Debiasing knowledge graph embeddings

Joseph Fisher
Amazon Alexa
fshjos@amazon.co.uk

Arpit Mittal*
Facebook
arpitmittal@fb.com

Dave Palfrey
Amazon Alexa
dpalfrey@amazon.co.uk

Christos Christodoulopoulos
Amazon Alexa
chrchrs@amazon.co.uk

Abstract

It has been shown that knowledge graph embeddings encode potentially harmful social biases, such as the information that women are more likely to be nurses, and men more likely to be bankers. As graph embeddings begin to be used more widely in NLP pipelines, there is a need to develop training methods which remove such biases. Previous approaches to this problem both significantly increase the training time, by a factor of eight or more, and decrease the accuracy of the model substantially. We present a novel approach, in which all embeddings are trained to be neutral to sensitive attributes such as gender by default using an adversarial loss. We then add sensitive attributes back on in whitelisted cases. Training time only marginally increases over a baseline model, and the debiased embeddings perform almost as accurately in the triple prediction task as their non-debiased counterparts.

1 Introduction and Related Literature

Learning embeddings of knowledge graph entities and relations is becoming an increasingly common first step in utilizing knowledge graphs for a range of graph and NLP tasks, from missing link prediction, (Bordes et al., 2013; Trouillon et al., 2016), to more recent methods integrating learned embeddings into language models, (Zhang et al., 2019; IV et al., 2019; Peters et al., 2019).

In (Fisher et al., 2020), it is shown that knowledge graph embeddings encode similar social biases to those observed in word embeddings ((Bolukbasi et al., 2016; Caliskan et al., 2017; Garg et al., 2017)), such as the information that men are more likely to be bankers and women more likely to be nurses. This is an unsurprising finding, given that the distribution of entities in knowledge graphs is highly skewed towards historically privileged members of society; there are many more male bankers in Wikidata than female bankers.

Such biases are potentially harmful as they can propagate to downstream tasks. If graph embeddings are used for knowledge base completion, the model would be less likely to be able to predict a female bankers profession than an equivalent male banker's profession. Alternatively, if graph embeddings are used as input to a Transformer (Vaswani et al., 2017) encoder as in (Peters et al., 2019), the same effects on coreference resolution, entity linking and other downstream task as have been observed with word embeddings will re-occur.

In light of this, it is important to develop methods which enable debiasing of the embeddings with respect to user-defined sensitive attributes (e.g. gender). A potential method for debiasing was presented in (Bose and Hamilton, 2019), in which the authors train a set of filter neural networks to remove sensitive information from embeddings. Although the method proves effective on the MovieLens1M dataset, it results in a significant drop in performance as measured on the triple prediction task for the FB15K dataset, and proves ineffective in removing more than one source of bias concurrently. In addition, our benchmarks indicate the extra computation needed to train the neural network filters increase overall training time by a factor of 8 or more, making the approach unsuitable for large knowledge graphs.

We present an alternative approach, which trains all embeddings to be neutral with respect to sensitive attributes such as gender by default using an adversarial loss. We then allow the user to add sensitive information back in for whitelisted cases. For example, we may allow the model to use nationality information when predicting the languages someone speaks. We evaluate the model on FB15K, FB3M and Wikidata, and show that it is significantly faster than previous approaches, less disrup-

*Work completed whilst at Amazon

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 7332–7345,
November 16–20, 2020.

tive to the model accuracy on the original triple prediction task, and effective at producing embeddings which are neutral with respect to user-defined sensitive attributes (we present two measures to evaluate the level of sensitive information which remains in the trained embeddings).

2 Knowledge Graph Embeddings

A knowledge graph is a set of facts in triple form, where a triple consists of two entities and a relation, e.g. (France, has_capital, Paris). The aim of graph embedding methods is to use these triples to learn a continuous vector representation of dimension d of all entities and relations. The standard approach defines a score function, $g(.)$, which transforms a triple in vector form to a scalar score denoting how likely this triple is to be correct. For example the function

$$s = g(E_1, R_1, E_2)$$

gives the score, s, that the triple composed of entities 1/2 and relation 1 is correct, where $E_{1/2}$ and R_1 are all embeddings of dimension d. The score function is generally composed of a transformation, which takes as input one entity embedding and the relation embedding and outputs a vector of the same dimension, and a similarity function, which calculates the similarity or distance between the output of the transformation function and the remaining entity embedding.

Transformation functions proposed in the literature include TransE (Bordes et al., 2013), ComplEx (Trouillon et al., 2016) and RotatE (Sun et al., 2019). In this paper we use the TransE function and the dot product similarity, though the debiasing methods are applicable to any choice:

$$S =< E_1 + R_1, E_2 > \quad (1)$$

2.1 Optimization of knowledge graph embeddings

Knowledge graph embeddings (in their basic form, with no debiasing) are trained by optimizing the entity and relation embeddings to produce a high score for positive (true) triples, and a low score for randomly generated false triples. This is illustrated in Figure 1, with a batch of three triples shown in Box 1. We calculate the scores of the positive triples using Equation 1 (shown in Box 2a), and then for each positive triple calculate the scores of N negative triples, with negatives created by randomly permuting the entities on either side; we permute the right hand side (rhs) of T_1 with $N = 2$ in Box 3 of the example figure. For the standard model (i.e. no debiasing), we pass the scores of the single positive triple and the N negative triples through the softmax function (denoted "sft" in the figure) and calculate the cross-entropy[1], denoted L_{CE}, between the resulting distribution and that with all the weight on the positive triple (Box 4). The steps can be summarized using Figure 1 as Boxes 1 $\rightarrow$ 2a $\rightarrow$ 3 $\rightarrow$ 4 $\rightarrow$ 5a. This is the standard approach for training graph embeddings, which we denote "Basic" in the results tables, and on top of which we add our debiasing techniques.

3 Debiasing Motivation

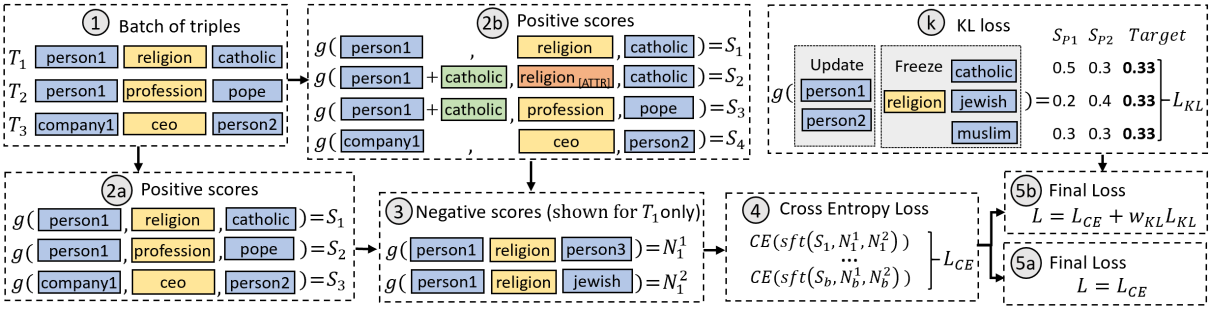
To motivate our work, we begin by introducing how biases may be encoded into the embeddings of human entities. First, we define a set of "sensitive attributes"; human characteristics which may be associated with unwanted stereotypes. In this paper we define gender, ethnicity, religion and nationality as "sensitive attributes", though any choice is possible and we do not claim this list to be exhaustive/correct. For each knowledge graph, there will be a set of relations which provide these attributes (e.g. for Freebase the relation "/people/person/gender" provides a person's gender), which we term "sensitive relations".

When embeddings are trained with positive triples such as (person1, gender, male), the embedding of person1 will be updated with information related to the rhs entity "male" in order to score this triple higher than negative triples, including (person1, gender, female). This in itself is uncontroversial - we do not mind if the model is able to predict a person's gender. However, as gender information is now encoded in the embedding of person1, the model is also able to use this information when scoring other triples, such as (person1, profession, banker). (Fisher et al., 2020) shows that as knowledge graphs such as Wikidata and Freebase include, for example, many more male bankers than female bankers, the model learns to use the encoded gender information when predicting the likelihood a person is a banker, alongside other harmful stereotypes.

As knowledge graphs are based in reality, it is not easy to mitigate this effect by manually balanc-

[1]Common alternatives are a ranking or logistic loss which also incentivize a high score for positive triples and a low score on negatives.

Figure 1: Training of a single batch with KL loss and attribute vectors



ing the graph[2]. Instead we aim to train all human's embeddings to be neutral with respect to sensitive attributes. That is, we wish to make it impossible to predict, for example, a person's gender, from their embedding. As a result, predictions made using these embeddings (such as about profession) will also be independent of these attributes. Note this imposes the constraint that we can no longer predict any unknown gender, religion, ethnicity or nationality of a human.[3]

4 Debiasing Architecture

4.1 Adversarial loss

A potential approach to avoiding information related to sensitive attributes being encoded in human's embeddings is to remove all triples containing sensitive relations from the training data. In Appendix A.3 , we show this is insufficient; a person's gender can often be predicted regardless, due to correlated relations. An alternative approach, introduced by (Bose and Hamilton, 2019), is to train a set of neural network "filters" to remove sensitive information from embeddings. We show (Tables 5 and 6) this approach to be ineffective at removing information about multiple sensitive attributes concurrently, as the output of each filter network is independent of one attribute only, leading to leaking of information when their outputs are averaged (see Appendix A.1 for details).

Instead, we leave the sensitive relations in the training data, and optimize their embeddings as normal.[4] We then add a Kullback-Leibler Divergence (KL-Divergence) based loss function to the model, which aims to make it impossible to make accurate predictions about these relations (e.g. about someone's gender). This addition to the "Basic" model is illustrated in Box k of Figure 1. During training, for each batch we extract the embeddings of all human entities in the batch; in Figure 1 these are denoted "person1" and "person2". We then calculate the score, S of these entities with each sensitive relation for each of the top M most frequent right hand side entities.

$$S_{p,j,m} = g(E_p, R_j, E_m)$$

where E_p denotes the embedding of Person p, R_j the embedding of sensitive relation j, and E_m the embedding of the rhs entity m. In the example Figure, we set $M = 3$, and use "religion" as the sensitive relation. In practice, we set M to 30 in all experiments, and define "top" as being the entities with the largest counts in the dataset. In the case of gender, for which there are only two rhs entities in the knowledge graph with significant counts, we simply try to balance the scores of the top two genders.

For each person i in the batch, we pass the scores for the top M right hand side entities through the softmax function. The KL-divergence is then calculated between this distribution and a target distribution, $\mathcal{G}$. In this paper, we use a balanced target distribution of weight $\frac{1}{M}$ (e.g. if $M = 3$ the distribution $\mathcal{G} = [0.33, 0.33, 0.33]$) and is denoted L_{KL}.

$$L_{KL} = \frac{1}{P}\frac{1}{J}\sum_{p=1}^{P}\sum_{j=1}^{J} KL(\mathcal{G}_j, sft(S_{P_{i,j,m}}))$$

In other words the KL loss is incentivizing the model, for the case of religion, to give an equal probability to a person having each of the top M religions (hence making it impossible for the model to predict their true religion). Note that the target distribution G does not need to be balanced, and

[2]There are no female U.S. Presidents in history, so we cannot balance this profession without inventing fake people or deleting male Presidents, which would distort/decimate the training data respectively.

[3]Given such relations are often in knowledge graphs, and that when they aren't predicting them is likely to be controversial, this is a small cost.

[4]That is, we update the embeddings of the relations for gender, religion etc. to attempt to enable the model to accurately predict someone's gender or religion.

can be defined by the user to put more/less weight on particular attributes.[5]

When minimizing L_{KL} we freeze the embeddings of the relations and the rhs entities, both of which have been trained using L_{CE} (i.e. to be effective at predicting an entity's sensitive attributes correctly), and update only the human entity embeddings. The KL loss and the original graph embedding loss are consequently trained adversarially to each other for the sensitive relations only.

The final loss (Box 5b) is a weighted average of the original cross-entropy loss, L_{CE} and L_{KL};

$$L = L_{CE} + w_{KL} L_{KL}$$

The weight w_{KL} controls how much emphasis we put on debiasing vs. the original triple prediction task. A discussion of the procedure for choosing w_{KL} follows in Section 4.3. We denote models trained with the KL loss included as "KL" in the results tables, and they can be summarized in Figure 1 as Boxes 1 $\rightarrow$ 2a $\rightarrow$ 3 $\rightarrow$ 4, k $\rightarrow$ 5b.

4.2 Attribute vectors

One limitation of this approach is that it prevents the model from using sensitive information (e.g. gender) for all triples. In some uncontroversial cases (e.g. predicting somebody's singing voice) we may wish to allow the use of such information. The second component of our architecture, a set of attribute vectors, facilitates this.

To illustrate, we label a set of whitelisted triples, for which we allow information from the sensitive relations to be used. We define such cases in two groups. Firstly, a set of relations for which we allow a particular sensitive attribute to be used for **all** entities. For example, when scoring the likelihood of the triple (person1, speaks_languages, french), we allow the model to use a person's nationality. We labelled a separate set of such relations for gender, religion, ethnicity and nationality, giving a total of 60 relations for Freebase and 88 for Wikidata, with examples in Appendix A.2 .

Secondly, for some relations we may only wish to allow sensitive attributes to be used for particular right hand side entities. For example, we may allow the religious attribute to be used when making a prediction of the likelihood of the triple (person1, profession, nun), but not allow it to be used when predicting the triple (person1, profession, banker). We labelled data for only one such relation (denoting someone's profession), with a total of 128 professions whitelisted for Freebase and 1411 for Wikidata. Details on labelling these professions can be found in Appendix A.2 alongside examples.[6]

4.2.1 Attribute vector training

To allow the model to use sensitive information in the whitelisted cases, for each right hand side entity of a sensitive relation (i.e. for each of the entities male, female, Catholic, Jewish etc.) we train a vector of the same dimension as the graph embeddings, termed an "attribute vector". For whitelisted triples, we can add this vector onto the human's embedding, allowing the model to utilize sensitive information.

This addition to the model is illustrated by replacing box 2a with 2b in Figure 1. We add the attribute vectors, shown as green rectangles, in two distinct cases. Firstly, for the whitelisted (for religion) triple T_2, predicting whether person1 is the pope. This allows the model to use the information that person1 is a Catholic when scoring this triple. Secondly, to aid in training useful attribute vectors only[7] we add a new triple for each triple in the batch which contains a sensitive relation (in this case T_1), replacing the relation with a twin denoted "religion$_{[\text{ATTR}]}$", shown in orange in Figure 1. In doing so, we incentivize the attribute vector to encode information about the correct sensitive attribute (in this case Catholicism), which in turn helps with predictions of whitelisted triples. Duplicating the sensitive relations is necessary, as the original sensitive relations are trained adversarially against the KL loss in Box 3, with no attribute vector added to the left hand side. The model with attribute vectors included is denoted "KL + Attr." in the results tables, and can be summarized using Figure 1 as Boxes 1 $\rightarrow$ 2b $\rightarrow$ 3 $\rightarrow$ 4, k $\rightarrow$ 5b.[8]

[5] E.g. for religion, one may wish to place particular emphasis on not being able to predict a believer vs. non-believer, grouping religions and defining a distribution accordingly.

[6] As with which attributes are denoted "sensitive", the decision about which relations and professions should be whitelisted with respect to these attributes is non-trivial and application dependent, and can be set by the user.

[7] i.e. we discard these relations after training as it is nonsensical predicting someone's religion when we know it.

[8] Note it is possible to freeze the embedding of person1 in the triples in rows 2 and 3 of Box 2, to avoid them updating with religious information. In practice, we found doing so was not useful, as unfrozen versions of these entities which showed up in negative samples resulted in skewed embeddings. Instead, we rely on the KL loss to enforce debiasing.

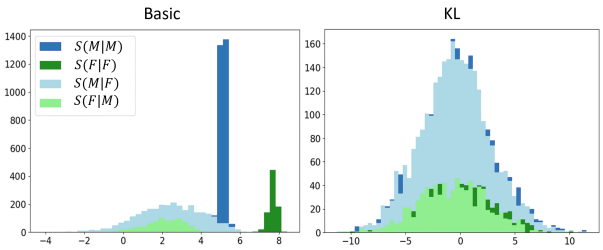
4.3 Methods for tuning W_{KL}

The model now includes two loss functions, as shown in Box 5b. The scalar weight w_{KL} controls the emphasis the model puts on debiasing vs. the original prediction task. In order to choose w_{KL} for each attribute[9], we introduce two methods of measuring the bias in the trained embeddings.

Note that all "tuning" results in this section (Figures 2, 3 and 4) are for the train (in sample) data. The motivation for this is that generally all entities appear in at least one triple in the train set (otherwise the model is not capable of producing an embedding for that entity). We wish to test whether the resulting trained embeddings of human entities contain sensitive information which could potentially be used in downstream tasks, including predicting new triples or as additional input to a language model etc. More precisely, we tune w_{KL} using the subset of human entities for which the triple (person, has_sensitive_attribute, sensitive_attribute) is in the training set. This ensures that even if this information is present in the training data, the debiasing is effective at removing it from the embedding.

To begin, we analyse the model's scores for each human entity when predicting each sensitive attribute (without the attribute vectors added on). The KL loss attempts to ensure that these scores are equalized. For example, the scores of (person1, gender, male) and (person1, gender, female) should be equal, so that we cannot predict a person's gender using the score function.

Figure 2: FB15K gender scores (in sample) for TransE model



In Figure 2, $S(F|F)$ denotes the score that a female entity is female, and $S(F|M)$ denotes the score a female entity is male. If the model is able to correctly identify female entities' gender from their embeddings, $S(F|F)$ should be greater than $S(F|M)$, as is clearly the case for the "Basic" model on the left.[10] For the "KL" model, shown on the right, the distributions of the scores overlap, as incentivized by the KL loss, indicating that the model now struggles to identify a person's gender.

To extend this analysis to all sensitive attributes, we calculate the difference between the score for a person's true attribute, and the top n false attributes. For example, in the case of a Catholic entity, we would calculate the score for the triple (person1, religion, catholic), and the scores (person1, religion, R) for all of the top 30 most frequent religions, R. We then calculate the difference between the true triple's score and the average score of the false triples.

Figure 3: Tuning of w_{KL} for FB15K, using the scores of triples with sensitive relations

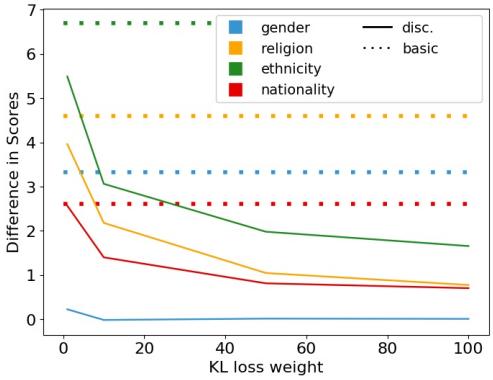


Figure 3 displays the results. The y-axis denotes the difference in scores, with the dotted horizontal lines giving the difference for the "Basic" model with no debiasing; the dotted line for gender therefore corresponds to the difference between the lighter and darker histograms on the left of Figure 2. The x-axis denotes the weight on the KL loss, w_{KL}. The solid coloured lines show the differences for the "KL" model. As we increase w_{KL}, more emphasis is put on reducing the sensitive information in the embeddings, leading to a reduction in the difference for all attributes.

For gender, it is relatively easy for the model to equalize the scores, as there are only two genders in FB15K. Equalization for TransE therefore corresponds to simply placing the sum of a human entity's embedding and the relation gender equidistant between the two gender embeddings. For the

[9] We can choose a different value of w_{KL} for gender, religion, ethnicity etc.

[10] The higher scores for female entities than male entities stems from a combination of the skewed distribution - there are more male entities than female entities in FB15K - and negative sampling. We are more likely to get unwanted positive triples in the negative samples for male entities, as there are more of them.

other three sensitive attributes, there are multiple rhs entities, and ensuring equidistance to each of them is no longer plausible. As a result, even as we increase w_{KL} to 100.0, the difference in scores approaches an asymptote. However, we can clearly see that the difference continues to decrease significantly for the higher values of w_{KL} for religion, ethnicity and nationality, suggesting a high weight is necessary.

As a second method of measuring the extent to which sensitive information remains in the trained embeddings, we train a feedforward neural network to try and predict the attributes of a human entity from their embedding alone. That is, the input to the network is the embedding of dimension d, and the output is a softmax distribution over labels (male and female for gender). We train the network using the cross-entropy loss between the output distribution and the correct class label. We use a single hidden layer of dimension 300 and ReLu activation function.

Figure 4: FB15K tuning of KL weight based on feedforward NN predictions

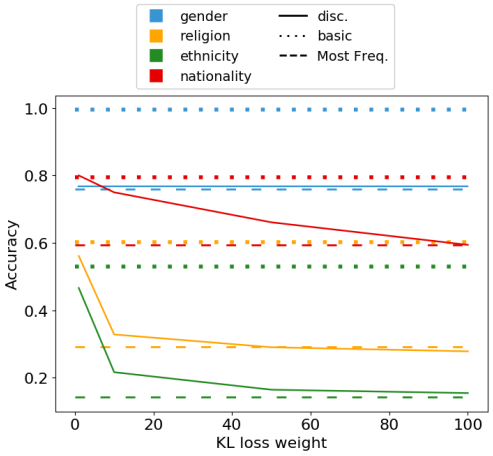


Figure 4 displays the results. The accuracy from predicting the most frequent class is illustrated with a dashed horizontal line. If the network is unable to extract any useful predictive information from the embedding, it will default to this. The accuracy when using the embeddings from the basic TransE model are shown with the horizontal dotted lines. For gender, we can see that the neural network achieves almost 100% accuracy in its predictions for the "basic" case. As before, a weight of 1.0 proves sufficient to remove the gender information from the embeddings. For religion, nationality and ethnicity a higher weight is needed, with the accuracy only approaching the most frequent case at weights of 50.0 and over.

The above methods of measuring the extent to which sensitive information remains in the embeddings suggest a high value of w_{KL} is necessary. However, this is not costless - a higher weight on the KL loss results in the model putting less emphasis on the original triple prediction task, and embeddings which are less informative. To illustrate, Table 1 shows the Mean Reciprocal Rank (MRR)[11] for both the KL only models and the KL + Attr. model on the FB15K test set, at different values of w_{KL}.

Table 1: MRR on FB15K test set for different KL loss weights

		Weight			
	Model	1.0	10.0	50.0	100.0
Biased	Basic		0.680		
Debiased	KL	0.673	0.660	0.658	0.654
	KL + Attr.	0.675	0.672	0.663	0.660

As we increase the weight, the MRR falls from the baseline of 0.680 to a minimum of 0.654 for the KL model. Part of this drop is regained in each case by the attribute vectors, but not the entire gap.

In light of Figures 3 and 4 we set w_{KL} to 1.0 for gender and to 100.0 for religion, ethnicity and nationality. That is, for the purposes of presenting our method in this paper, we tune w_{KL} to remove as much sensitive information as possible, using the results presented on the train set only. In practice w_{KL} should be tuned by the researcher for the specific dataset they are using and depending on the importance they place on debiasing vs. model accuracy. If this approach is taken, the validation set MRR should be monitored, and one may allow some bias to remain if validation MRR drops with high w_{KL}.

We note that the tuning step increases the computational cost significantly; a separate model has to be trained for each value of w_{KL} experimented with. However, the results presented here, and in Appendix A.4 for FB3M, suggest that this step is necessary, as the value of W_{KL} chosen varies across both sensitive attribute and datasets.

5 Experimental Details

5.1 Datasets

We evaluate our model on three datasets; FB15K, FB3M (both of which are subsets of the full Free-

[11] See Section 6.2 for a description of the MRR

base knowledge graph), and Wikidata. For FB3M there is no standard train/val/test split in the literature, so we randomly subsample 10,000 triples as a validation set, and 100,000 triples as a test set. Note that (as discussed at the start of Section 4.3) we use the train set for tuning W_{KL}, and as a result, do not use the validation splits in this paper, though they could be used by a practitioner for monitoring the MRR during tuning. For Wikidata, we first filtered out all triples which contained a string entity (as opposed to an entity with a wiki QID), and then removed all relations/entities which had fewer than 5 observations. This left a total of 283M triples, from which we randomly sampled a test set of size 200,000.

Table 2: Dataset statistics

Dataset	Ents	Rels	Train	Val	Test
FB15K	14.9K	1.3K	483K	50K	59K
FB3M	3M	6.6K	24M	10K	100K
Wikidata	20M	1.1K	283M	—	200K

As discussed in Section 4.2, our approach assumes that we will never predict the sensitive attributes of a person (their gender, ethnicity etc.) directly. To evaluate our model, we therefore remove all sensitive relations which provide these attributes from the validation and test datasets; roughly 2% of triples. We denote the resulting datasets as FB15K (filtered), FB3M (filtered) and Wikidata (filtered).

5.2 Hyperparameters

We use the AdaGrad (Duchi et al., 2011) optimizer with a learning rate of 0.1, and perform linear learning rate warmup over epoch 1. We train for 50 epochs for FB15K and FB3M and 10 for Wikidata. Training is implemented using the PyTorch-BigGraph library (Lerer et al., 2019). For the Bose and Hamilton (2019) comparison we use the author's opensource code and the same model (TransE) and hyperparameters as our work, with the filter network dimensions to the recommended levels, and a low value of gamma of 10.0, to try to match the accuracy of our model.

6 Results

We present results from three perspectives: speed (training time), accuracy (ability to predict correct triples given the embeddings), and debiasing (how much sensitive information remains in the trained embeddings).

6.1 Training Time

We present the training time per epoch relative to the basic TransE model. All models were trained using the PTBG framework on a desktop with an Intel Core i7-7700 CPU with 8 cores. Table 3 displays the results for FB15K. The "Basic" model takes 68 seconds per epoch (spe). For each debiased approach we use the whitelisted labels described in Section 4.2 to denote which entities need to be debiased.[12] The additional neural networks in (Bose and Hamilton, 2019) push the training time per epoch to 533.3 seconds, around an 8x increase.

Table 3: Per epoch model training times for FB15K

	Model	Seconds per epoch (spe)
Biased	Basic	68.4
	Bose & Ham.	533.3
Debiased	KL	71.0
	KL + Attr.	89.4

Next, we benchmark the speed of the discriminator and KL-loss only ("KL"). As this works through the model's own score function, we can group the training of the sensitive relations with existing batches of triples, meaning the hit to computation time is minimal, increasing to 71.0 spe.

Finally, we evaluate the model with the attribute vectors as well ("KL + Attr."), which we train concurrently. Despite being a simple calculation (addition of vectors), there is a computational cost from indexing which entities we need to add each additive vector to, resulting in a time of 89.4 spe.

Although these times are benchmarked on FB15K only, the relative differences in spe stays constant for the larger datasets, as time to read from/write to memory is negligible. Consequently, we are able to train our full model on the Wikidata knowledge graph for 10 epochs in a time of around 10 hours on a system with 64 cpus.

6.2 Triple Prediction

We evaluate the accuracy of the embeddings on the triple prediction problem, where we aim to predict the likelihood of a triple being correct by calculating its score relative to negative triples. As is common in the literature (Bordes et al., 2011, 2013) we report results in terms of the Mean Reciprocal Rank (MRR), hits@1, hits@10 and hits@50. For FB15K we replace either the lhs or rhs of the triple

[12] In general the majority of triples with humans require debiasing with respect to at least one sensitive attribute.

with all remaining entities, whereas for FB3M and Wikidata we randomly sample 50000 negative entities.

Throughout the remaining results, for FB15K we use the tuned values of w_{KL} described in Section 4.3. For FB3M and Wikidata, we found these values to be too small. Consequently, we increased the values for FB3M to 100.0 for gender and 500.0 for the other three attributes. See Appendix A.4 for a full discussion. For Wikidata, we didn't rerun the experiments due to the computational cost.

Table 4: Test set results for TransE embeddings

	Model	MRR	h@1	h@10	h@50
FB15K (filtered)					
Biased	Basic	0.680	0.555	0.871	0.937
Debiased	Bose & Ham.	0.426	0.300	0.655	0.821
	KL	0.671	0.534	0.853	0.924
	KL + Attr.	0.679	0.537	0.861	0.931
FB3M (filtered)					
Biased	Basic	0.684	0.612	0.794	0.843
Debiased	KL	0.682	0.611	0.792	0.840
	KL + Attr.	0.684	0.611	0.798	0.846
Wikidata (filtered)					
Biased	Basic	0.493	0.380	0.703	0.837
Debiased	KL	0.485	0.373	0.693	0.827
	KL + Attr.	0.495	0.383	0.705	0.837

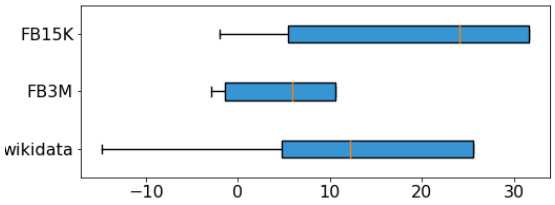
Table 4 displays the results. The "Basic" model achieves an MRR of 0.680 and hits@10 of 0.871 for FB15K. The neural network based filters approach of (Bose and Hamilton, 2019) significantly reduces these metrics, to 0.426 and 0.655 respectively. The "KL" approach leads to a much smaller drop in MRR and hits@10, to 0.671 and 0.853 respectively for FB15K, with a drop of similar magnitude for FB3M and Wikidata.

This result is expected; by limiting the model's ability to stereotype based on gender, religion, nationality and ethnicity, we expect to do worse at predicting both controversial relations such as profession, and whitelisted ones such as speaks_languages.

The aim when adding the attribute vectors ("KL + Attr.") for whitelisted relations is to recover some of this drop. We see a small increase of the MRR from 0.671 to 0.679, and of the hits@10 from 0.853 to 0.861 for FB15K, with similar results for the two larger datasets. We do not get back to the biased TransE accuracy for FB15K, but exceed it slightly for FB3M and Wikidata.

To understand if the attribute vectors are effective at increasing performance on the whitelisted relations, Figure 5 shows the distribution of the percentage changes in test set MRR for each whitelisted relation when we move from the "KL" model to the "KL + Attr." model[13]. That is, for each whitelisted relation, such as speaks_languages, we calculate the test set MRR **for this relation only** for both the "KL" model and the "KL + Attr." model, and take the percentage difference between the two scores. Figure 5 shows the distribution of these differences for all whitelisted relations, with the box spanning from the lower to upper quartiles, and the whiskers at the 5th and 95th percentiles.

Figure 5: Percentage increase in MRR for whitelisted relations when using attribute vectors



We see a substantial improvement in the model's ability to predict triples with whitelisted relations when adding the attribute vectors back in, with a median improvement of around 25% for FB15K, 6% for FB3M and 12% for Wikidata. With the KL loss only, it will be very hard to predict, for example, the languages that someone speaks when their embedding is independent of nationality. By adding the nationality information back in via an attribute vector, this prediction becomes simpler.

6.3 Debiasing

To evaluate the debiasing we present the same metrics introduced in Section 4.3, in which we tuned the values of w_{KL} for FB15K.[14] For (Bose and Hamilton, 2019), we give two sets of results. Firstly, denoted with [s], their model with just the single relevant filter network applied (the "gender" filter for the **Gender** column etc.), and secondly with all four filters applied, denoted [a]. For both variants, the scores of the correct attributes are consistently higher than the incorrect attributes, with the result particularly clear for the [a] variant. We discuss the reasons for this in Appendix A.1 .

[13] We include only the whitelisted relations that have more than 5 observations in the test datasets.

[14] As such the FB15K results in Tables 5 and 6 mirror those in Section 4.3.

In contrast, our model reduces the difference in scores for a human entity's true sensitive attributes and the alternatives (shown in Table 5) significantly for FB15K and FB3M. For Wikidata, we still see a notable gap between these scores (for example of 0.49 for gender), suggesting some information about these attributes remains in the embeddings.

Table 5: Difference in scores metric for debiasing

	Gender	Rel.	Ethn.	Nat.
FB15K (filtered)				
Basic	2.79	4.41	6.64	2.85
Bose & Ham. [s]	2.75	3.15	3.62	1.53
Bose & Ham. [a]	6.85	9.76	11.13	5.46
KL + Attr.	0.19	0.60	1.26	0.47
FB3M (filtered)				
Basic	2.25	6.34	7.57	6.41
KL + Attr.	0.01	0.78	1.87	1.46
Wikidata (filtered)				
Basic	2.06	6.82	7.98	7.44
KL + Attr.	0.49	0.71	1.12	1.70

This conclusion is mirrored in Table 6, which shows the accuracy of a neural network trained to predict the sensitive attributes from the trained embeddings. For Bose and Hamilton (2019) we are still able to predict the correct sensitive attribute substantially more accurately than the Most Frequent baseline (we can predict a person's gender with 93.4% accuracy), indicating that despite the significant drop in model accuracy shown in 4, the model has not removed all bias. Our model, for FB15K and FB3M, reduces the accuracy of the neural network to very close to the most frequent class. For Wikidata, enough sensitive information remains to be able to predict some of the attributes very accurately. For example, we can predict a person's gender from their embedding with a 97.9% accuracy. These results suggest that higher values of w_{KL} would be suitable for Wikidata, if it is important that all sensitive information is removed.

A key component of our approach is that it operates through the model's usual score function, $g(.)$. This enables the fast training time relative to alternative methods, but raises a potential limitation; there is only an incentive via the KL loss to reduce sensitive information in the embeddings which can be detected by the (potentially very simple) function g. This is the motivation for introducing the neural network based method of measuring remaining sensitive information; to expose if information remains in the embedding which can be exposed by a more complicated (e.g. neural network based) function. Whilst the results in this paper suggest that this is not the case, care should be taken in practice to ensure that the function used for measuring the remaining sensitive information is as powerful as any downstream model for which the graph embeddings are an input.

Table 6: Accuracy of NN trained to predict sensitive attributes from embeddings

	Gender	Rel.	Ethn.	Nat.
FB15K (filtered)				
Most Frequent	0.767	0.292	0.142	0.594
Basic	0.990	0.552	0.534	0.799
Bose & Ham. [s]	0.933	0.406	0.391	0.732
Bose & Ham. [a]	0.934	0.423	0.472	0.736
KL + Attr.	0.767	0.291	0.166	0.594
FB3M (filtered)				
Most Frequent	0.778	0.256	0.227	0.380
Basic	0.990	0.756	0.686	0.844
KL + Attr.	0.788	0.306	0.431	0.435
Wikidata (filtered)				
Most Frequent	0.777	0.381	0.269	0.239
Basic	0.998	0.843	0.805	0.749
KL + Attr.	0.979	0.694	0.738	0.644

7 Summary

We have presented a novel method for debiasing knowledge graph embeddings, which is both significantly faster (allowing training on large knowledge graphs such as Wikidata in realistic timeframes) and less disruptive to accuracy than previous approaches. We demonstrated that the approach is also more effective in removing sensitive information from trained embeddings than previous methods, and through attribute vectors, give the user the flexibility to allow sensitive information to be used in predictions when desired.

References

Christine Basta, Marta Ruiz Costa-jussà, and Noe Casas. 2019. Evaluating the underlying gender bias in contextualized word embeddings. *CoRR*, abs/1904.08783.

Tolga Bolukbasi, Kai-Wei Chang, James Y. Zou, Venkatesh Saligrama, and Adam Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. *CoRR*, abs/1607.06520.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, pages 2787–2795, USA. Curran Associates Inc.

Antoine Bordes, Jason Weston, Ronan Collobert, and Yoshua Bengio. 2011. Learning structured embeddings of knowledge bases. In *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, AAAI'11, pages 301–306. AAAI Press.

Avishek Joey Bose and William Hamilton. 2019. Compositional fairness constraints for graph embeddings. *CoRR*, abs/1905.10674.

Aylin Caliskan, Joanna Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356:183–186.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *J. Mach. Learn. Res.*, 12:2121–2159.

Harrison Edwards and Amos Storkey. 2015. Censoring representations with an adversary. *arXiv preprint arXiv:1511.05897*.

Joseph Fisher, Arpit Mittal, Dave Palfrey, and Christos Christodoulopoulos. 2020. Measuring social bias in knowledge graph embeddings. In *Proceedings of the Knowledge-Graph Bias workshop*. AKBC.

Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. 2017. Word embeddings quantify 100 years of gender and ethnic stereotypes. *CoRR*, abs/1711.08412.

Hila Gonen and Yoav Goldberg. 2019. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. *CoRR*, abs/1903.03862.

Mark Graham, Bernie Hogan, Ralph K. Straumann, and Ahmed Medhat. 2014. Uneven geographies of user-generated information: Patterns of increasing informational poverty. *Annals of the Association of American Geographers*, 104(4):746–764.

Robert L. Logan IV, Nelson F. Liu, Matthew E. Peters, Matt Gardner, and Sameer Singh. 2019. Barack's wife hillary: Using knowledge-graphs for fact-aware language modeling. *CoRR*, abs/1906.07241.

Adam Lerer, Ledell Wu, Jiajun Shen, Timothee Lacroix, Luca Wehrstedt, Abhijit Bose, and Alex Peysakhovich. 2019. PyTorch-BigGraph: A Large-scale Graph Embedding System. In *Proceedings of the 2nd SysML Conference*, Palo Alto, CA, USA.

David Madras, Elliot Creager, Toniann Pitassi, and Richard S. Zemel. 2018. Learning adversarially fair and transferable representations. *ArXiv*, abs/1802.06309.

Thomas Manzini, Yao Chong Lim, Yulia Tsvetkov, and Alan W Black. 2019. Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. *arXiv preprint arXiv:1904.04047*.

Michaël Mathieu, Junbo Jake Zhao, Pablo Sprechmann, Aditya Ramesh, and Yann LeCun. 2016. Disentangling factors of variation in deep representation using adversarial training. In *NIPS*.

Chandler May, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. *CoRR*, abs/1903.10561.

Malvina Nissim, Rik van Noord, and Rob van der Goot. 2019. Fair is better than sensational: Man is to doctor as woman is to doctor. *CoRR*, abs/1905.09866.

Matthew E. Peters, Mark Neumann, Robert L. Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations.

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. Gender bias in coreference resolution. *arXiv preprint arXiv:1804.09301*.

Gabriel Stanovsky, Noah A. Smith, and Luke Zettlemoyer. 2019. Evaluating gender bias in machine translation. *CoRR*, abs/1906.00591.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. In *International Conference on Learning Representations*.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier, and Guillaume Bouchard. 2016. Complex embeddings for simple link prediction. *CoRR*, abs/1606.06357.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Claudia Wagner, David García, Mohsen Jadidi, and Markus Strohmaier. 2015. It's a man's wikipedia? assessing gender inequality in an online encyclopedia. *CoRR*, abs/1501.06307.

Claudia Wagner, Eduardo Graells-Garrido, and David García. 2016. Women through the glass-ceiling: Gender asymmetries in wikipedia. *CoRR*, abs/1601.04890.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: enhanced language representation with informative entities. *CoRR*, abs/1905.07129.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018a. Gender bias in coreference resolution: Evaluation and debiasing methods. *CoRR*, abs/1804.06876.

Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. 2018b. Learning gender-neutral word embeddings. *CoRR*, abs/1809.01496.

A Appendices

A.1 Discussion of performance of (Bose and Hamilton, 2019) with multiple filters

To remove bias from an embedding, (Bose and Hamilton, 2019) propose a set of neural network filters, $f_k(.)$, which take as input the baseline (potentially biased) embedding, and output an embedding of the same dimension with the sensitive information removed. For each sensitive attribute (gender, religion etc.), they have a separate filter network. The "compositional" approach proposed suggests they can use multiple filter networks to allow the final embedding to be "invariant w.r.t. some set of sensitive attributes, $S \subseteq \{1, ..., K\}$". To do this, they compose the final debiased embedding as the averaged output of S filtered embeddings, as shown in Equation (6) of the paper:

$$C - ENC(u, S) = \frac{1}{|S|} \sum f_k(ENC(u))$$

where u is the input embedding, S, ENC is the encoder model [15] and S is the set of filters for each attribute k. We find that as each filter network $f_k(.)$ is trained to only remove a single sensitive attribute (for example, gender), when the outputs of multiple filters are combined, the remaining outputs (for example from the filters for religion, nationality and ethnicity), leak gender information back into the final representation. This explains the notable difference in Tables 5 and 6 between the [s] version of their model, in which we only use one filter network (and which is the version used to provide the debiasing results in the authors code), and the [a] versions, in which we apply all four filter networks.

A.2 Labelling of whitelisted relations and professions

In Section 4.2, we introduced the concept of whitelisted relations, such as "speaks_languages", for which we allow some sensitive information (in this case nationality) to be used by the model. For Freebase and Wikidata we labelled such relations by hand, and provide some examples in Table 7. In total we whitelisted 60 relations for Freebase and 88 for Wikidata.

On top of this, for the relations "profession", we labelled each right hand side entity (i.e. each job type) with the sensitive attributes which could be used. For example, when predicting if someone is a nun, we allow the use of religious and gender information, whereas for predicting whether someone is a doctor, we allow the use of no sensitive information. We labelled the Freebase professions by hand, but for Wikidata there are around 12,000 professions, so we automated the process using keywords in the job description, wikidata subclasses, and properties.

For religion, a profession is whitelisted if any of the following three clauses apply:

1. Any of these keywords appear in professions definition:

   [religious, religion, divine]

2. The profession is a subclass (5 levels of inference) of any of these entities:

   [cleric, religious character, saint]

3. The profession has any of the following properties:

   [religion]

We use the same three clauses for gender, ethnicity and nationality, with different sets of keywords, subclasses and properties. We then filtered out false positives (there were less than five in total) manually.

Example professions are provided for each sensitive attribute in Table 8.

As mentioned in the paper, we do not suggest that the chosen relations/professions are complete/correct, and utilize them only as indicative of the types of relations/professions that may be used in practice, to demonstrate our debiasing approach.

Table 7: Example whitelisted relations

Freebase	
Gender	/music/opera_singer/voice_type
Religion	/religion/founding_figure/religion_founded
Ethnicity	/people/person/languages
Nationality	/people/person/place_of_birth

Wikidata	
Gender	P26 (spouse)
Religion	P119 (place of burial)
Ethnicity	P25 (mother)
Nationality	P102 (member of political party)

A.3 Removing sensitive triples from training data

An obvious initial approach to attempting to make all entities neutral with respect to sensitive attributes is to simply take the sensitive relations

[15] TransE in this paper, which does not update the input embedding unlike more complex models.

Table 8: Example whitelisted professions

Freebase	
Gender	/m/05cyczs (Crown Princess)
Religion	/m/0djbw (Rabbi)
Ethnicity	/m/0df9z (Holy Roman Emperor)
Nationality	/m/07068 (Samurai)

Wikidata	
Gender	Q16511993 (Queen)
Religion	Q208762 (Chaplain)
Ethnicity	—
Nationality	Q636207 (United States Attorney General)

(gender, religion etc.) out of the training data. In this section, we show that doing so is not sufficient, and that active debiasing is required (which we carry out via the KL loss in our framework).

We use the neural network based measure of debiasing introduced in Section 4.3, in which we optimize a feedforward neural network to predict sensitive attributes from human entities embeddings post-training. If sensitive information remains in the embeddings, it will be possible for the network to be more accurate in its predictions than simply predicting the most frequent class.

Table 9: Accuracy of NN trained to predict sensitive attributes from embeddings for FB15K

	Gender	**Rel.**	**Ethn.**	**Nat.**
Most Frequent	0.767	0.292	0.142	0.594
Basic	0.990	0.552	0.534	0.799
Removed attributes	0.853	0.402	0.420	0.690

Table 9 shows the accuracy of the networks predictions for the most frequent class, the "Basic" TransE trained embeddings with no debiasing, and the "Removed attributes" model, in which we simply remove all sensitive relations from the training data. Although removing the sensitive relations lowers the accuracy of predictions relative to the "Basic" model, for each sensitive attribute it is possible to do better than the "Most Frequent" prediction, indicating that sensitive information remains.

Although we do not conduct a thorough investigation into the exact cause of this, it is likely due to relations which are highly correlated with the sensitive attributes remaining. For example, the relation "speaks_languages" provides information on the nationality a person is likely to have. In a more worrying example, the model may learn to infer, for example, gender, from the profession which somebody has (given the one-sided distribution of some professions in the datasets). In order to avoid this, active debiasing is required.

A.4 Additional FB3M Results

As discussed in Section 4.3, we tuned the values of w_{KL} on the FB15K dataset only, as tuning is computationally expensive in that it requires multiple repetitions. This resulted in a choice of w_{KL} of 1.0 for gender, and of 100.0 for religion, ethnicity and nationality.

Table 10: Difference in scores metric for debiasing

FB3M (filtered)	**Gender**	**Rel.**	**Ethn.**	**Nat.**
Basic	2.25	6.34	7.57	6.41
KL + Attr. (o)	0.44	2.40	2.72	2.09
KL + Attr.	0.01	0.78	1.87	1.46

However, when we used these same weights for FB3M, the model retained an ability to predict the sensitive attributes, as shown in Table 10, which is analogous to Table 5 in the main paper. The results with the original weights (tuned on FB15K) are denoted with an (o). For each sensitive attribute, the model is able to predict the correct attribute of a human entity using the discriminator relations, although the difference in scores is brought down significantly relative to the "Basic" method.

Table 11: Accuracy of NN trained to predict sensitive attributes from embeddings

FB3M (filtered)	**Gender**	**Rel.**	**Ethn.**	**Nat.**
Most Frequent	0.778	0.256	0.227	0.380
Basic	0.990	0.756	0.686	0.844
KL + Attr. (o)	0.979	0.588	0.334	0.608
KL + Attr.	0.788	0.306	431	0.435

This result is supported by Table 11, which indicates that a neural network can be trained to predict the correct attributes from a person's embedding with the original weights, getting, for example, a 97.9% accuracy in the case of gender.

In light of this result, we increased the values of w_{KL} for FB3M, to 100.0 for gender, and 500.0 for religion, ethnicity and nationality. These are the results which we presented in the main paper, and they are repeated in each of the Tables in this section. With the higher weights, the differential in scores is brought down, reaching close to zero in the case of gender, and it becomes much harder to predict the sensitive attributes from the trained embeddings.

Table 12: Test set results for TransE embeddings

	Model	MRR	h@1	h@10	h@50
FB3M (filtered)					
Biased	Basic	0.684	0.612	0.794	0.843
Deb.	KL (o)	0.688	0.619	0.793	0.840
	KL	0.682	0.611	0.792	0.840
	KL + Attr. (o)	0.693	0.624	0.797	0.845
	KL + Attr.	0.684	0.611	0.798	0.846

The increased emphasis on the KL loss comes at a cost of some accuracy however, as shown in Table 12. With the original weight, the hits at 10 and MRR remain at baseline levels, and we get a slightly higher than baseline performance once the attribute vectors are added in. As we increase the values of w_{KL}, the discriminator only results fall below the "Basic" model, with some of this regained by the attribute vectors, mirroring the results in the main paper.