfactual data augmentation lowers CER for both the advantaged and disadvantaged groups, but it provides greater benefit for the advantaged group. For example, on the CORAAL dataset, as shown in Table III, female performance is in general better than that of male speakers. Counterfactual data augmentation improves CER more for female than male speakers, hence it increases the male-female gap; it also increases the inter-speaker standard deviation. Counterfactual regularization, by contrast, improves all three metrics. Fairness (standard deviation and the male-female gap) is improved relative to the baseline system, and relative to the counterfactual augmentation strategy. Average CER is improved relative to baseline, but does not beat the average CER achieved by the counterfactual augmentation strategy. While gaining from fairness perspective, we lose some of the advantage that would have been provided by pure data augmentation in this case.

The experiments described above are on the CORAAL dataset. In the next experiment, we will test our method on a standard American English dataset, namely, 100 hr subset of LibriSpeech [39]. This dataset contains only gender information of the speaker; hence, we will only test the performance when the protected attribute is gender. The experimental procedure is similar to the CORAAL dataset except that the generator is now trained on the Librispeech train subset. Table IV shows the CER performance on the LibriSpeech. Since LibriSpeech is a larger dataset with read speech, in general, we operate at a lower CER level as compared to CORAAL. Optimal values of $\lambda_{CLM}$ were found to be slightly higher for Librispeech than for CORAAL; when $\lambda_{CLM} = 300$, we observe reduction in both the overall CER as well as the inter-speaker standard deviation. Furthermore, we reduce the gender gap from 1.7% to 1.4%.

Note that in Librispeech [39] males have lower CER than females (8.9% vs. 10.6% in Table IV), but that the opposite is true in CORAAL (in Table III, male speakers have 44%
TABLE V: CER statistics from the counterfactual dialect experiments on LibriSpeech+CORAAL

| Model                  | CER | Std  | Libri | CORAAL | Diff |
|------------------------|-----|------|-------|--------|------|
| Individual baselines   | 24.8| 15.6 | 9.8   | 38.0   | 28.2 |
| LibriSpeech model      | 29.2| 20.1 | 9.8   | 46.4   | 36.6 |
| CF $\lambda_{\text{CLM}} = 0$ | 23.7| 11.6 | 14.6  | 34.9   | 20.3 |
| CF $\lambda_{\text{CLM}} = 10$ | 21.1| 10.8 | 12.6  | 31.5   | 18.9 |
| CF $\lambda_{\text{CLM}} = 50$ | 23.3| 10.6 | 15.3  | 33.4   | 18.2 |

CER, while female speakers have 32.6%). This result has been previously reported: several studies have reported lower error rates for male than female speakers in standard dialects [18], including standard American English [13], whereas the CORAAL dataset has been shown to have the opposite trend [14]. Two explanations are possible. First, male and female dialects of African American Language (AAL) have been reported to differ [40]; it is possible that women’s AAL poses less difficulty for ASR. Second, though the CORAAL training set has equal numbers of male and female speakers, it has more female speech, and the greater quantity of training data may reduce CER.

We will next present the results on combined LibriSpeech and CORAAL datasets where the protected attribute is dataset, which, among the variables available to us, is the variable most probably correlated with race. LibriSpeech contains speech in a variety of dialects, but the modal dialect is Standard American English; race of speakers is not annotated. Data from CORAAL are assumed to exemplify AAL. Table V shows the CERs of these experiments. The first row shows the results from the individual baselines, i.e., decoding each dataset with its own model. Since LibriSpeech model performs better than CORAAL, in the second row, we show the case where we decode the CORAAL dataset using the LibriSpeech model. Then, we perform counterfactual training for various $\lambda_{\text{CLM}}$. As we can see, without any counterfactual adaptation, we have a very large performance gap for Standard American English versus African American Language. When we train a model on combined CORAAL and LibriSpeech ($\lambda_{\text{CLM}} = 0$), we already see an improvement in terms of fairness, i.e., lower standard deviation across speakers and lower CER difference between groups. If we also apply counterfactual training ($\lambda_{\text{CLM}} > 0$), we further reduce the standard deviation and the CER difference while slightly improving the overall CER.

VI. DISCUSSION

Systems are trained with an individual fairness objective, but results are reported using group disparity (overall male CER vs. female CER). This is mainly because counterfactuals do not really exist. In order to compensate for the lack of factual comparisons between matched individuals, we included the standard deviation across all speakers as a proxy for the individual differences. A full discussion of whether there is a trade-off between individual and group fairness is out of scope of this study and we refer to [41] for a relevant discussion. One way to investigate the improvement in individual differences is to look at the CER differences between a speaker and their counterfactual realizations. Next, we will visualize these results on CORAAL dataset.

In Figs. 7-9, we show the total CER difference between real and counterfactual settings of each protected attribute where the protected attributes are gender, age, and education, respectively. In each case, the left subfigure shows the absolute CER gap from the baseline system and the right figure shows that of the log probability matching system. Colors in the figures are shaded such that identical colors in left and right subfigures correspond to the same absolute CER gap. In all three figures, we see that the model obtained from counterfactual training has lower CER gap among categories, by an order of magnitude.

The counterfactual feature generation step, in this paper, is a type of voice conversion system, but since it is not trained for the task of optimum voice conversion, it does not generate counterfactual speech that would fool a human listener: the differences between the factual and counterfactual spectrograms ($x$ and $x_{\lambda_{\text{CLM}}} \tilde{a}$) are actually quite small. It is therefore necessary to ask whether the counterfactually generated features are really counterfactual, in any meaningful sense: we could pose the null hypothesis that $x_{\lambda_{\text{CLM}}} \tilde{a}$ is simply a version of $x$ perturbed by random noise. It is well known that noise-augmented training can improve ASR (e.g., [42]), but the effect of noise augmentation on fairness has not been previously reported, so we tested it. Table VI summarizes the results of counterfactual log probability matching with $\lambda_{\text{CLM}} \in \{200, 300\}$, in which the counterfactual speech has been replaced by a spectrogram with additive Gaussian noise.
TABLE VI: Counterfactual data generated by Gaussian noise

| Model                  | $\lambda_{CLM}$ | $\sigma_n$ | CER Stdev M F M-F |
|------------------------|------------------|------------|-------------------|
| Noise-regularize       | 200              | $10^{-3}$  | 37.3 8.1 42.9 32.2 10.7  |
| Noise-regularize       | 300              | $10^{-3}$  | 38.7 8.0 44.2 33.7 10.5  |

The noise level is set to $\sigma_n = 10^{-3} \approx \|\tilde{x}_{A-e} - \hat{x}\|_2$, i.e., approximately the RMS difference between factual and counterfactual spectrograms in the experiments of Table III. When we compare Table VI and Table III, we see that random perturbation and counterfactually-generated speech provide similar gains in overall CER (the former is better at $\lambda_{CLM} = 200$, the latter is better at $\lambda_{CLM} = 300$), but that counterfactual training performs better in terms of fairness (both Stdev and M-F difference).

VII. CONCLUSION

In this work, we investigated the ASR training problem from an individualized counterfactual fairness point-of-view. We propose that for any given individual, if this person were from the opposite gender, but spoke the same words with similar style, fair ASR should estimate the same characters as its output. We formulated this as an additional loss term that is added to the CTC loss due to the original input. We compared three approaches: matching the log-softmax outputs from the ASR model, matching the CTC loss and matching the log-posterior of characters given the ground truth sequence. In the experiments on CORAAL, we showed that there is generally a trade-off between the CER and fairness of the system. CTC-matching and log-posterior matching achieved fairness, at the expense of significant increases in average CER, but log probability matching successfully improved both the fairness and the average CER of the recognizer. We verified this observation on gender, age or education level as protected attributes. The same method was also demonstrated to reduce the female-male CER gap in LibriSpeech, and to reduce differences between the error rates of speech samples from the LibriSpeech and CORAAL corpora.

The loss function that involves log-probability matching only assumes that we have (log-)softmax outputs from the network and is independent of the specific model architectures. Hence, as future work, we would like to investigate the use of counterfactual log probability matching for other neural architectures, for both voice conversion and ASR. We especially think that the effect of improved voice conversion on the final ASR performance should be investigated as a future work.

ACKNOWLEDGEMENTS

This work was supported by Institute for Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No. 2019-0-01396, Development of framework for analyzing, detecting, mitigating of bias in AI model and training data). Opinions and findings are those of the authors, and are not endorsed by IITP.

APPENDIX A

LOGITS VERSUS LOG-PROBABILITIES

The logit function is defined to be

$$\sigma^{-1}\left(P(\hat{Y} = 0|X, A)\right) = \log \frac{P(\hat{Y} = 0|X, A)}{1 - P(\hat{Y} = 0|X, A)}$$

$$= \log \frac{P(\hat{Y} = 0|X, A)}{P(\hat{Y} = 1|X, A)} \quad (21)$$

For a binary classifier, equating logits between real and counterfactual inputs would mean matching the log-probability terms $\log P(\hat{Y}|X, A)$: the numerator and denominator of the fraction in Eq. (21) must add up to 1, so fixing the value of Eq. (21) is equivalent to fixing both the numerator and denominator. Now, if we consider ASR or specifically the character recognition problem, we have a multi-class classifier $(K > 2)$ and the logit terms become

$$\sigma^{-1}\left(P(\hat{Y} = k|X, A)\right) = \log \frac{P(\hat{Y} = k|X, A)}{1 - P(\hat{Y} = k|X, A)}$$

$$= \log \frac{P(\hat{Y} = k|X, A)}{P(\hat{Y} \neq k|X, A)} \quad (22)$$

If we achieve the perfect match of the logits ($\forall k \in \{1, \ldots, K\}$), then this would again imply equality of log-probabilities resulting from the real and counterfactual inputs. However, during training the difference between real and counterfactual outputs will not be 0 for all $k$. Since the goal is to match probabilities and since it is also easier to compute the log-softmax as compared to the logits for the multi-class case, we use the log-softmax outputs ($\log P(\hat{Y} = k|X, A)$) rather than the logits in our experiments.

REFERENCES

[1] C. Dwork, M. Hardt, T. Pitassi, O. Reingold, and R. Zemel, “Fairness through awareness,” in Proceedings of the 3rd Innovations in Theoretical Computer Science Conference, 2012, pp. 214–226.

[2] A. Chouldechova, “Fair prediction with disparate impact: A study of bias in recidivism prediction instruments,” Big data, vol. 5, no. 2, pp. 153–163, 2017.

[3] A. Chouldechova and M. G’Sell, “Fairer and more accurate, but for whom?” arXiv preprint arXiv:1707.00046, 2017.

[4] P. Bickel, E. Hammel, and J. O’Connell, “Sex bias in graduate admissions: Data from berkeley,” Science, vol. 187, no. 417S, pp. 398–404, 1975.

[5] M. Srivastava, H. Heidari, and A. Krause, “Mathematical notions vs. human perception of fairness: A descriptive approach to fairness for machine learning,” in ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2019.
[6] J. Pearl, *Probabilistic Reasoning in Intelligent Systems*. San Francisco, CA: Morgan Kaufman, 1988.

[7] S. A. Friedler, C. Scheidegger, and S. Venkatasubramanian, “On the (im)possibility of fairness,” 2016.

[8] M. J. Kusner, J. Loftus, C. Russell, and R. Silva, “Counterfactual fairness,” in *Advances in Neural Information Processing Systems*, 2017, pp. 4066–4076.

[9] S. Dash and A. Sharma, “Counterfactual generation and fairness evaluation using adversarially learned inference,” *arXiv preprint arXiv:2009.08270*, 2020.

[10] J. Joo and K. Kärkkäinen, “Gender slopes: Counterfactual fairness for computer vision models by attribute manipulation,” *arXiv preprint arXiv:2005.10430*, 2020.

[11] D. Kaushik, E. Hovy, and Z. C. Lipton, “Learning the difference that makes a difference with counterfactually-augmented data,” in *Proc. International Conference on Learning Representations ICLR*, 2020.

[12] P.-S. Huang, H. Zhang, R. Jiang, R. Stanforth, J. Welbl, J. Rae, V. Maini, D. Yogatama, and P. Kohli, “Reducing sentiment bias in language models via counterfactual evaluation,” in *Findings of the Association for Computational Linguistics: EMNLP*, 2020, pp. 65–83.

[13] R. Tatman, “Gender and dialect bias in youtube’s automatic captions,” in *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, 2017, pp. 53–59.

[14] A. Konecne, A. Nam, E. Lake, J. Ndelle, M. Quaerty, Z. Mengesha, C. Toops, I. R. Rickford, D. Jurafsky, and S. Goel, “Racial disparities in automated speech recognition,” *Proceedings of the National Academy of Sciences*, vol. 117, no. 4, pp. 7684–7689, 2020.

[15] D. Povey and G. Saon, “Feature and model space speaker adaptation with full covariance gaussians,” in *Ninth International Conference on Spoken Language Processing*, 2006.

[16] G. Saon, H. Soltan, D. Nahamoo, and M. Picheny, “Speaker adaptation of neural network acoustic models using i-vectors,” in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2013, pp. 55–59.

[17] K. Vesely, S. Watanabe, K. Žmolíková, M. Karafiát, L. Burget, and J. H. Černocký, “Sequence summarizing neural network for speaker adaptation,” in *IEEE International conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2016, pp. 5315–5319.

[18] M. Garnerin, S. Rossato, and L. Besacier, “Gender representation in french broadcast corpora and its impact on asr performance,” in *Proceedings of the 1st International Workshop on AI for Smart TV Content Production, Access and Delivery*, 2019, pp. 3–9.

[19] S. R. Pföhl, T. Duan, D. Y. Ding, and N. H. Shah, “Counterfactual reasoning for fair clinical risk prediction,” in *Machine Learning for Healthcare Conference*, 2019, pp. 325–358.

[20] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, “Connectionist temporal classification: labelling unsegmented data with recurrent neural networks,” in *Proceedings of the 23rd International Conference on Machine Learning*, 2006, pp. 369–376.

[21] A. Graves, A.-R. Mohamed, and G. Hinton, “Speech recognition with deep recurrent neural networks,” in *IEEE International conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2013, pp. 6645–6649.

[22] A. Graves, “Sequence transduction with recurrent neural networks,” *arXiv preprint arXiv:1211.3711*, 2012.

[23] W. Chiu, N. Jaitly, Q. Le, and O. Vinyals, “Listen, attend and spell: A neural network for large vocabulary conversational speech recognition,” in *IEEE International conference on acoustics, speech and signal processing (ICASSP)*, 2016, pp. 4960–4964.

[24] T. Horri, S. Watanabe, Y. Zhang, and W. Chiu, “Advances in joint CTC-attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM,” in *Proc. ISCA Interspeech*, 2017, pp. 949–953.

[25] H. Soltan, H. Liao, and H. Sak, “Neural speech recognizer: Acoustic-to-word lstm model for large vocabulary speech recognition,” in *Proc. ISCA Interspeech*, 2017, pp. 3707–3711.

[26] A. Chouldechova and A. Roth, “The frontiers of fairness in machine learning,” *arXiv preprint arXiv:1810.08810*, 2018.

[27] V. Nazeran and J. Rubin, “Fairness definitions explained,” in *2018 IEEE/ACM International Workshop on Software Fairness (fairware)*. IEEE, 2018, pp. 1–7.

[28] D. Madras, E. Creager, T. Pitassi, and R. Zemel, “Fairness through causal awareness: Learning causal latent-variable models for biased data,” in *FAT’19: Proceedings of the Conference on Fairness, Accountability and Transparency*, 2019, pp. 349–358.

[29] M. Hardt, E. Price, and N. Srebro, “Equality of opportunity in supervised learning,” in *Advances in Neural Information Processing Systems*, 2016, pp. 3315–3323.