Compositional Demographic Word Embeddings

Charles Welch, Jonathan K. Kummerfeld, Verónica Pérez-Rosas, and **Rada Mihalcea**
Computer Science & Engineering
University of Michigan
{cfwelch,jkummerf,vrncapr,mihalcea}@umich.edu

Abstract

Word embeddings are usually derived from corpora containing text from many individuals, thus leading to general purpose representations rather than individually personalized representations. While personalized embeddings can be useful to improve language model performance and other language processing tasks, they can only be computed for people with a large amount of longitudinal data, which is not the case for new users. We propose a new form of personalized word embeddings that use demographic-specific word representations derived compositionally from full or partial demographic information for a user (i.e., gender, age, location, religion). We show that the resulting demographic-aware word representations outperform generic word representations on two tasks for English: language modeling and word associations. We further explore the trade-off between the number of available attributes and their relative effectiveness and discuss the ethical implications of using them.

1 Introduction

Word embeddings are used in many natural language processing tasks as a way of representing language. Embeddings can be efficiently trained on large corpora using methods like word2vec or GloVe (Mikolov et al., 2013; Pennington et al., 2014), which learn one vector per word. These embeddings capture syntactic and semantic properties of the language of all individuals who contributed to the corpus. However, they are unable to account for user-specific word preferences (e.g., using the same word in different ways across different contexts), particularly for individuals whose usage deviates from the majority. These individual preferences are reflected in the word's nearest neighbors. As an example, Table 1 shows the way two users use the word "health" and the word's five nearest neighbors in their respective personalized embedding spaces. The word is used in similar

User	Example Use	Nearest Neighbors
A	doctors think this is bad for her **health** ...	preventative, insurance, reform, medical, education
B	it is usually bad for your **health** ...	professional, mental, conduct, experiences, online
All	N/A	medical, preventative, insurance, safety, healthcare

Table 1: Nearest neighbors of the word "health" for two different users in personalized and a generic embedding space.

contexts, where contextual embeddings may give similar representations, but it has different salient meanings in the personal space of each user. User A tends to talk more about preventative care and insurance, while user B tends to talk about people's experiences affecting their mental health.

The typical approach in natural language processing (NLP) is to use one-size-fits-all language representations, which do not account for variation between people. This may not matter for people whose language style is well represented in the data, but could lead to worse support for others (Pavalanathan and Eisenstein, 2015; May et al., 2019; Kurita et al., 2019). While the way we produce language is not a direct consequence of our demographics or any other grouping, it is possible that by tailoring word embeddings to a group we can more effectively model and support the way they use language.

Additionally, personalized embeddings can be useful for applications such as predictive typing systems that auto-complete sentences by providing suggestions to users, or dialog systems that follow the style of certain individuals or professionals (e.g., counselors, advisors). They can also be used to match the communication style of a user, which would signal cooperation from a dialog agent.

In this paper, we propose compositional demographic word embeddings as a way of building personalized word embeddings by leveraging data from users sharing the same demographic attributes

(e.g., age: *young*, location: *Europe*). Our proposed method has the benefits of personalized word representations, while at the same time being applicable to users with limited or no data.

To implement and evaluate our proposed method, we build a large corpus of Reddit posts from 61,981 users for whom we extract self-reported values of up to four demographic properties: age, location, gender, and religion. We examine differences in word usage and association captured by the demographics we extracted and discuss the limitations and ethical considerations of using or drawing conclusions from this method. We explore the value of compositional demographic word embeddings on two English NLP tasks: language modeling and word associations. In both cases, we show that our proposed embeddings improve performance over generic word representations.

2 Related Work

Embedding Bias. Recent work on embeddings has revealed and attempted to remove racial, gender, religious, and other biases (Manzini et al., 2019; Bolukbasi et al., 2016). The bias in our corpora and embeddings have a societal impact and risks exclusion and demographic misrepresentation (Hovy and Spruit, 2016). This means that users of certain regions, ages, or genders may find NLP technologies more difficult to use. For instance, when using standard corpora for POS tagging, Hovy and Søgaard (2015) found that models perform significantly lower on younger people and ethnic minorities. Similarly, results on text-based geotagging show best results for men over 40 (Pavalanathan and Eisenstein, 2015).

Similar results are starting to be found in embeddings produced by contextual embedding methods (May et al., 2019; Kurita et al., 2019). We focus on non-contextual embedding methods because of their computational efficiency, which is crucial if many separate representations are being learned. Additionally, there may not be a large amount of available data for underrepresented groups and these contextualized models require billions of tokens for training. Recent work has also shown that static embeddings are competitive with contextualized ones in some settings (Arora et al., 2020).

Personalization. The closest work is Garimella et al. (2017)'s exploration of demographic-specific word embedding spaces. They trained word embeddings for male and female speakers who live in the USA and India using skip-gram architectures that learn a separate word matrix for each demographic group (e.g., male speakers from the USA).

Another line of work used discrete (Hovy, 2015) or continuous values (Lynn et al., 2017) to learn speaker embeddings: a single vector for each user. The speaker embedding is appended to the input of the recurrent or output layer, and trained simultaneously with the rest of the model. This idea applies to any contextual information type and was introduced as a way to condition language models on topics learned by topic modeling (Mikolov and Zweig, 2012). It has since been used as a way of representing users in tasks such as task-oriented and open-domain dialog (Wen et al., 2013; Li et al., 2016), information retrieval based on book preferences (Amer et al., 2016), query auto-completion (Jaech and Ostendorf, 2018), authorship attribution (Ebrahimi and Dou, 2016), sarcasm detection (Kolchinski and Potts, 2018), sentiment analysis (Zeng et al., 2017), and cold-start language modeling Huang et al. (2016). Finally, a recent study by King and Cook (2020) compared how to improve a language model with user-specific data using priming and interpolation, depending on the amount of data available, learning a new model for each user.

More generally, personalization has been extensively applied to marketing, webpage layout, product and news recommendation, query completion, and dialog (Eirinaki and Vazirgiannis, 2003; Das et al., 2007). Welch et al. (2019a,b) explored predicting response time, common messages, and speaker relationships from personal conversation data. Zhang et al. (2018) conditioned dialog systems on artificially constructed personas and Madotto et al. (2019) used meta-learning to improve this process. Goal-oriented dialog has used demographics (i.e. age, gender) to condition system response generation, showing that this relatively coarse grained personalization improves system performance (Joshi et al., 2017).

Social Media. We use social media data with demographic attributes inferred from user posts. Prior work has explored extraction or prediction of attributes such as age, gender, region, and political orientation (Rao et al., 2010; Rangel et al., 2013). Work on analyzing the demographics of social media users also includes race/ethnicity, income level, urbanity, emotional stability, personality traits (McCrae and Costa Jr, 1997), and life satisfaction (Duggan and Brenner, 2013; Correa et al., 2010).

One particularly relevant study by Gjurković and Šnajder (2018) presented a corpus of Reddit users with personality information as well as some demographics for a subset of users. Unlike our approach, which is based on text content, they extract information from Reddit flairs, a type of user tag. Out of their set of 10,295 users, 2,253 are also in our set of users (22% of theirs, 0.5% of ours) that have one or more demographic labels, confirming the speculation in their paper that extracting demographics from text is a complementary approach that captures more information about users in their data. Other work has used Reddit posts to identify users who were diagnosed with depression (Yates et al., 2017) and to construct personas for personalized dialog agents (Mazaré et al., 2018).

Language Models. To evaluate embeddings, we consider language modeling, a task that has long been used for speech recognition and translation, and more recently been widely used for model pre-training. A range of models have been developed, with progressively larger models trained on more data (e.g., Dai et al., 2019). Variations of the LSTM have consistently achieved state-of-the-art performance without massive compute resources (Merity et al., 2018a; Melis et al., 2019; Merity, 2019; Li et al., 2020). We use the AWD-LSTM (Merity et al., 2018b) in our experiments, as it achieves very strong performance, has a well-studied codebase, and can be trained on a single GPU in a day.

3 Dataset

Our first contribution is a new dataset. We use English Reddit comments as they are publicly available, are written by many users, and span multiple years.[1] We extract demographic properties of users from self-identification in their text.

3.1 Finding Demographic Information

Reddit users do not have profiles with personal information fields that we could scrape. Instead, we developed methods to extract demographic information from the content of user posts.

In order to determine what kind of information we can extract about users, we performed a preliminary analysis. We manually labeled a random sample of 132 statements that users made about

themselves. We specifically searched for statements starting with phrases such as 'i am a' or 'i am an'. In our sample: 36% clearly stated the user's age, religion, gender, occupation, or location; 34% contained descriptive phrases that were difficult to categorize like 'i am a big guy' or 'i am a lazy person'; and 30% mentioned attributes such as sexual orientation, dietary restrictions, political affiliations, or hobbies that were rare overall.

Based on our analysis, we decided to focus on age, religion, gender, occupation, and location as the main attributes.[2] These were extracted as follows:

Age. We extracted the user's age using a regular expression.[3] During this process, we found users that were matched to different ages due to the corpus covering user activity across several years. In those cases, we removed users whose age difference was greater than the time span of our corpus. Additionally, we excluded users who said they were less than 13 years of age, as this violates the Reddit terms of service. We decided to split the age into two groups, young and old at a threshold of 30, as this split was used in previous work (Rao et al., 2010), and it gave a reasonable split for our data and the data we used for testing word associations (Garimella et al., 2017).

Gender. Gender was extracted by searching for statements referring to oneself as a 'boy', 'man', 'male', 'guy', for male, or 'girl', 'woman', 'female', 'gal', for female. Manual inspection revealed some users indicated that they were of both genders. In that case, if one gender occurred less than one fifth of the time we took the majority of the reported gender, otherwise we removed the user from our dataset. We acknowledge that this approach excludes transgender, gender fluid, and a range of non-binary people, and may misgender people as well (see § 7 for further discussion of these issues).

Location. To obtain location information, we searched for phrases such as 'i am from' and 'i live in.' Next, whenever either the next token is (1) tagged as a location by a named entity recognizer (Manning et al., 2014), (2) a noun, or (3) the word 'the', we select all subsequent tokens in the phrase as the user location. Manual inspection of matches showed that Reddit users are not consistent in the

---

[1] https://www.reddit.com/r/datasets/comments/3bxlg7/i_have_every_publicly_available_reddit_comment/

[2] We attempted to extract occupations, but found they were difficult to identify and group because there are many different occupations, many ways of stating one's occupation, and many ways to describe the same occupation.

[3] .*?(i am|i\'m) (\\d+) (years|yrs|yr) old[^e].*?

granularity of reported location. Statements included cities, state, province, country, continent, or geographical region. Based on the number of users per country, we decided to merge some countries into region labels while leaving others separate. This resulted in the following set of regions: USA, Asia, Oceania, UK, Europe, Canada. We further matched location statements to lexicons to resolve the location to one of these regions, removing common relative location words.[4] For larger population regions of Canada and the USA, we match statements using state abbreviations, province names, highest population cities, and in the USA we also match the capital cities. For other regions we only match the highest population cities as there were too many cases to cover.

Religion. To extract religion, we searched for the five largest global religious populations,[5] counting 'secular', 'atheist', and 'agnostic' as one non-religious group. We used a regular expression[6] and filtered users who stated beliefs in more than one of these five groups.

3.2 Post-processing

The resulting dataset was further filtered to remove known bots.[7] For the demographic data we consider two subsets. First, the set of users for which all four attributes are known (4Dem). With this set we perform ablation experiments on the number of known attributes in a controlled manner. However, it is important to note that this set may not be representative of most users on Reddit, as it focuses on users willing to divulge a range of demographic attributes. Our second sample addresses this by including users for whom we identify two or more of the demographic attributes (2+Dem). Statistics for these sets are described in Table 2, along with the training, development, and test splits used for the language modeling experiments.

The distribution of demographic values for each of these sets is shown in Figure 1. Looking at the set of all users in our data who have at least two known demographic attributes (2+Dem), we find

---

[4]northern, western, eastern, southern, downtown, suburbs

[5]From https://www.adherents.com/, although note that since our study the domain name has been hijacked by a payday loans service. The site is archived by the Library of Congress at https://www.loc.gov/item/lcwaN0003960/

[6].*?(i am|i\'m) (a)?(christian | muslim | secular | atheist | agnostic | hindu | buddhist).*?

[7]https://www.reddit.com/r/autowikibot/wiki/redditbots

Set	Users	Posts
All Reddit	13,213,172	1,430,935,783
2+Dem		
Total	61,627	205,394,970
Training	34,110	50,000
Validation	9,190	10,000
Test	9,143	10,000
4Dem		
Total	354	3,433,062
Test	354	10,000

Table 2: Statistics for two Reddit sets: with at least two demographic attributes (2+Dem), or all four demographic attributes (4Dem). Training, development, test splits used in the language modeling experiments are also shown. First row shows overall number of posts and users from the entire set of Reddit posts.
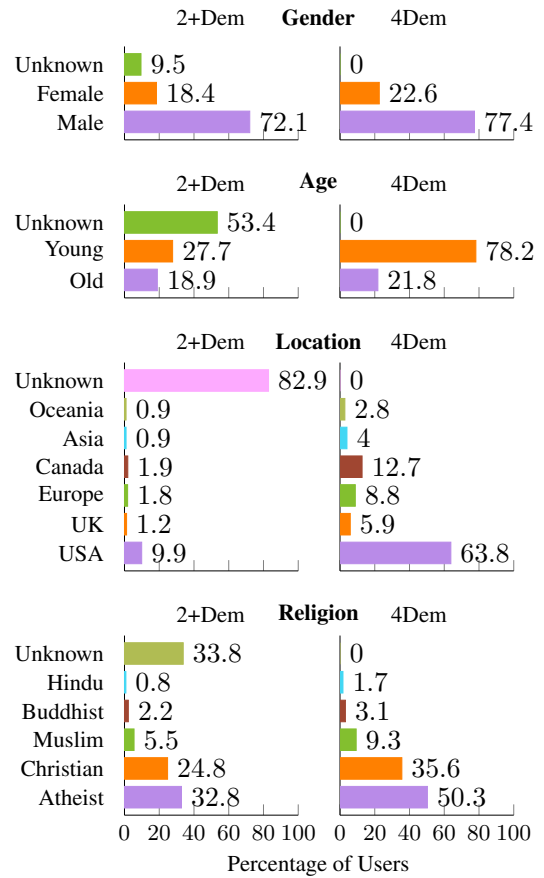

Figure 1: Distribution of the four demographic attributes in our two datasets, showing the set with all demographics known on the right and the random sample from those with at least two known on the left.

that 83% of the time location is unknown. Age and religion are the next most frequently missing at 53% and 34% respectively. Gender is more likely to be known than the other attributes: only 10% of users in this subset have an unknown gender. In a manual evaluation of all our extracted attribute labels for the 100 users, we found accuracies of

Gender		Age		Religion		Location	
Male	Female	Young	Old	Christian	Atheist	USA	Canada
blush		health		embodying		america	
blushing	brow	regen	care	exalting	unionism	europe	original
smile	eyeshadow	mana	reform	creaturely	mercantilist	country	tv
chortle	bronzer	aid	healthcare	extols	american	canada	worst
swoon	nars	permanent	education	mysteriousness	corporatocracy	sweden	hot
snicker	nyx	condition	coverage	idolization	unfree	mexico	space
wince	lipstick	treatment	high-deductible	magnanimity	proletarian	china	actual
chuckle	mascara	mental	socialized	asceticism	environmentalist	india	body
blushes	primer	preventative	medical	imbuing	wage-slavery	africa	home
smirk	concealer	benefits	insurance	unalterable	communistic	usa	move
guffaw	highlighter	medical	condition	mortification	free-marketeers	britain	nation

Table 3: Examples of words with low overlap in nearest neighbors, showing how meaning can differ across the values of a demographic attribute.

94% for location and gender, 78% for religion, and 96% for age. Additional details of this evaluation are provided in the supplementary material.

4 Generating Compositional Demographic Word Embeddings

We propose two methods for learning compositional demographic embeddings. The first learns a generic embedding for each word and a vector representation of each demographic attribute (including 'unknown'). This is memory efficient, as we need only 19 vectors to cover all of our attributes. In the second method, for each word we learn (a) a generic embedding and (b) a vector for each demographic attribute. This is more expressive, but requires twenty vectors for each word.

4.1 Demographic Attribute Vectors

In this approach we jointly learn a matrix for words and a separate vector for each demographic value. The word matrix $W \in \mathbb{R}^{|V| \times k}$ has a row for each word in the vocabulary and a k-dimensional vector for each embedding. The demographic values can be represented by another matrix $D \in \mathbb{R}^{|C| \times k}$, where C is the set of all demographic values (e.g., male, female, christian, USA). The hidden layer is calculated as $h = W_h^\intercal (W_w + C_g + C_l + C_r + C_a)$ where w represents the one-hot encoding of an input word and g, l, r, a represent the demographic values of the speaker. This is a modified skip-gram architecture (Mikolov et al., 2013) with a hierarchical softmax, which sums five terms so that back-propagation updates the word representation as well as the demographic values.

We use posts from all users to train embeddings for words that occur at least five times across all users. This yields a vocabulary of 503k words. We learn 100-dimensional embeddings with an initial learning rate of 0.025 and a window size of five.

4.2 Demographic Word Matrices

When learning demographic matrices we separately run our skip-gram model for each of the demographic attributes (e.g., gender) and learn a generic word matrix $W_G \in \mathbb{R}^{|V| \times k}$ and a value specific word matrix for each value, v, of the given attribute, A, (e.g., male, female) $W_v \in \mathbb{R}^{|V| \times k}, \forall v \in A_v$. This changes the hidden layer calculation to $h = W_h^\intercal G_w + W_h^\intercal W_{vw}$, with hidden layer weights W_h, and the model then learns a generic word representation, in matrix G, while learning the value specific impact on the meaning of that word.

Differences Across Demographic Embeddings. In order to understand what our embeddings capture, we examine words that have different representations across demographics. We can look at the nearest neighbors of a given query word across the embedding spaces for different demographics. We perform this analysis on both the demographic matrices and vectors, finding less variation in the neighbors when using demographic vectors, making them less interesting. We show examples of words with low overlap in nearest neighbors for demographic matrices in Table 3. These show the differences in word meaning across groups.

5 Language Modeling

We first examine the usefulness of our embeddings by showing that they can help us better model a user's language. We consider two experiments. First, we focus on compositional demographic embeddings and sample 50k posts from our corpus for training the language model and 5k for each of validation and test. Next, we compare with a user-specific model on a sample of our data with text

from just 100 users who each have a large amount of data available in our corpus, with an average of 3.2 million tokens per user.

In both experiments, we use the language model developed by Merity et al. (2018b,a). As discussed in § 2, this model was recently state-of-the-art and has been the basis of many variations. We modify it to initialize the word embeddings with the ones we provide and to concatenate multiple embedding vectors as input to the recurrent layers. The rest of the architecture is unaltered. We tried adding rather than concatenating and found no improvement. We chose to concatenate the inputs with the intuition that the network would learn how to combine the information itself.

We explored various hyperparameter configurations on our validation set and found the best results using dropout with the same mask for generic and demographic-specific embeddings, untied weights, and fixed input embeddings. Untying and fixing input embeddings is supported by concurrent work (Welch et al., 2020b). Each model is trained for 50 epochs. We use the version from the epoch that had the best validation set perplexity, a standard metric in language modeling that measures the accuracy of the predicted probability distribution.

5.1 Demographic Perplexity Evaluation

Table 4 shows results for our demographic personalization methods, which are designed to handle new users for whom we have demographics but not much text data. The first method, demographic vectors, performs no better than generic embeddings. This is surprising since prior work has achieved success on a range of tasks with this kind of representation (see § 2). We suspect that for language modeling the variations are too fine-grained to be captured by a single vector. However, demographic matrices do improve significantly over generic embeddings. A model with all demographics improves the most, but we also see improvements when only one demographic value is known.

The LSTM hidden layer size is the same across models, but the change in the input size affects the total number of parameters. To control for this, we ran our baseline model and model initialized with generic words with a larger input size, matching the number of parameters in our best models. As shown in Table 4, this increase in parameters does not improve performance.

Model Type and Input Size	2+Dem	4Dem
Baseline, 100	123.8	124.6
Baseline, 500	125.1	126.1
Generic Words Only, 100	116.0	112.1
Generic Words Only, 500	115.8	112.6
Demographic Vectors, 200	116.7	113.0
Demographic Matrices		
+ Age Only, 200	109.4	110.3
+ Gender Only, 200	109.4	109.9
+ Location Only, 200	109.7	112.9
+ Religion Only, 200	110.9	112.0
+ All Demographics, 500	**107.7**	**109.1**

Table 4: Perplexity on the demographic data. Our demographic-based approach improves performance. The difference between the last row and generic words is significant ($p < 0.00001$ with a permutation test).

Att.	Value	2+Dem		4Dem	
		0D	4D	0D	4D
Age	Young	107.1	103.6	110.6	108.0
	Old	115.1	111.1	114.0	112.0
	Unknown	112.3	108.6	-	-
Location	USA	108.5	105.7	108.1	105.1
	Canada	135.7	132.6	110.0	107.6
	Oceania	111.0	108.7	114.8	112.8
	Europe	130.0	128.2	133.0	130.1
	Asia	109.3	108.6	145.3	145.4
	UK	115.3	113.5	96.9	96.9
	Unknown	111.0	107.1	-	-
Religion	Christian	116.5	111.9	108.5	105.9
	Atheist	106.4	103.2	112.7	109.9
	Muslim	112.7	108.5	109.5	108.4
	Hindu	122.4	115.6	158.1	159.5
	Buddhist	114.1	111.7	116.4	114.1
	Unknown	122.3	109.1	-	-
Gender	Male	113.5	109.2	115.2	112.6
	Female	100.9	97.8	102.7	100.4
	Unknown	122.3	118.5	-	-

Table 5: Perplexity for language models with no demographics (0D) or with all four demographic matrices (4D) with results broken down by demographic values.

5.1.1 Ablation Experiments

Table 4 shows results when using no demographics (top 4 rows), one demographic at a time (rows 6-9) and all four demographics (row 10). Each attribute improves perplexity, with age and gender improving it more than location and religion.

Additionally, we perform a breakdown of the performance of our demographic matrices language model on each of the demographic groups. These results are shown in Table 5. We do see worse performance on some minorities as compared to other groups for the same model, although that is not always the case (gender, for instance, shows better perplexities for female than for male, and Muslim shows lower perplexity than Christianity, which has substantially more data). When we use the de-

mographic word embeddings in our model, we are able to improve performance for all demographic groups, including minorities.

We also find that the performance on the 'unknown' group increases in all cases with our largest improvement on 'unknown' religion. The unknown is explicitly modeling people in our dataset who have either (1) stated this demographic information with a value that we model but not in a way that our regular expressions identify, (2) stated this demographic information with a value that we do not model, or (3) have not stated this demographic information. In the second case, the effect is that it is useful to know which demographic groups the speaker does not belong to. In the third case, it may be that not sharing this particular piece of information (while sharing other personal information) says more about what the speaker will tend to say.

5.2 Comparison with User Representations

For users with a lot of data, it is possible to train a user-specific model, with embeddings that capture their unique language use. We would expect this to be better than our demographic embeddings, but also only be feasible for users with a lot of data. This experiment compares our demographic approach with a user-specific approach.

We create a model for each user using the sample that has a large amount of data for 100 users (3.2 million tokens each on average) as done in concurrent work (Welch et al., 2020a). We tried two approaches, user vectors and user matrices, which are analogous to our demographic vectors and matrices. The difference is that rather than having a separate vector / matrix for each demographic we have a separate vector / matrix for each user. Our split sizes for language model experiments are the same as the demographic experiments.

Results. Table 6 shows results for generic embeddings, user vectors, user matrices, and demographic matrices. We find that user vectors, as have been used widely in previous work (Kolchinski and Potts, 2018; Li et al., 2016), do not improve performance. Both our demographic and user matrices improve performance over generic embeddings with comparable performance. While we chose 100 users with a lot of data, they had less data than the amount used to train each demographic specific model. The relationship between the amount of data, its similarity to a user's writing, and the effect on performance is an interesting open question.

Model	PPL
Generic Word Embeddings	63.94
User Vectors	68.98
User Matrices	61.69
Demographic Matrices	61.80

Table 6: Comparing our demographic-based approach with two user-specific approaches. Perplexities are generally lower than previous tables because the threshold for rare words being made UNK was higher.

6 Demographic Word Associations

As a second evaluation, we consider word associations, a core task in NLP that probes the relatedness or similarity between words. Data is collected for the task by presenting a stimulus word (e.g., *cat*) and asking people what other words come to mind (e.g., *dog* or *mouse*). Earlier systems relied on resources such as WordNet to solve the task, but most recent work has used word embeddings.

Data. For our evaluation, we use data from Garimella et al. (2017). They constructed a word association dataset and experimented with learning separate word embedding matrices for different demographic groups. To collect the data, they (1) asked crowd workers to write one word associated with a single word prompt and (2) asked the workers their gender, age, location, occupation, ethnicity, education, and income. Only gender and location information was released, but the authors provided age information upon request.

Evaluation. As in prior work, we consider evaluation metrics defined in terms of: f_w, the number of people who listed word w for a stimulus; f_{max}, the highest f_w across all words chosen for a stimulus; and t, the number of participants given a stimulus. *best* is f_w divided by f_{max}, where w is the word in the embedding space closest to the stimulus word; *ooN* (out-of-N) is $\sum f_w/t$ for the N words in the embedding space closest to the stimulus word; both are averaged over all stimulus words.

We consider two experiments. One directly matches Garimella et al. (2017), testing each demographic group separately. Since our interest is in compositionality, we also introduce a setting where the data is split into eight disjoint sets, one for each combination of the three attributes.

Models. Garimella et al. (2017) proposed two methods, which we merge by taking the best result from either one. We considered only our demographic matrix embeddings as they performed best

Method	best IN	best US	oo3 IN	oo3 US	oo10 IN	oo10 US	best M	best F	oo3 M	oo3 F	oo10 M	oo10 F	best Y	best O	oo3 Y	oo3 O	oo10 Y	oo10 O
C-SGM	0.09	0.03	0.14	0.07	0.19	0.10	0.13	0.16	0.20	0.20	0.25	0.26	-	-	-	-	-	-
Ours G	**0.18**	**0.21**	**0.18**	**0.40**	0.31	0.63	0.17	0.17	**0.22**	0.26	0.35	0.42	0.18	0.18	**0.19**	0.26	0.31	0.42
Ours G+D	0.17	0.19	0.16	0.39	**0.32**	**0.64**	**0.18**	0.20	**0.22**	**0.27**	0.37	**0.45**	**0.19**	0.21	**0.19**	0.29	0.32	0.44

Table 7: Comparison of demographic-aware word association similarities for our embeddings using (G)eneric or (G)eneric+(D)emographic, and the best results of the two variants of the composite skip-gram model (C-SGM) from Garimella et al. (2017). We show improved results for (US), (IN)dia, (M)ale, and (F)emale, and provide new results using age for (Y)ounger than 30 and (O)lder.

Metric	Embeddings	Gender: M, Location: IN, Age: Y	Gender: M, Location: IN, Age: O	Gender: M, Location: US, Age: Y	Gender: M, Location: US, Age: O	Gender: F, Location: IN, Age: Y	Gender: F, Location: IN, Age: O	Gender: F, Location: US, Age: Y	Gender: F, Location: US, Age: O	Macro
best	Generic	**0.178**	0.164	0.209	0.223	0.171	0.175	0.198	0.213	0.191
	Age	0.175	**0.169**	**0.211**	**0.239**	0.167	0.180	**0.207**	0.225	0.197
	Age + Gender	0.175	**0.169**	**0.211**	**0.239**	**0.174**	**0.181**	**0.207**	**0.227**	**0.198**†
	Age + Gender + Location	0.163	0.161	0.187	0.198	0.158	0.176	0.180	0.187	0.176
oo3	Generic	0.116	0.105	0.205	0.271	0.118	0.126	0.216	0.365	0.190
	Age	0.119	0.111	0.207	**0.284**	0.121	0.137	0.221	0.378	0.197
	Age + Gender	0.120	0.111	0.207	**0.284**	0.123	0.136	**0.235**	0.378	0.199
	Age + Gender + Location	**0.131**	**0.117**	**0.214**	0.265	**0.125**	**0.145**	0.234	**0.383**	**0.203**†
oo10	Generic	0.217	0.194	0.346	0.440	0.209	0.231	0.364	0.588	0.324
	Age	**0.230**	0.205	**0.362**	**0.456**	**0.227**	0.246	**0.395**	**0.615**	**0.342**
	Age + Gender	**0.230**	0.205	**0.362**	**0.456**	**0.227**	0.250	0.389	0.614	**0.342**
	Age + Gender + Location	0.227	**0.214**	0.339	0.432	0.224	0.249	0.373	0.581	0.329

Table 8: Results on the 8 disjoint word association subsets for each combination of attributes. Similarities concatenate three embeddings that are each either generic, or specific to that demographic attribute. Overall, using age and gender in combination gives the best performance, though using all three is better on oo3. † indicates statistically significant improvement (permutation test, $p < 0.001$) over the next best model on the marked metric.

on language modeling. For the experiment with separate demographics, we use the appropriate embeddings. For the experiment with combinations of demographics, we concatenate the embeddings. We also compare to concatenation of generic embeddings learned for each attribute (this performs better than any individual generic embedding).

Results. Table 7 shows results on the single-demographic experiment. We achieve higher performance, but that may come from the change in training dataset.[8,9] Table 8 shows results on the multi-demographic setting. We include only the best pair (age and gender) due to space. We have seen in earlier experiments that location does not perform as well as the other attributes and found the same trend here. Overall, composing demographic-based representations helps, with a combination of all three attributes consistently performing well on the *oo3* metric, while having two helps on the *best* metric. Generic embeddings only score the highest

on one subset: Male, India, Young.

7 Limitations and Ethical Considerations

This work uses demographic information to modify language representation. This type of work is encouraged by the numerous arguments outlined in (Perez, 2019), which demonstrate the need for demographic data disaggregation in order to make decisions and build technologies that are equitable for all. We view our work as an initial investigation of differences in language model performance across demographics and how technology can be improved for the identified groups. Our results in Tables 4 and 5 show that using demographic information can enable the development of language tools that improve performance for all groups compared to simply training on all data.

Although we show that some language production aspects are correlated with demographic information, we do not believe the way we speak is a direct and only consequence of one's demographics, neither do we claim that this is the ideal information source for it or that this will necessarily hold for populations sampled significantly differently than in our study. As a consequence, it

---

[8]Their models are trained on 67.6m tokens of blog data, while ours are trained on 1,400m tokens of Reddit data.

[9]We see a larger gain for the US than the IN evaluation. This may be because in our data location is unknown for many users and India is underrepresented (so much so that we aggregate it into all of Asia).

is possible that using demographics in embedding construction could accentuate bias, although this remains to be studied. Those that use our method should account for this possibility.

Our study uses four demographic variables and only covers a subset of the potential values of each demographic. For instance, we do not use the same granularity across locations, include all locations, religions, or gender identities. We simplify age into ranges (actual user age estimates are available in our released data). The groups 'secular', 'agnostic', and 'atheist' are grouped into one broader group. Our sample is further biased by the choice of platform as each platform contains text from different populations. Users in our sample are predominately young, male, atheist, and live in the United States.

When using gender as a study variable, we followed the recommendations of Larson (2017). Our "gender" extraction method does not refer to biological sex. After running gender extraction patterns, users are assigned to either the 'male', 'female', or 'unknown' label, meaning that on the basis of these phrases one's gender identity is assumed to be binary or to be a gender identity unknown to our model, which may include those who are transgender, non-binary, or those who do not wish to disclose their gender. However, we are aware that the use of regular expressions for the extraction of demographic attributes can lead to false positives and false negatives (error rates are provided in the supplemental material) and that there exists a bias in using these strategies, as populations that do not wish to be identified are less likely to explicitly make such statements. For transparency, our released code includes the scripts used to assign demographic labels.

Above we discussed concerns for incorrect demographic assignment when developing models. There are also potential negative consequences when using these models in a deployed system. Our embeddings can only be used when the demographics of a user are known. This may be acceptable if the user voluntarily self-reports their demographics with the understanding that they will alter the predictions they receive. However, if demographics are automatically inferred there is a risk of misattribution, which depending on the application may have negative consequences.

A separate consideration is the environmental impact of this approach. Compared to the standard method, our approach does involve training more models, but the cost of inference is likely only marginally higher. We believe the additional cost in training is worth the benefits to individual users.

Finally, we acknowledge that components of our method could potentially be used for user profiling (Rangel et al., 2013) and/or surveillance of target populations, thus exposing members of underrepresented groups to harms such as discrimination and coercion and threatening intellectual freedom (Richards, 2013). Similarly, the language models could be used to generate text in the style of a target population or at least to estimate the label distribution of a given text, which would help obfuscate the identity of the author (Potthast et al., 2018). This obfuscation could help hide an author's identity in order to avoid surveillance or could be used maliciously to infiltrate communities online. We advocate against the use of our methods for these or other ethically questionable applications.

8 Conclusions

We proposed a novel method of generating word representations by composing demographic-specific word vectors. Through experiments on two core language processing tasks, language modeling and word associations, we show that demographic-aware word representations outperform generic embeddings. We also find that demographic matrices perform much better than demographic vectors. Through several ablation analyses we show that word embeddings that leverage multiple demographic attributes give better performance than those using single attributes.

To support future work that can help model individuals and demographics, our code is available at http://lit.eecs.umich.edu and our dataset of Reddit users with demographic labels is available upon request. We hope this will support work on solutions for NLP applications and resources that can better serve minorities and underrepresented groups.

References

Nawal Ould Amer, Philippe Mulhem, and Mathias Géry. 2016. Toward word embedding for personalized information retrieval. In *Neu-IR: The SIGIR 2016 Workshop on Neural Information Retrieval*.

Simran Arora, Avner May, Jian Zhang, and Christopher Ré. 2020. Contextual embeddings: When are they worth it? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics.

David Bamman, Chris Dyer, and Noah A. Smith. 2014. Distributed representations of geographically situated language. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 828–834, Baltimore, Maryland. Association for Computational Linguistics.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. 2016. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Advances in Neural Information Processing Systems*, pages 4349–4357.

Teresa Correa, Amber Willard Hinsley, and Homero Gil De Zuniga. 2010. Who interacts on the web?: The intersection of users' personality and social media use. *Computers in human behavior*, 26(2):247–253.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy. Association for Computational Linguistics.

Abhinandan S Das, Mayur Datar, Ashutosh Garg, and Shyam Rajaram. 2007. Google news personalization: scalable online collaborative filtering. In *Proceedings of the 16th international conference on World Wide Web*, pages 271–280. ACM.

Maeve Duggan and Joanna Brenner. 2013. *The Demographics of Social Media Users, 2012*, volume 14. Pew Research Center's Internet & American Life Project Washington, DC.

Javid Ebrahimi and Dejing Dou. 2016. Personalized semantic word vectors. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 1925–1928. ACM.

Magdalini Eirinaki and Michalis Vazirgiannis. 2003. Web mining for web personalization. *ACM Transactions on Internet Technology (TOIT)*, 3(1):1–27.

Aparna Garimella, Carmen Banea, and Rada Mihalcea. 2017. Demographic-aware word associations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2285–2295, Copenhagen, Denmark. Association for Computational Linguistics.

Matej Gjurković and Jan Šnajder. 2018. Reddit: A gold mine for personality prediction. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*, pages 87–97, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Dirk Hovy. 2015. Demographic factors improve classification performance. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 752–762, Beijing, China. Association for Computational Linguistics.

Dirk Hovy and Anders Søgaard. 2015. Tagging performance correlates with author age. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 483–488, Beijing, China. Association for Computational Linguistics.

Dirk Hovy and Shannon L. Spruit. 2016. The social impact of natural language processing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 591–598, Berlin, Germany. Association for Computational Linguistics.

Yu-Yang Huang, Rui Yan, Tsung-Ting Kuo, and Shou-De Lin. 2016. Enriching cold start personalized language model using social network information. In *International Journal of Computational Linguistics & Chinese Language Processing, Volume 21, Number 1, June 2016*.

Aaron Jaech and Mari Ostendorf. 2018. Personalized language model for query auto-completion. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 700–705, Melbourne, Australia. Association for Computational Linguistics.

Chaitanya K Joshi, Fei Mi, and Boi Faltings. 2017. Personalization in goal-oriented dialog. *arXiv preprint arXiv:1706.07503*.

Milton King and Paul Cook. 2020. Evaluating Approaches to Personalizing Language Models. In *Proceedings of the Twelfth International Conference on Language Resources and Evaluation (LREC 2020)*, Marseille, France. European Language Resources Association (ELRA).

Y. Alex Kolchinski and Christopher Potts. 2018. Representing social media users for sarcasm detection. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1115–1121, Brussels, Belgium. Association for Computational Linguistics.

Keita Kurita, Nidhi Vyas, Ayush Pareek, Alan W Black, and Yulia Tsvetkov. 2019. Quantifying social biases in contextual word representations. In *Proceedings of GeBNLP 2019: 1st ACL Workshop on Gender Bias for Natural Language Processing*.

Brian Larson. 2017. Gender as a variable in natural-language processing: Ethical considerations. In *Proceedings of the First ACL Workshop on Ethics in Natural Language Processing*, pages 1–11, Valencia, Spain. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 994–1003, Berlin, Germany. Association for Computational Linguistics.

Yinqiao Li, Chi Hu, Yuhao Zhang, Nuo Xu, Yufan Jiang, Tong Xiao, Jingbo Zhu, Tongran Liu, and Changliang Li. 2020. Learning architectures from an extended search space for language modeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6629–6639, Online. Association for Computational Linguistics.

Veronica Lynn, Youngseo Son, Vivek Kulkarni, Niranjan Balasubramanian, and H. Andrew Schwartz. 2017. Human centered NLP with user-factor adaptation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1146–1155, Copenhagen, Denmark. Association for Computational Linguistics.

Andrea Madotto, Zhaojiang Lin, Chien-Sheng Wu, and Pascale Fung. 2019. Personalizing dialogue agents via meta-learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5459, Florence, Italy. Association for Computational Linguistics.

Christopher D. Manning, Mihai Surdeanu, John Bauer, Jenny Finkel, Steven J. Bethard, and David McClosky. 2014. The Stanford CoreNLP natural language processing toolkit. In *Association for Computational Linguistics (ACL) System Demonstrations*, pages 55–60.

Thomas Manzini, Lim Yao Chong, Alan W Black, and Yulia Tsvetkov. 2019. Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 615–621, Minneapolis, Minnesota. Association for Computational Linguistics.

Chandler May, Alex Wang, Shikha Bordia, Samuel R. Bowman, and Rachel Rudinger. 2019. On measuring social biases in sentence encoders. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 622–628, Minneapolis, Minnesota. Association for Computational Linguistics.

Pierre-Emmanuel Mazaré, Samuel Humeau, Martin Raison, and Antoine Bordes. 2018. Training millions of personalized dialogue agents. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2775–2779, Brussels, Belgium. Association for Computational Linguistics.

Robert R McCrae and Paul T Costa Jr. 1997. Personality trait structure as a human universal. *American psychologist*, 52(5):509.

Gábor Melis, Tomáš Kočiský, and Phil Blunsom. 2019. Mogrifier lstm. *arXiv preprint arXiv:1909.01792*.

Stephen Merity. 2019. Single headed attention rnn: Stop thinking with your head. *arXiv preprint arXiv:1911.11423*.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018a. An analysis of neural language modeling at multiple scales. *arXiv preprint arXiv:1803.08240*.

Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018b. Regularizing and optimizing LSTM language models. In *International Conference on Learning Representations (ICLR)*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *International Conference on Learning Representations (ICLR)*.

Tomas Mikolov and Geoffrey Zweig. 2012. Context dependent recurrent neural network language model. In *2012 IEEE Spoken Language Technology Workshop (SLT)*, pages 234–239. IEEE.

Umashanthi Pavalanathan and Jacob Eisenstein. 2015. Confounds and consequences in geotagged twitter data. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2138–2148, Lisbon, Portugal. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Caroline Criado Perez. 2019. *Invisible women: Exposing data bias in a world designed for men*. Random House.

Martin Potthast, Felix Schremmer, Matthias Hagen, and Benno Stein. 2018. Overview of the author obfuscation task at pan 2018: A new approach to measuring safety. In *CLEF*.

Francisco Rangel, Paolo Rosso, Moshe Koppel, Efstathios Stamatatos, and Giacomo Inches. 2013. Overview of the author profiling task at pan 2013. In *CLEF Conference on Multilingual and Multimodal Information Access Evaluation*, pages 352–365. CELCT.

Delip Rao, David Yarowsky, Abhishek Shreevats, and Manaswi Gupta. 2010. Classifying latent user attributes in twitter. In *Proceedings of the 2nd international workshop on Search and mining user-generated contents*, pages 37–44. ACM.

Neil M Richards. 2013. The dangers of surveillance. *Harv. L. Rev.*, 126:1934.

Charles Welch, Jonathan K. Kummerfeld, Verónica Pérez-Rosas, and Rada Mihalcea. 2020a. Exploring the value of personalized word embeddings. In *Proceedings of the 28th International Conference on Computational Linguistics*.

Charles Welch, Rada Mihalcea, and Jonathan K. Kummerfeld. 2020b. Improving low compute language modeling with in-domain embedding initialisation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*.

Charles Welch, Verónica Pérez-Rosas, Jonathan K. Kummerfeld, and Rada Mihalcea. 2019a. Learning from personal longitudinal dialog data. *IEEE Intelligent systems*, 34(4).

Charles Welch, Verónica Pérez-Rosas, Jonathan K. Kummerfeld, and Rada Mihalcea. 2019b. Look who's talking: Inferring speaker attributes from personal longitudinal dialog. In *Proceedings of the 20th International Conference on Computational Linguistics and Intelligent Text Processing (CICLing)*, La Rochelle, France.

Tsung-Hsien Wen, Aaron Heidel, Hung-yi Lee, Yu Tsao, and Lin-Shan Lee. 2013. Recurrent neural network based language model personalization by social network crowdsourcing. In *Interspeech*, pages 2703–2707.

Andrew Yates, Arman Cohan, and Nazli Goharian. 2017. Depression and self-harm risk assessment in online forums. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2968–2978, Copenhagen, Denmark. Association for Computational Linguistics.

Ziqian Zeng, Yichun Yin, Yangqiu Song, and Ming Zhang. 2017. Socialized word embeddings. In *International Joint Conferences on Artificial Intelligence*, pages 3915–3921.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL)*.

A Annotation of Demographic Attributes

In order to verify the accuracy of our demographic attribute assignment, we manually annotated a sample of 100 users from the dataset. Our extraction of attributes with regular expressions and rules was meant to have high-precision. It is likely that more attributes marked 'unknown' by our extraction could be filled in upon manual inspection. We evaluate the retrieved attributes for these 100 users by viewing the set of all posts that matched our extraction rules and attempting to annotate age, religion, gender, and location. The annotation instructions were to identify the value of these four attributes based on the annotators interpretation of the text of the posts. Then, for cases where the extracted attribute is not 'unknown', we calculate the percentage of times that they are the same. We get 94% for location and gender, 78% for religion, and 96% for age. It should also be noted that despite the annotators best efforts, it is not possible to know the actual ground truth values.

B Reproducibility Criteria

For each item in the list we have a section below with the relevant information.

B.1 Experimental Results

A clear description of the mathematical setting, algorithm, and/or model. The model we use is described in Merity et al. (2018b). We modify it to support weight freezing and initialization. In Section 2 where they describe the weight-dropped LSTM, we concatenate our vectors for user-specific and demographic representations to x_t.

The embeddings are obtained from the model described in Bamman et al. (2014) for the demographic and user matrices. To obtain demographic vectors, we treat C, from Section 2, as a matrix whose rows represent the demographic attribute of a speaker (e.g. male, female) independent of the word used. The model updates the same way, changing a generic word vector and relevant demographic attribute vectors when backpropagating.

A link to a downloadable source code, with specification of all dependencies, including external libraries

- AWD-LSTM code is available from `https://github.com/salesforce/awd-lstm-lm`.
- Embedding code is available from `https://github.com/dbamman/geoSGLM`.

Code modifications will be available at `http://lit.eecs.umich.edu`. We use PyTorch 1.0.1 with CUDA 10.0.103.

Description of computing infrastructure used Each model is trained on one NVIDIA Tesla V100 GPU.

Average runtime for each approach Our methods take between 260 and 1450 seconds per epoch depending on the approach.

Number of parameters in each model The number of parameters for the model that uses all four demographic attributes has the most parameters at 249,752,492. Our smallest model is the user representation comparison which has 48,066,614.

Corresponding validation performance for each reported test result Validation perplexities are reported for the 2+Dem validation set. Table 5 validation perplexities:

- demographic matrices ppl 62.57
- 500d baseline ppl 127.54
- 100d baseline ppl 124.39
- 100d generic ppl 111.70
- demographic age ppl 109.61
- demographic location ppl 109.93
- demographic gender ppl 109.68
- demographic religion ppl 110.88

Table 6 validation perplexities:

- user vectors ppl 69.59
- user matrices ppl 62.11
- demographic matrices ppl 62.57
- generic ppl 65.44

Explanation of evaluation metrics used, with links to code Perplexity for language models is common and is implemented in Merity's code. The word association metrics for best, oo3, and oo10 are described in Garimella et al. (2017) and we have reimplemented these metrics in order to compare to their results.

B.2 Hyperparameter Search

In our initial experiments on the 2+Dem validation set we chose the highest performing hyperparameters from the following list. One value listed means we used this value as described in Merity's code. Parameters were manually tuned and the best validation perplexity was chosen to use for all experiments.

- embedding dimension: [100, 500] – best 100
- LSTM hidden size (nhid): [550, 1150] – best 1150
- wdrop: 0.0
- dropouti: 0.5
- dropouth: 0.5
- dropoute: 0.1
- embedding composition function: [addition, concatenation] – best concatenation
- tied weights: [true, false] – best false
- frozen pretrained embeddings: [true, false] – best true
- LSTM layers (nlayers): 3
- learning rate: [10, 30] – best 30

We also experimented with embedding dropout masks. We initially had separate masks for the generic and concatenated demographic-specific embeddings but if one is masked and not the other it doesn't mask all information about that word. When we tried embedding dropout with the same mask for each concatenated vector perplexity dropped several points.

The vocabulary size for demographic experiments was 502k, while the experiments for individual users had a vocabulary size of 177k words.

B.3 Datasets

Relevant statistics such as number of examples See section 3 of the paper.

Details of train/validation/test splits See Table 2 for details of the 2+Dem and 4Dem experiments and Section 5.2 for details on the user representation comparison.

Explanation of any data that were excluded, and all pre-processing steps See Section 3.2.

A link to a downloadable version of the data
The Reddit data from 2007-2015 is available
from https://www.reddit.com/r/datasets/
comments/3bxlg7/i_have_every_publicly_
available_reddit_comment/. Our subset of
demographic labeled comments will be available
upon request and our code can be used to label
more Reddit data from after this collection was
posted.

**For new data collected, a complete description
of the data collection process, such as instruc-
tions to annotators and methods for quality con-
trol.** See Section 3 for data collection details and
the beginning of this supplemental material for de-
tails on the annotation process.