Predictive Embeddings for Hate Speech Detection on Twitter

Rohan Kshirsagar[1] **Tyrus Cukuvac**[1] **Kathleen McKeown**[1] **Susan McGregor**[2]
[1]Department of Computer Science at Columbia University
[2]School of Journalism at Columbia University
rmk2161@columbia.edu thc2125@columbia.edu
kathy@cs.columbia.edu sem2196@columbia.edu

Abstract

We present a neural-network based approach to classifying online hate speech in general, as well as racist and sexist speech in particular. Using pre-trained word embeddings and max/mean pooling from simple, fully-connected transformations of these embeddings, we are able to predict the occurrence of hate speech on three commonly used publicly available datasets. Our models match or outperform state of the art F1 performance on all three datasets using significantly fewer parameters and minimal feature preprocessing compared to previous methods.

1 Introduction

The increasing popularity of social media platforms like Twitter for both personal and political communication (Lapowsky, 2017) has seen a well-acknowledged rise in the presence of toxic and abusive speech on these platforms (Hillard, 2018; Drum, 2017). Although the terms of services on these platforms typically forbid hateful and harassing speech, enforcing these rules has proved challenging, as identifying hate speech speech at scale is still a largely unsolved problem in the NLP community. Waseem and Hovy (2016), for example, identify many ambiguities in classifying abusive communications, and highlight the difficulty of clearly defining the parameters of such speech. This problem is compounded by the fact that identifying abusive or harassing speech is a challenge for humans as well as automated systems.

Despite the lack of consensus around what constitutes abusive speech, *some* definition of hate speech must be used to build automated systems to address it. We rely on Davidson et al. (2017)'s definition of hate speech, specifically: "language that is used to express hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group."

In this paper, we present a neural classification system that uses minimal preprocessing to take advantage of a modified Simple Word Embeddings-based Model (Shen et al., 2018) to predict the occurrence of hate speech. Our classifier features:

- A simple deep learning approach with few parameters enabling quick and robust training
- Significantly better performance than two other state of the art methods on publicly available datasets
- An interpretable approach facilitating analysis of results

In the following sections, we discuss related work on hate speech classification, followed by a description of the datasets, methods and results of our study.

2 Related Work

Many efforts have been made to classify hate speech using data scraped from online message forums and popular social media sites such as Twitter and Facebook. Waseem and Hovy (2016) applied a logistic regression model that used one- to four-character n-grams for classification of tweets labeled as racist, sexist or neither. Davidson et al. (2017) experimented in classification of hateful as well as offensive but not hateful tweets. They applied a logistic regression classifier with L2 regularization using word level n-grams and various part-of-speech, sentiment, and tweet-level metadata features.

Additional projects have built upon the data sets created by Waseem and/or Davidson. For example, Park and Fung (2017) used a neural network approach with two binary classifiers: one to predict the presence abusive speech more generally, and another to discern the form of abusive speech.

Zhang et al. (2018), meanwhile, used pre-trained word2vec embeddings, which were then fed into a convolutional neural network (CNN) with max pooling to produce input vectors for a Gated Recurrent Unit (GRU) neural network. Other researchers have experimented with using metadata features from tweets. Founta et al. (2018) built a classifier composed of two separate neural networks, one for the text and the other for metadata of the Twitter user, that were trained jointly in interleaved fashion. Both networks used in combination - and especially when trained using transfer learning - achieved higher F1 scores than either neural network classifier alone.

In contrast to the methods described above, our approach relies on a simple word embedding (SWEM)-based architecture (Shen et al., 2018), reducing the number of required parameters and length of training required, while still yielding improved performance and resilience across related classification tasks. Moreover, our network is able to learn flexible vector representations that demonstrate associations among words typically used in hateful communication. Finally, while metadata-based augmentation is intriguing, here we sought to develop an approach that would function well even in cases where such additional data was missing due to the deletion, suspension, or deactivation of accounts.

3 Data

In this paper, we use three data sets from the literature to train and evaluate our own classifier. Although all address the category of hateful speech, they used different strategies of labeling the collected data. Table 1 shows the characteristics of the datasets.

Data collected by Waseem and Hovy (2016), which we term the **Sexist/Racist (SR)** data set[1], was collected using an initial Twitter search followed by analysis and filtering by the authors and their team who identified 17 common phrases, hashtags, and users that were indicative of abusive speech. Davidson et al. (2017) collected the **HATE dataset** by searching for tweets using a lexicon provided by *Hatebase.org*. The final data set we used, which we call **HAR**, was collected by Golbeck et al. (2017); we removed all retweets reducing the dataset to 20,000 tweets. Tweets were labeled as "Harrassing" or "Non-Harrassing"; hate speech was not explicitly labeled, but treated as an unlabeled subset of the broader "Harrassing" category(Golbeck et al., 2017).

Dataset	Labels and Counts			Total
SR	Sexist 3086	Racist 1924	Neither 10,898	**15,908**
HATE		Hate Speech 1430	Not Hate Speech 23,353	**24,783**
HAR		Harassment 5,285	Non Harassing 15,075	**20,360**

Table 1: Dataset Characteristics

4 Transformed Word Embedding Model (TWEM)

Our training set consists of N examples $\{X^i, Y^i\}_{i=1}^N$ where the input X^i is a sequence of tokens $w_1, w_2, ..., w_T$, and the output Y^i is the numerical class for the hate speech class. Each input instance represents a Twitter post and thus, is not limited to a single sentence.

We modify the SWEM-concat (Shen et al., 2018) architecture to allow better handling of infrequent and unknown words and to capture non-linear word combinations.

4.1 Word Embeddings

Each token in the input is mapped to an embedding. We used the 300 dimensional embeddings for all our experiments, so each word w_t is mapped to $x_t \in \mathbb{R}^{300}$. We denote the full embedded sequence as $x_{1:T}$. We then transform each word embedding by applying 300 dimensional 1-layer Multi Layer Perceptron (MLP) W_t with a Rectified Liner Unit (ReLU) activation to form an updated embedding space $z_{1:T}$. We find this better handles unseen or rare tokens in our training data by projecting the pretrained embedding into a space that the encoder can understand.

4.2 Pooling

We make use of two pooling methods on the updated embedding space $z_{1:T}$. We employ a max pooling operation on $z_{1:T}$ to capture salient word

[1] Some Tweet IDs/users have been deleted since the creation, so the total number may differ from the original

features from our input; this representation is denoted as m. This forces words that are highly indicative of hate speech to higher positive values within the updated embedding space. We also average the embeddings $z_{1:T}$ to capture the overall meaning of the sentence, denoted as a, which provides a strong conditional factor in conjunction with the max pooling output. This also helps regularize gradient updates from the max pooling operation.

4.3 Output

We concatenate a and m to form a document representation d and feed the representation into a 50 node 2 layer MLP followed by ReLU Activation to allow for increased nonlinear representation learning. This representation forms the preterminal layer and is passed to a fully connected softmax layer whose output is the probability distribution over labels.

5 Experimental Setup

We tokenize the data using Spacy (Honnibal and Johnson, 2015). We use 300 Dimensional Glove Common Crawl Embeddings (840B Token) (Pennington et al., 2014) and fine tune them for the task. We experimented extensively with pre-processing variants and our results showed better performance without lemmatization and lower-casing (see supplement for details). We pad each input to 50 words. We train using RMSprop with a learning rate of .001 and a batch size of 512. We add dropout with a drop rate of 0.1 in the final layer to reduce overfitting (Srivastava et al., 2014), batch size, and input length empirically through random hyperparameter search.

All of our results are produced from 10-fold cross validation to allow comparison with previous results. We trained a logistic regression baseline model (line 1 in Table 2) using character ngrams and word unigrams using TF*IDF weighting (Salton and Buckley, 1987), to provide a baseline since HAR has no reported results. For the SR and HATE datasets, the authors reported their trained best logistic regression model's[2] results on their respective datasets.

Method	**SR**	**HATE**	**HAR**
LR(Char-gram + Unigram)	0.79	0.85	0.68
LR(Waseem and Hovy, 2016)	0.74	-	-
LR (Davidson et al., 2017)	-	0.90	-
CNN (Park and Fung, 2017)	0.83	-	-
GRU Text (Founta et al., 2018)	0.83	0.89	-
GRU Text + Metadata	**0.87**	0.89	-
TWEM (Ours)	0.86	**0.924**	**0.71**

Table 2: F1 Results[3]

6 Results and Discussion

The approach we have developed establishes a new state of the art for classifying hate speech, outperforming previous results by as much as 12 F1 points. Table 2 illustrates the robustness of our method, which often outperform previous results, measured by weighted F1. [4]

Using the Approximate Randomization (AR) Test (Riezler and Maxwell, 2005), we perform significance testing using a 75/25 train and test split

to compare against (Waseem and Hovy, 2016) and (Davidson et al., 2017), whose models we reimplemented. We found 0.001 significance compared to both methods. We also include in-depth precision and recall results for all three datasets in the supplement.

Our results indicate better performance than several more complex approaches, including Davidson et al. (2017)'s best model (which used word and part-of-speech ngrams, sentiment, readability, text, and Twitter specific features), Park and Fung (2017) (which used two fold classification and a hybrid of word and character CNNs, using approximately twice the parameters we use excluding the word embeddings) and even recent work by Founta et al. (2018), (whose best model relies on GRUs, metadata including popularity, network reciprocity, and subscribed lists).

On the SR dataset, we outperform Founta et al. (2018)'s text based model by 3 F1 points, while just falling short of the Text + Metadata Interleaved Training model. While we appreciate the potential added value of metadata, we believe a tweet-only classifier has merits because retriev-

[2] Features described in Related Works section

[3] SR: Sexist/Racist (Waseem and Hovy, 2016), HATE: Hate (Davidson et al., 2017) HAR: Harassment (Golbeck et al., 2017)

[4] This was used in previous work, as confirmed by checking with authors

ing features from the social graph is not always tractable in production settings. Excluding the embedding weights, our model requires 100k parameters , while Founta et al. (2018) requires 250k parameters.

6.1 Error Analysis

False negatives[5]

Many of the false negatives we see are specific references to characters in the TV show "My Kitchen Rules", rather than something about women in general. Such examples may be innocuous in isolation but could potentially be sexist or racist in context. While this may be a limitation of considering only the content of the tweet, it could also be a mislabel.

> Debra are now my most hated team on #mkr after least night's ep. Snakes in the grass those two.

Along these lines, we also see correct predictions of innocuous speech, but find data mislabeled as hate speech:

> @LoveAndLonging ...how is that example "sexism"?

> @amberhasalamb ...in what way?

Another case our classifier misses is problematic speech within a hashtag:

> :D @nkrause11 Dudes who go to culinary school: #why #findawife #notsexist :)

This limitation could be potentially improved through the use of character convolutions or subword tokenization.

False Positives

In certain cases, our model seems to be learning user names instead of semantic content:

> RT @GrantLeeStone: @MT8_9 I don't even know what that is, or where it's from. Was that supposed to be funny? It wasn't.

Since the bulk of our model's weights are in the embedding and embedding-transformation matrices, we cluster the SR vocabulary using these transformed embeddings to clarify our intuitions about the model (8). We elaborate on our clustering approach in the supplement. We see that the model learned general semantic groupings of words associated with hate speech as well as specific idiosyncrasies related to the dataset itself (e.g. *katieandnikki*)

Cluster	Tokens
Geopolitical and religious references around Islam and the Middle East	bomb, mobs, jewish, kidnapped, airstrikes, secularization, ghettoes, islamic, burnt, murderous, penal, traitor, intelligence, molesting, cannibalism
Strong epithets and adjectives associated with harassment and hatespeech	liberals, argumentative, dehumanize, gendered, stereotype, sociopath,bigot, repressed, judgmental, heinous, misandry, shameless, depravity, scumbag,
Miscellaneous	turnt, pedophelia, fricken, exfoliated, sociolinguistic, proph, cissexism, guna, lyked, mobbing, capsicums, orajel, bitchslapped, venturebeat, hairflip, mongodb, intersectional, agender
Sexist related epithets and hashtags	malnourished, katieandnikki, chevapi, dumbslut, mansplainers, crazybitch, horrendousness, justhonest, bile, womenaretoohardtoanimate,
Sexist, sexual, and pornographic terms	actress, feminism, skank, breasts, redhead, anime, bra, twat, chick, sluts, trollop, teenage, pantyhose, pussies, dyke, blonds,

Table 3: Projected Embedding Cluster Analysis from SR Dataset

7 Conclusion

Despite minimal tuning of hyper-parameters, fewer weight parameters, minimal text preprocessing, and no additional metadata, the model performs remarkably well on standard hate speech datasets. Our clustering analysis adds interpretability enabling inspection of results.

Our results indicate that the majority of recent deep learning models in hate speech may rely on word embeddings for the bulk of predictive power and the addition of sequence-based parameters provide minimal utility. Sequence based approaches are typically important when phenomena such as negation, co-reference, and context-dependent phrases are salient in the text and thus, we suspect these cases are in the minority for publicly available datasets. We think it would be valuable to study the occurrence of such linguistic phenomena in existing datasets and construct new datasets that have a better representation of subtle forms of hate speech. In the future, we plan to investigate character based representations, using character CNNs and highway layers (Kim et al.,

[5] Focused on the SR Dataset (Waseem and Hovy, 2016)

2016) along with word embeddings to allow robust representations for sparse words such as hashtags.

References

Thomas Davidson, Dana Warmsley, Michael Macy, and Ingmar Weber. 2017. Automated hate speech detection and the problem of offensive language. *arXiv preprint arXiv:1703.04009*.

Kevin Drum. 2017. Twitter Is a Cesspool, But It's Our Cesspool. *Mother Jones*.

Antigoni-Maria Founta, Despoina Chatzakou, Nicolas Kourtellis, Jeremy Blackburn, Athena Vakali, and Ilias Leontiadis. 2018. A unified deep learning architecture for abuse detection. *arXiv preprint arXiv:1802.00385*.

Jennifer Golbeck, Zahra Ashktorab, Rashad O Banjo, Alexandra Berlinger, Siddharth Bhagwan, Cody Buntain, Paul Cheakalos, Alicia A Geller, Quint Gergory, Rajesh Kumar Gnanasekaran, et al. 2017. A large labeled corpus for online harassment research. In *Proceedings of the 2017 ACM on Web Science Conference*, pages 229–233. ACM.

Graham Hillard. 2018. Stop Complaining about Twitter — Just Leave It. *National Review*.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1373–1378, Lisbon, Portugal. Association for Computational Linguistics.

Ian T Jolliffe. 1986. Principal component analysis and factor analysis. In *Principal component analysis*, pages 115–128. Springer.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. 2016. Character-aware neural language models. In *AAAI*, pages 2741–2749.

Issie Lapowsky. 2017. Trump faces lawsuit over his Twitter blocking habits. *Wired*.

Ji Ho Park and Pascale Fung. 2017. One-step and two-step classification for abusive language detection on twitter. *CoRR*, abs/1706.01206.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Stefan Riezler and John T Maxwell. 2005. On some pitfalls in automatic evaluation and significance testing for mt. In *Proceedings of the ACL workshop on* *intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 57–64.

Gerard Salton and Chris Buckley. 1987. Term weighting approaches in automatic text retrieval. Technical report, Cornell University.

Dinghan Shen, Guoyin Wang, Wenlin Wang, Martin Renqiang Min, Qinliang Su, Yizhe Zhang, Ricardo Henao, and Lawrence Carin. 2018. On the use of word embeddings alone to represent natural language sequences.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*, pages 88–93.

Z. Zhang, D. Robinson, and J. Tepper. 2018. Detecting hate speech on twitter using a convolution-gru based deep neural network. © 2018 Springer Verlag.

A Supplemental Material

We experimented with several different preprocessing variants and were surprised to find that reducing preprocessing improved the performance on the task for all of our tasks. We go through each preprocessing variant with an example and then describe our analysis to compare and evaluate each of them.

A.1 Preprocessing

Original text

> RT @AGuyNamed_Nick Now, I'm not sexist in any way shape or form but I think women are better at gift wrapping. It's the XX chromosome thing

Tokenize (Basic Tokenize: Keeps case and words intact with limited sanitizing)

> RT @AGuyNamed_Nick Now , I 'm not sexist in any way shape or form but I think women are better at gift wrapping . It 's the XX chromosome thing

Tokenize Lowercase: Lowercase the basic tokenize scheme

> rt @aguynamed_nick now , i 'm not sexist in any way shape or form but i think women are better at gift wrapping . it 's the xx chromosome thing

Token Replace: Replaces entities and user names with placeholder)

> ENT USER now , I 'm not sexist in any way shape or form but I think women are better at gift wrapping . It 's the xx chromosome thing

Token Replace Lowercase: Lowercase the Token Replace Scheme

> ENT USER now , i 'm not sexist in any way shape or form but i think women are better at gift wrapping . it 's the xx chromosome thing

We did analysis on a validation set across multiple datasets to find that the "Tokenize" scheme was by far the best. We believe that keeping the case in tact provides useful information about the user. For example, saying something in all CAPS is a useful signal that the model can take advantage of.

Preprocessing Scheme	**Avg. Validation Loss**
Token Replace Lowercase	0.47
Token Replace	0.46
Tokenize	0.32
Tokenize Lowercase	0.40

Table 4: Average Validation Loss for each Preprocessing Scheme

A.2 In-Depth Results

	Waseem 2016			Ours		
	P	R	F1	P	R	F1
none	0.76	0.98	0.86	0.88	0.93	**0.90**
sexism	0.95	0.38	0.54	0.79	0.74	**0.76**
racism	0.85	0.30	0.44	0.86	0.72	**0.78**
			0.74			**0.86**

Table 5: SR Results

	Davidson 2017			Ours		
	P	R	F1	P	R	F1
none	0.82	0.95	0.88	0.89	0.94	**0.91**
offensive	0.96	0.91	0.93	0.95	0.96	**0.96**
hate	0.44	0.61	**0.51**	0.61	0.41	0.49
			0.90			**0.924**

Table 6: HATE Results

Method	Prec	Rec	F1	Avg F1
Ours	0.713	0.206	**0.319**	**0.711**
LR Baseline	0.820	0.095	0.170	0.669

Table 7: HAR Results

A.3 Embedding Analysis

Since our method was a simple word embedding based model, we explored the learned embedding space to analyze results. For this analysis, we only use the max pooling part of our architecture to help analyze the learned embedding space because it encourages salient words to increase their values to be selected. We projected the original pre-trained embeddings to the learned space using the time distributed MLP. We summed the embedding dimensions for each word and sorted by the sum in descending order to find the 1000 most salient word embeddings from our vocabulary. We then ran PCA (Jolliffe, 1986) to reduce the dimensionality of the projected embeddings from 300 dimensions to 75 dimensions. This captured about 60% of the variance. Finally, we ran K means clustering for $k = 5$ clusters to organize the most salient embeddings in the projected space.

The learned clusters from the SR vocabulary were very illuminating (see Table 8); they gave insights to how hate speech surfaced in the datasets. One clear grouping we found is the misogynistic and pornographic group, which contained words like *breasts*, *blonds*, and *skank*. Two other clusters had references to geopolitical and religious issues in the Middle East and disparaging and resentful epithets that could be seen as having an intellectual tone. This hints towards the subtle pedagogic forms of hate speech that surface.

We ran silhouette analysis (Pedregosa et al., 2011) on the learned clusters to find that the clusters from the learned representations had a 35% higher silhouette coefficient using the projected embeddings compared to the clusters created from the original pre-trained embeddings. This reinforces the claim that our training process pushed hate-speech related words together, and words from other clusters further away, thus, structuring the embedding space effectively for detecting hate speech.

Cluster	**Tokens**
Geopolitical and religious references around Islam and the Middle East	bomb, mobs, jewish, kidnapped, airstrikes, secularization, ghettoes, islamic, burnt, murderous, penal, traitor, intelligence, molesting, cannibalism
Strong epithets and adjectives associated with harassment and hatespeech	liberals, argumentative, dehumanize, gendered, stereotype, sociopath,bigot, repressed, judgmental, heinous, misandry, shameless, depravity, scumbag,
Miscellaneous	turnt, pedophelia, fricken, exfoliated, sociolinguistic, proph, cissexism, guna, lyked, mobbing, capsicums, orajel, bitchslapped, venturebeat, hairflip, mongodb, intersectional, agender
Sexist related epithets and hashtags	malnourished, katieandnikki, chevapi, dumbslut, mansplainers, crazybitch, horrendousness, justhonest, bile, womenaretoohardtoanimate,
Sexist, sexual, and pornographic terms	actress, feminism, skank, breasts, redhead, anime, bra, twat, chick, sluts, trollop, teenage, pantyhose, pussies, dyke, blonds,

Table 8: Projected Embedding Cluster Analysis from SR Dataset

1 Supplemental Material

We experimented with several different preprocessing variants and were surprised to find that reducing preprocessing improved the performance on the task for all of our tasks. We go through each preprocessing variant with an example and then describe our analysis to compare and evaluate each of them.

1.1 Preprocessing

Original text

> RT @AGuyNamed_Nick Now, I'm not sexist in any way shape or form but I think women are better at gift wrapping. It's the XX chromosome thing

Tokenize (Basic Tokenize: Keeps case and words intact with limited sanitizing)

> RT @AGuyNamed_Nick Now , I 'm not sexist in any way shape or form but I think women are better at gift wrapping . It 's the XX chromosome thing

Tokenize Lowercase: Lowercase the basic tokenize scheme

> rt @aguynamed_nick now , i 'm not sexist in any way shape or form but i think women are better at gift wrapping . it 's the xx chromosome thing

Token Replace: Replaces entities and user names with placeholder)

> ENT USER now , I 'm not sexist in any way shape or form but I think women are better at gift wrapping . It 's the xx chromosome thing

Token Replace Lowercase: Lowercase the Token Replace Scheme

> ENT USER now , i 'm not sexist in any way shape or form but i think women are better at gift wrapping . it 's the xx chromosome thing

We did analysis on a validation set across multiple datasets to find that the ”Tokenize” scheme was by far the best. We believe that keeping the case in tact provides useful information about the user. For example, saying something in all CAPS is a useful signal that the model can take advantage of.

Preprocessing Scheme	**Avg. Validation Loss**
Token Replace Lowercase	0.47
Token Replace	0.46
Tokenize	0.32
Tokenize Lowercase	0.40

Table 1: Average Validation Loss for each Preprocessing Scheme

1.2 In-Depth Results

	Waseem ?			Ours		
	P	R	F1	P	R	F1
none	0.76	0.98	0.86	0.88	0.93	**0.90**
sexism	0.95	0.38	0.54	0.79	0.74	**0.76**
racism	0.85	0.30	0.44	0.86	0.72	**0.78**
			0.74			**0.86**

Table 2: SR Results

	Davidson ?			Ours		
	P	R	F1	P	R	F1
none	0.82	0.95	0.88	0.89	0.94	**0.91**
offensive	0.96	0.91	0.93	0.95	0.96	**0.96**
hate	0.44	0.61	**0.51**	0.61	0.41	0.49
			0.90			**0.924**

Table 3: HATE Results

Method	Prec	Rec	F1	Avg F1
Ours	0.713	0.206	**0.319**	**0.711**
LR Baseline	0.820	0.095	0.170	0.669

Table 4: HAR Results

1.3 Embedding Analysis

Since our method was a simple word embedding based model, we explored the learned embedding space to analyze results. For this analysis, we only use the max pooling part of our architecture to help analyze the learned embedding space because it encourages salient words to increase their values to be selected. We projected the original pre-trained embeddings to the learned space using the time distributed MLP. We summed the embedding dimensions for each word and sorted by the sum in descending order to find the 1000 most salient word embeddings from our vocabulary. We then ran PCA (?) to reduce the dimensionality of the projected embeddings from 300 dimensions to 75 dimensions. This captured about 60% of the variance. Finally, we ran K means clustering for $k = 5$ clusters to organize the most salient embeddings in the projected space.

The learned clusters from the SR vocabulary were very illuminating (see Table 5); they gave insights to how hate speech surfaced in the datasets. One clear grouping we found is the misogynistic and pornographic group, which contained words like *breasts*, *blonds*, and *skank*. Two other clusters had references to geopolitical and religious issues in the Middle East and disparaging and resentful epithets that could be seen as having an intellectual tone. This hints towards the subtle pedagogic forms of hate speech that surface.

We ran silhouette analysis (?) on the learned clusters to find that the clusters from the learned representations had a 35% higher silhouette coefficient using the projected embeddings compared to the clusters created from the original pre-trained embeddings. This reinforces the claim that our training process pushed hate-speech related words together, and words from other clusters further away, thus, structuring the embedding space effectively for detecting hate speech.

Cluster	**Tokens**
Geopolitical and religious references around Islam and the Middle East	bomb, mobs, jewish, kidnapped, airstrikes, secularization, ghettoes, islamic, burnt, murderous, penal, traitor, intelligence, molesting, cannibalism
Strong epithets and adjectives associated with harassment and hatespeech	liberals, argumentative, dehumanize, gendered, stereotype, sociopath,bigot, repressed, judgmental, heinous, misandry, shameless, depravity, scumbag,
Miscellaneous	turnt, pedophelia, fricken, exfoliated, sociolinguistic, proph, cissexism, guna, lyked, mobbing, capsicums, orajel, bitchslapped, venturebeat, hairflip, mongodb, intersectional, agender
Sexist related epithets and hashtags	malnourished, katieandnikki, chevapi, dumbslut, mansplainers, crazybitch, horrendousness, justhonest, bile, womenaretoohardtoanimate,
Sexist, sexual, and pornographic terms	actress, feminism, skank, breasts, redhead, anime, bra, twat, chick, sluts, trollop, teenage, pantyhose, pussies, dyke, blonds,

Table 5: Projected Embedding Cluster Analysis from SR Dataset