ComVec: Unsupervised Learning of Ideological Communities

Jeremiah Milbauer[1,2]*, Jay Dhanoa*[1]

[1]Undergraduate, University of Chicago Department of Computer Science
[2]Undergraduate, University of Chicago Department of Philosophy
{jmilbauer, jdhanoa} @ uchicago.edu

Abstract

First, we introduce a new kind of unsupervised language modeling task: learning representations of communities. Next, we investigate two distinct approaches to this task. The first is to consider a "community" to be a collection of individuals who interact with each other. To this end, we embed internet communities on the website *Reddit.com* using user activity between multiple subreddits. The second approach is to consider a "community" in terms of its linguistic properties, and in particular the specific dialect (or "code") spoken in that community. To this end, we take a state-of-the-art language model and use transfer learning to create distinct language models for each community on the website, and then compare these language models (which represent the dialect of the community) to generate meaningful embeddings.

Key words: language modeling, computational linguistics, unsupervised learning, human-computer interaction, computational social science, data-mining, representation learning

1. Introduction

Sociologists have long been concerned with the problem of social network analysis. The primary goal of this kind of research is to identify and understand social entities and the manner in which they interact. In recent years, due to the advent of large-scale digital social media, the Internet had become a prime area of study for social network analysis.

In particular, we are interested in the representation of ideological communities on the internet. It is widely believed that insulated communities on the internet form ideological echo chambers – but while research has been performed to compare the political preferences of a large number of users in a single social graph [1] – we were unable to find current research on how to identify discrete ideological communities or understand them. We feel a particular urgency in understanding how communities are related to each other, what communities (taken as a single entity) believe, and how those beliefs change over time.

In this paper, we focus mainly on the representation of a community (not the identification). We use data scraped from *reddit.com*, an online social media platform where users can submit posts and comment on them within independent "subreddit" communities. In particular, we selected a set of "subreddits" consisting of both widely popular, politically unaffiliated subreddits, and subreddits committed to a particular political orientation. We then experiment with two distinct approaches to the representation of community: a network-based approach, where communities are considered to be "collections of communicating individuals", and a linguistics-motivated approach where communities are considered to be "a space for communicating in a shared dialect".

For the network-based approach we model a community as simply a vector of the individuals that contribute to that community. For the language-based approach, we fine-tune a unique language model for each independent community and visualize those communities (using PCA, Isomap, MDS, and T-SNE) based on a dissimiliarity graph between each learned language model. This approach takes advantage of the "code-switching" phenomenon, in which language speakers change the way in which they speak depending on the context.

We were successful in using the dialect-based approach to identify what we hypothesized to be ideologically similar communities, in some cases. While the dimensions of the resulting embedding are difficult to understand, many of our visualizations demonstrate some sort of "left"-"right" spectrum. This sort of exploration does not easily lend itself to quantitative evaluation. However, in the case of the language-based approach, we were able to quantitatively measure our ability to model each community dialect – for communities with large amounts of data, we reached language model perplexity in the $20 - 30$ range. For communities with less available data, we typically had perplexity between 30 and 50. We also developed a simple model for predicting the source subreddit of a given n-gram, and used this to measure the confusion between different subreddits.

2. Related Work

Certainly language modeling is a very developed field. In our work, we rely on an initial pretrained language model developed at OpenAI [2]. We extend OpenAI's model with our own neural language model.

However, despite the general interest in language modeling there is very little work related to this particular question. Language models have been used to model dialects rather than language [3], but not with dialects as similar as those present in different English-speaking internet communities. We do not expect significant morphological or syntactic differences between the communities we model.

Past work has been done in relation to language modeling for low-resource languages, but typically deals with using unsupervised learning to augment a small dataset. [4].

Work has also been done using reddit as a source for sociological investigation [5] [6], but not with an understanding of communal ideology as the explicit goal.

3. Methods

Here we detail the two approaches we experimented with, and their results.

3.1. A Brief Note about Data and Computational Challenges

We downloaded monthly comment archives of Reddit from *https://files.pushshift.io/reddit/comments/*, dated from January 2015 to January 2019. In total, this dataset represented over 2 TB (uncompressed) of JSON data corresponding to posts from each subreddit, which made storing and processing it a relatively meaningful computational challenge.

Even after extraction from each subreddit, we found that the relatively large number of comments from some of the larger subreddits (e.g. AskReddit), in tandem with their length, meant that we had to devote a nontrivial amount of time to developing a tradeoff between RAM usage [1] and accelerating processes, either with multiprocessing (which is RAM intensive), or with GPU acceleration [2]

3.2. Network Approach

We consider each subreddit to be comprised of a set of posts P. Associated with every post p_i is a karma score u_i, which is the net score (upvotes - downvotes). We can use the karma score on reddit as a sort of metric for in-community agreement. Affiliated with every post is a controversial flag, which helps us differentiate between posts that have a low karma score (but with many upvotes and downvotes) and posts that have only received a few upvotes and no downvotes.

For each post, we additionally have affiliated with it a unique username $u_1, ..., u_k$. For the purposes of the network approach, we construct a sparse vector $S \in \mathbb{R}^k = \langle u_1, ..., u_k \rangle$, where k is the number of users, and u_k is the number of posts by user k in this subreddit. On constructing this $R \times K$ matrix, where R is the number of subreddits we consider, we perform dimensionality reduction with Isomap and TruncatedSVD [3] to 2 dimensions, and correspondingly visualize those results. We tuned the results of this on the number of nearest neighbors used in the Isomap calculation. We performed this calculation for several cutoffs of overall user post frequency [4]

3.3. Language Model Approach

First, we set up a recent, very successful language model developed by OpenAI (Vaswani, et al. 2017). Next, we imported pretrained weights, released by OpenAI, to use this language model as a base for a transfer-learned fully-connected feed-forward neural network used to predict the 5th word of a given 5-gram, called the *RedditLM*. This network was trained on 6000000 5-grams randomly sampled from *reddit.com*. Next, we uniquely fine-tuned the *RedditLM* for each of 34 subreddits, each either widely popular and apolitical or explicitly polarized, using between 15000 and 2000000 randomly sampled 5-grams per subreddit.

Sentences are first embedded with the OpenAI transformer

[1] Per figure **??**, we initially embed every token of every utterance into memory using the OpenAI Transformer. The process of doing so for something as large as AskReddit in order to create our training set is relatively memory intensive, and doubly so when we are running two separate instances of our code with different GPUs within the same system.

[2] GPU acceleration on each card is memory bound by the size of the models involved, along with the batch size that we were using.

[3] sklearn's built in implementations of PCA and CCA do not support sparse matrices, and a dense matrix implementation of this requires ¿64 GB of memory

[4] Wherein we only consider posts from users that posted more than t times across our considered subreddits within our four-year window

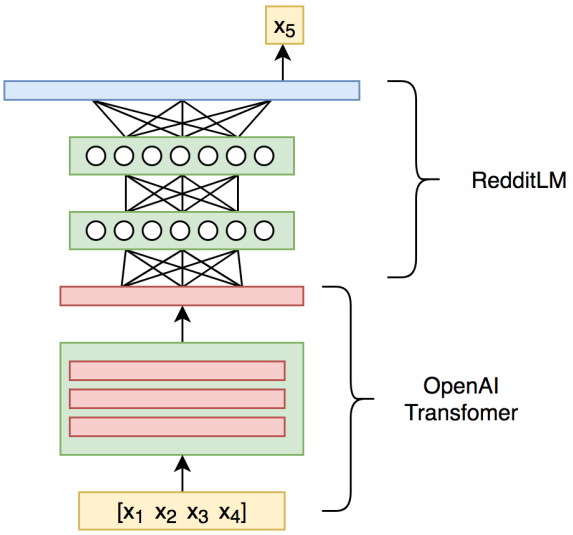

Figure 1: Diagram of the Language Model

into a vector of shape $(1, 768)$, and then the RedditLM is trained.

3.3.1. *OpenAI Transfomer Model*

The OpenAI language model is a "Transformer" model trained on a corpus of 7000 books. [2], [7]. For the sake of brevity we will not go into the details of the model here. It is possible to download pre-trained model weights from OpenAI. The pre-trained transformer model had a next-word-prediction accuracy of approximately 13% on reddit 5-grams.

We denote our use of the OpenAI model as:

$$\text{Tfm}(x_1, x_2, x_3, x_4)$$

where x_1, x_2, x_3, x_4 is a sequence of one-hot word embeddings using a vocabulary of 40,000 words, and the output of the function is the final layer of the OpenAI transfomer (diagrammed in red) which has dimension 1×768.

3.3.2. *RedditLM*

Our feed-forward language model is relatively simple.

We use the following architecture:

$$\begin{aligned} \text{fc1}(x) &= \text{ReLU}(W_1 x) \\ \text{fc2}(x) &= \text{ReLU}(W_2 x) \\ \text{out}(x) &= \text{Softmax}(W_3 x) \end{aligned}$$

Where W_1 is 768×256, W_2 is 256×256, and W_3 is $256 \times V$.

We train using cross entropy loss between the output distribution from out(fc2(fc1(x)) and the one-hot distribution for the actual next word in the 5-gram.

First, we pretrain *RedditLM* using 5000000 5-grams randomly sampled from reddit and embedded with $\mathit{Tfm}(x_1, x_2, x_3, x_4)$, over 10 epochs with a learning rate of .001.

For each subreddit, we embed as many 5-grams as possible from that subreddit (up to 2000000) using $\text{Tfm}(x_1, x_2, x_3, x_4)$

and then copy the weights from the pre-trained *RedditLM* and then fine tune the *RedditLM* over 10 to 200 epochs (depending on the number of available 5-grams). We intend for the *RedditLM* to be a representation of the dialect, so we are not particularly concerned about overfitting or generalization.

We use this to produce a set of 34 language models. For some dialect $d \in S$, the set of subreddit communities, we now have:

$$\text{RedditLM}_d$$

We can predict the next word in a given dialect/subreddit by taking:

$$\text{nextword}_d(x_1, x_2, x_3, x_4) = \arg\max_{x \in V} \text{RedditLM}_d(x_1, x_2, x_3, x_4)$$

4. Results

4.1. Network Modeling

One problem the network modeling apporach faced is that many users post across all different kinds of subreddits. Some users are active posters in both extremely conservative and extremely liberal communities. Visualizing the network-represented communities with Isomap, we saw that a cluster forms with the most popular subreddits all relatively similar, as they share users. We also saw that liberal-election related subreddits embed close to each other, and that some communities, namely "Conservative" (which is hyper-conservative, as opposed to "conservatives") and "videos", being very dissimilar to all the other political subreddits.

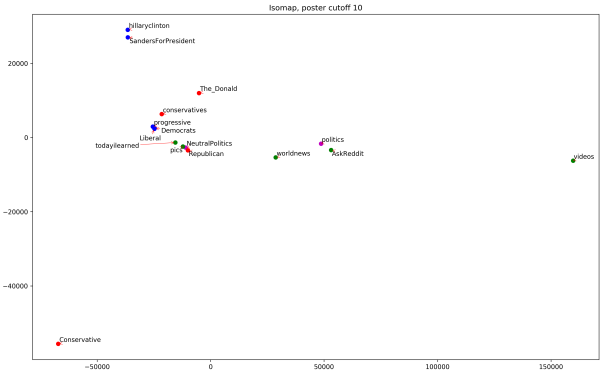

Figure 2: Network Visualization with Isomap, neighbors=7

4.2. Language Modeling

We considered using the resulting community-dialect-associated language models to identify the "dialect" of a given 5-gram from reddit. For some dialect d, and a 5-gram w, we can approximate the probability of a particular dialect being responsible for the word as:

$$p(d|w) = \frac{p_d(w_i|w_{i-4}, w_{i-3}, w_{i-2}, w_{i-1})p(w)}{p(d)}$$

if p_d is the score assigned to the continuation by the RedditLM_d We can ignore $p(w)$ because it is identical for every dialect, and our goal is to choose the dialect that maximizes $p(d|w)$. We also choose to reduce the effect of $p(d)$ (we use $\frac{\log(\text{gram\_count}(d))}{\text{gram\_count}(D)}$) because of the unfortunate data imbalance between subreddits – including this term results in nearly every prediction tending towards the $AskReddit$ community.

Using this technique we computed a confusion matrix over all the subreddits whose dialect we learned:

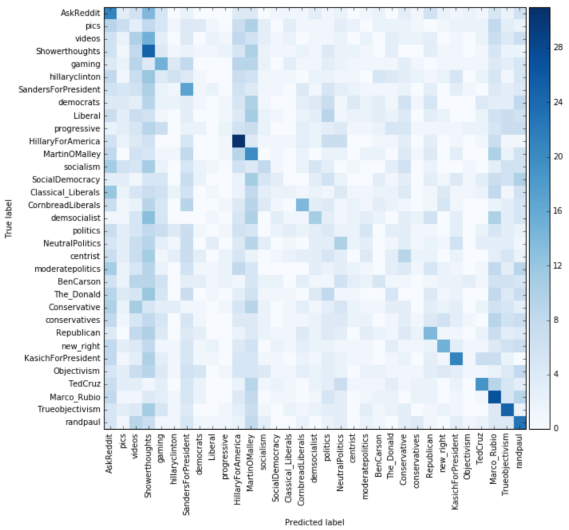

Figure 3: Confusion Matrix for Dialect Prediction

We noticed that certain subreddits (in particular randpaul, Trueobjectivism, Marco_Rubio, TedCruz, MartinOmalley, HillaryForPresident) seem to "claim" a lot of the sentences. This suggests that the language model learned for these subreddits was actually quite good compared to the other subreddits, or that there is significant regularity to the language of campaign-oriented subreddits. Eliminating these from the mix, we get:

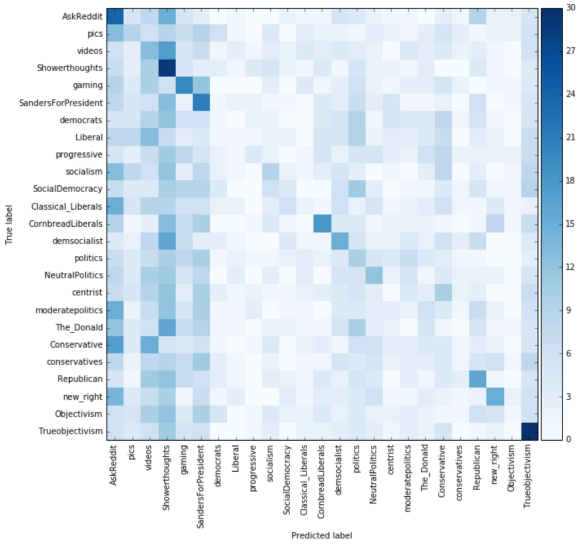

Figure 4: Confusion Matrix for Dialect Prediction, no campaigns

Again, prediction tends toward the extremely popular subreddits. Interestingly, we notice confusion between the widely-

used "politics" subreddit and "The_Donald" (a popular pro-Trump community) and the "SocialDemocracy", "Liberal", and "Democrats" subreddits – suggesting that those subreddits tend to have similar discussions to "politics". "Politics" itself gets confused with "moderatepolitics" and slightly with the allegedly moderate subreddits. Also interesting (to me) is the similarity between "conservative" and "centrist." Perhaps the "centrist" subreddit is not actually that "centrist".

We also used a "joint perplexity" metric to embed communities in space. For two subreddits, a and b, with sets of 5-grams s_a and s_b we took the "joint perplexity" as:

$$\text{joint\_perp}(a, b) = \text{perp}(a, s_b) + \text{perp}(b, s_a) - \text{perp}(a, s_a) - \text{perp}(b, s_b)$$

From this, we represent each community in a distance graph where its distance to each subreddit is its joint perplexity with that subreddit. We also experimented with representing each community as a vector of its perplexity with other subreddits. We looked at embeddings with PCA (on the vector-approach), MDS (neighbor-approach), Isomap (vector-approach), and T-SNE (neighbor-approach). Overall, there were some trends:

1. A dense central clumping of popular subreddits, probably due to inter-community conversation and the sheer availability of data leading to better language models.
2. A vague spectrum from "Left" to "Right"

This is demonstrated well by our MDS embedding:

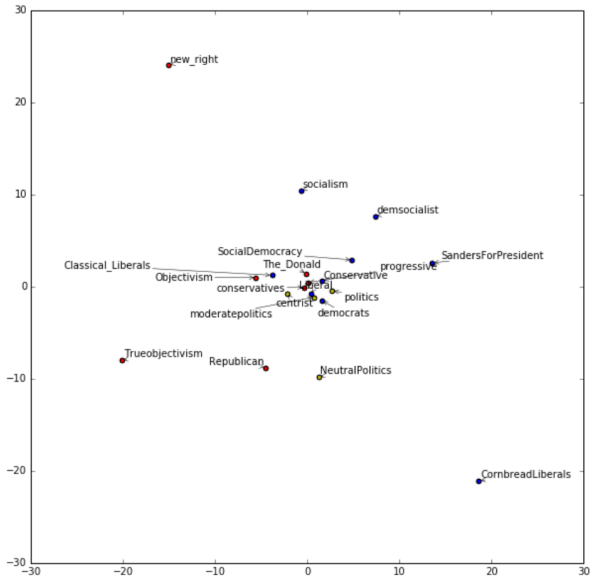

Figure 5: Joint Perplexity visualized with MDS

We observe a spectrum of left-to-right, literally embedded approximately left-to-right. We also see two more fringe political subreddits ("new_right" and "CornbreadLiberals", an alt-right group and a southern liberal group) more distant in linguistic space. It seems as though our x-axis might represent something like "political ideology."

We also see a spectrum trend in our results from T-SNE:

Here, we see left-wing subreddits on the left, and right wing subreddits on the right. It seems that neither the "Liberal" subreddit nor the "centrist" subreddit are both fairly unique. We also notice that the "gaming" subreddit is embedded close to the right-wing communities, and the "AskReddit" subreddit is closer to the left-wing communities. This conforms with some of the public criticism of the online gaming community [8].

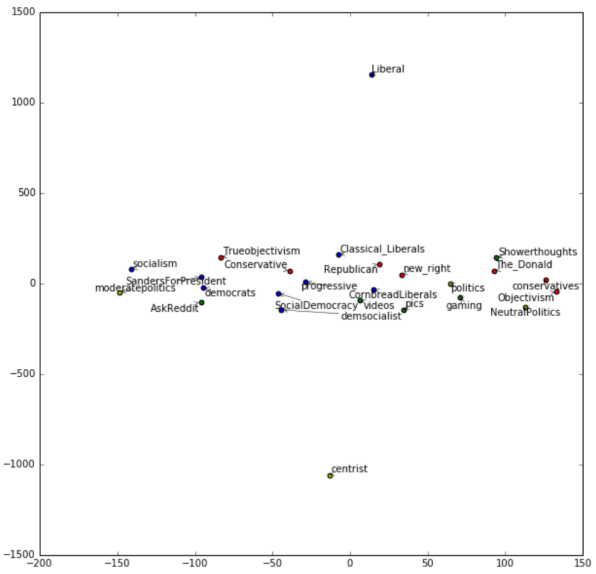

Figure 6: Joint Perplexity visualized with T-SNE (perplexity=20)

5. Discussion

5.1. Key Conclusion

The results of the language-modeling approach to community ideology representation are promising, but there is quite a ways to go for improvement. We have demonstrated that some degree of political ideology can be learned with a dialect-modeling approach, and that this approach identifies community similarities where a network-based appraach fails. As this technique is refined, it can prove to be an effective way of identifying an ideological space on the internet, a key component of understanding the spread of misinformation, stopping cyber-propaganda campaigns, and preventing radicalization.

5.2. Future Work

Future explorations of the approach to modeling community dialect could improve upon our language modeling approach and develop even better language models for each online community.

It would also be worth investigating different ways of using perplexity to compute subreddit similarity.

An extension of this work could involve developing an ideological space (using polarized political subreddits) and then computing language models for non-political subreddits using data from different time periods. Those subreddits could then be embedded in the static political space over time, detailing ideological transformations within a community.

Additionally, it would be worthwhile to explore a topic-modeling approach to community representation. Topics could be broadly learned across *Reddit* and communities could each be embedded in a topic space based on the key convesations in those communities. One might suppose that each community is

a sort of linear combination of discussion topics.

6. Acknowledgements / Respective Contributions

Jay did a lot of the heavy lifting relating to the preprocessing + extraction of the Reddit data. He additionally wrote out code for the network approach, and got the computationally heavy portions working with the GPU. One of the major challenges we had when working with this dataset was the scale of it, which meant that we had to do a lot of juggling of data between CPU, RAM, and disk while running code to keep enough loaded in memory to run it all quickly.

Jeremiah focused on defining our approach to transfer learning and fine-tuning the models, setting up the OpenAI Transformer Model, and writing the code for our neural net. He also produced the analysis of the language-model performance, and wrote up most of the report. Some of the big challenges for Jeremiah was thinking through OpenAI's paper, and designing a model that could work on top of the pretrained sentence embeddings, and thinking through ways of visualizing and understanding the language modeling approach.

Many of the challenges in the project were addressed by Jay and Jeremiah working together over a single computer, tweaking code and models until all the data types lined up, the code ran on the GPU, and everything went smoothly. What fun!

7. References

[1] Pablo Barbera, John T. Jost, Jonathan Nagler, Joshua Tucker, Richard Bonneau. "Tweeting from Left to Right: Is Online Political Communication More Than an Echo Chamber?" *Psychological Science*, 2015.

[2] Alex Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever, "Improving Language Understanding by Generative Pre-Training". *Preprint.* Accessed: https://openai.com/blog/language-unsupervised/

[3] Fadi Biadsy, Julia Hirschberg, and Nizar Habash, "Spoken Arabic dialect identification using phonotactic modeling," *Proceedings of the EACL 2009 Workshop on Computational Approaches to Semitic Languages*, pp.53-61, 2009.

[4] Anton Ragni, Kate M. Knill, Shakti P. Rath and Mark J. F. Gales, "Data augmentation for low resource languages," *INTERSPEECH*, 2014.

[5] Bryan Dosono, Bryan Semaan, Jeff Hemsley, "Exploring AAPI Identity Online: Political Ideology as a Factor Affecting Work on Reddit" *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pp.2528-2535, 2017.

[6] Cody Buntain and Jennifer Golbeck, "Identifying social roles in reddit using network structure", *Proceedings of the 23rd International Conference on World Wide Web*, pp. 615-620, 2014

[7] A. Vaswani, N. Shazeer, N. Parmar, J. UszKoreit, L. Jones, A. N. Gomez, L. Kaiser, I. Polosukhin, "Attention is All you Need" *Advances in Neural Information Processing Systems*, poster, 2017.

[8] Koshonna L. Gray and David J. Leonard, *Woke Gaming: Digital Challenges to Opression and Social Injustice*, University of Washington Press: 2018.

[4] Andrew M. Dai, Quoc V. Le, "Semi-supervised Sequence Learning", *Advances in Neural Information Processing Systems*, poster, 2015.