Università degli studi di Torino

SCUOLA DI SCIENZE DELLA NATURA

Corso di Laurea Magistrale in Fisica dei Sistemi Complessi

Tesi di Laurea Magistrale

Inference on features-based interactions in a task of link prediction

Application on Reddit activity

Relatore:
Prof.
Paolotti Daniela

Controrelatore:
Prof.
Panisson Andre'

Candidato:
D'Ignazi Jacopo

Anno Accademico 2020/2021

Abstract

In the last decades, social networks have become one of -if not the most- relevant places where we engage with others and develop our opinions. The role of digital media in shaping our society is thus undoubtedly one if the most important matter in scientific discussion and political debate, raising either interests and concerns.
New field of research, such as Computational Social Science, have born to deal with the unprecedented amount of traces left by users' activity on the internet: not only we can ask how social media impact our daily life, but also how we can study these data to have a better understanding of human behavior.

In this context, questions on human nature have been provided both new data and techniques to work with. It is now possible to ask *what are the factors that drive social interactions?* And also *how do opinions develop and shape up?* Given the impact that social structures can have on political matters, researchers are interested in evaluating how opinions might polarize through interactions on social media.

Following this line of research, in our work we were interested in studying how different individual characteristics could play a role on their social habits. We thus took into account both sociodemographic features (like age, gender and affluence) and their opinion (like partisanship preference, being an environmentalist, part of a opinion based community and more), and developed a probabilistic model on the interactions of users on the popular platform "Reddit".
Our goal was thus to see whether individuals had any preference on who to interact with, based on their characteristics. In particular, we were asking whether echo chambers could be observed and what features they could be associated with.
Our main findings is that, contrary to the common narrative of people interacting with only like-minded people, we observed that this is not the case in our dataset; we indeed observed how users of Reddit actually often preferred engaging in debate with opposite opinion users, while preferring talking to individuals of the same sociodemographic extraction.

In chapter 1 I will discuss our motivations and provide general overview on the topic. In section 2 I will briefly present the current state of research in the field, both on the matter on opinion dynamics and the techniques used in this context. In section 3 I will then present the details of the strategy we were using and the theoretical framework developed; in section 4 this strategy is validated both on synthetic data and compared against previous work, also discussing some improvement introduced by our approach. In chapter 5 I finally present the results from using our model on real data obtained from Reddit, concluding in chapter 6 highlighting the efficacy of our approach and an overview on our findings.

Contents

1 Introduction **1**
- 1.1 Our work . . . 2
- 1.2 Dataset . . . 4

2 State of the art **5**
- 2.1 Polarization and echo chambers . . . 5
- 2.2 Theoretical approaches . . . 7

3 Methodology **13**
- 3.1 Data Ingestion Pipeline . . . 13
  - 3.1.1 Users selection . . . 14
- 3.2 Features extraction . . . 16
  - 3.2.1 Features from embedding . . . 17
  - 3.2.2 Features from clustering . . . 18
  - 3.2.3 Local activity and popularity . . . 20
- 3.3 Text based analysis . . . 21
  - 3.3.1 Sentiment analysis . . . 21
  - 3.3.2 Topic recognition . . . 21
- 3.4 Model . . . 23
  - 3.4.1 Structure . . . 23
  - 3.4.2 Logit Model . . . 24
  - 3.4.3 Comparison with other models . . . 26
  - 3.4.4 Learning algorithms . . . 28

4 Validation **30**
- 4.1 Features validity and iterpretation . . . 31
  - 4.1.1 Embedding based features . . . 34
  - 4.1.2 Cluster based features . . . 35
- 4.2 Test on synthetic data . . . 37
  - 4.2.1 Generating features . . . 37
  - 4.2.2 Generating interactions . . . 39
  - 4.2.3 Observations . . . 40
- 4.3 Comparison with state of the art results . . . 45
  - 4.3.1 Models' equivalence . . . 46
  - 4.3.2 Coherence with expected results . . . 49
  - 4.3.3 Improvements . . . 52

5 Results **54**
5.1 Features statistics 54
5.2 Interactions' probability 56
5.2.1 Context dependent similarities 56
5.2.2 Common trends 59
5.3 Yearly trends 62
5.3.1 Sociodemographic features 62
5.3.2 Political opinion features 62
5.4 Interactions' sentiment 64

6 Conclusion **66**

Bibliography **75**

Chapter 1

Introduction

In a world where digital media are becoming more and more integrated in our daily lives, the task of evaluating how these new platforms impact our beliefs and habits is becoming more and more relevant; the rise of the internet has indeed provided people new possibilities for accessing information, interacting with each other, and different means through which this can happen.

Each of these interactions will leave some sort of trace, that can often be retrieved and studied in the context of what is generally called "Data science". These data are now being used for marketing purposes, monitoring mass opinions, either with arguably malicious purposes or scientific interest. Most importantly for us, from a scientific point of view, data of users' actions on the internet will carry information about social habits and the state of the world.
The scientific disciplines concerned with studying society and human behavior, like sociology and psychology, have indeed been provided a new perspective on the world, and new data about their object of study. Together with the growth of digital media we thus have seen in the last decades the rise of new scientific branches, that aim to pair with "soft sciences" in order to study the traces left by users on the internet.

This is the context that gave birth to so called "Computational Social Science" (CSS), which focuses on investigating social behavior through simulation, modeling, network analysis, and media analysis [1]. This new perspective on human behavior has led to many interesting findings, which shed lights on what drives our activities and how society ultimately shapes itself.
Topics of interest in CSS range from group formation, information diffusion, polarization of opinions and much more, with the objective of integrating knowledge of human behavior into sociology and the debate around social media. Recent works have provided insights about how individual opinion might change in social interactions [2], how polarization is shaped [3], and even how social network reacted to informations about the 2020 Coronavirus outbreak[4].

Among these results, and current object of research, a topic that has raised greater interest and concerns in recent years is the one of opinion polarization and echo chambers [3] [5] [6]. Polarization is a phenomenon where people's opinions tend to shape toward two opposite extremes, thus reducing mediation between counterparts, enhancing biases and conflict; it has been observed how polarization can occur either along

party lines, ideological lines and any resonant/divisive topic [7]. Echo chambers are instead social structures where individuals are surrounded by like-minded peers while insulated from others' points of view, and are thought to be a cause of polarization.

Polarization is becoming a defining feature of contemporary politics and it has been observed how this phenomenon is on the rise [8]. Concerns have been expressed on how social network could facilitate the formation of echo chambers [9], thus deepening the ideological divide and arguably endangering healty mass opinion development and constructive debate.
This is what led us to look into this issue in more details, asking what characteristics in individuals are more likely driving their interactions on social networks.

1.1 Our work

We focused our observations on years between 2016-2020 in "Reddit", a popular platform where users can engage in conversations, and freely discuss about any topic of their choice. Our dataset will so consist in comments of a user (author) responding to another (target), together with the features associated with the two.

In our work we were thus interested in looking if any echo chambers like structure could be observed, and eventually what are the specific characteristics that might lead to it. To look at the problem from a broader perspective, we took into account both ideological/opinion features (partisanship preferences, being an environmentalist, pro gun and so on) and socio-demographic ones (such as gender, age and affluence).
The question we were trying to answer was then: *based on their features, how could we expect users to interact with each other?* And specifically, to address the matter of echo chambers: *are individual preferring interactions with like minded people?*
In order to do so, we proceeded in two steps: we first extracted characteristics of interest from the data at our disposal, and then we followed the approaches of generative and discriminative models to evaluate their interactions.

The task of features extraction from users' activity is an interesting one on itself, since raw data from Reddit did not come with an obvious way to determine users' opinions and sociodemographic statuses. We thus made the *assumption that users' activity on the platform was in some way representative of their beliefs and social statuses*, allowing us the inference of their features based on what they were doing on Reddit itself.
In this line of thoughts we adopted different techniques and came up with a strategy that gave us reasonable results and a broad set of features types; these technique, namely one based on graph embedding and one based on communities clustering, are presented in details in sections 3.2.1 and 3.2.2. [1].

We then developed a theoretical framerowk where we could measure how much each feature pair (one of an author, the other of the target) contributed in driving their interactions. As it will be discussed in section 3.4, we indeed managed to develope a model where each parameter was related to the log odds of each feature pair leading

[1]Since concerns could be raised on the actual meaning and reliability of the features we extracted, the matter is further addressed in section 4.1; the actual results of this process are then presented in section 5.1

to an interaction. Within this framework, we discuss in sections 3.4.1 and 3.4.2 two different approaches that perform this task, and decided to applied both of them.

The first model of our choice was thus an instance of Markov Chain Monte Carlo method [10], a sampling based strategy to estimate posterior of model's parameter, commonly used in state of the art inference models [11][12]. The second one was instead an instance of Logistic regression, together with p-values and confidence intervals to establish statistical significance of our findings; this technique is also used in state of the art research [6], proven to be simple but reliable.
These two strategy came with upside and downsides, but in section 3.4.3 we proved how these two models are actually equivalent in certain scenarios; this result is then generalized to prove that indeed many realization of different models could be equivalent -or at most comparable- in many cases.

In section 4.2 we present our results on synthetic data, observing versatility and reliability of model's behaviour on different feature types and preprocessing choices. In section 4.3 we then compare our approach with state of the art results [6], observing not only coherence with their findings but also improvements in terms usability and results' interpretability.
These tests served us as model validation process, and we later proceeded to apply the whole pipeline on our datasets. Each dataset consisted in a year of comments observed in specific 'subreddit' (subsection of Reddit), preprocessed in order to remove noise and keep only meaningful interactions (as discussed in section 3.1). To gather more information on the interactions we later performed sentiment analysis on comment's text, and associated an average sentiment to each pair of features -as is presented in section 3.3.

This approach gave us not only the influence that each feature had on users' interactions, but also "the quality" of those. Results on the probability alone are presented in section 5.2 and 5.3, and the observed sentiment values in section 5.4. Through these different perspective we were able to establish how *no echo chambers was present in our datasets - and in certain cases interactions between opposite sided opinion were even more likely- although these cross-group interactions averagely had lower sentiment values.*

These findings tells us that arguing/debating with opposite-opinion users can be a relevant factor in driving interactions, and that echo chambers can not emerge when this happens; on the contrary we observed systematically assortative behavior with respect to sociodemographic features and segregative behavior of certain specific-opinion communities, which told us that topic-driven interactions could be a cause of division among users - although at the time of writing, further work is being done to explore this hypothesis.
We conclude in section 6 with a summary of our findings and an overview on our model's efficacy: while some process might still need to be refined, we were glad to see how this approach has proven to be either versatile and insightful in this use case.

Further work is currently being done to gather informations about how topic can influence different interactions, and to refine some step of the features extraction process. It whould be indeed interesting to observe how interactions probabilities and sentiments might change with respect to the topic the users are talking about.

With the current work, however, we managed to prove that echo chambers are not to be taken for granted, and users behavior in relation with their characteristics might respond to a more complex combination of factors. Furthermore we observed how individuals that belong to a community tend to interact mostly with others belonging to the same community, and that they spontaneously tend to talk to other of same sociodemographic extraction.

1.2 Dataset

Observations were made on data retrieved from Reddit, a popular social network organized in communities where users can discuss about any specific topic. These communities are called "subreddit" and are organized in a tree-like structure: in each subreddit, following certain subreddit rules, any user can make a submission to propose a topic to discuss. Different users can then comment on that submission and, as time goes on, the directed tree will grow from users responding to each others' comments. Reddit also implement a function of "upvote/downvote" which can be a useful mean to estabilish a comment's popularity.

By deciding their rules, each community (aka subreddit) will also influence the kind of interaction that can happend in it: certain communities require participants to be aligned to some specific opinions, while other might make no requirement other than providing a general topic of discussion.
For this reason, some of the subreddits can be good neutral grounds to observe how users freely interact with each other; on the contrary, the participation in some specific subreddit could be a good indication that a user is aligned with some specific set of beliefs.
Either this neutrality or polarization of communities can be exploited for our study: neutral grounds can provide good dataset to observe opinion based interactions, while polarized communities can be used to extract users' opinions themselves.

We thus chose the subreddits 'news' and 'politics' as dataset sources, and focused our attention on timeframes of one year. In these years and subreddits we chose to observe only the most active users, assuming their activity was most significant of the opinions they represented; we then looked at their activity on other subreddit to extimate their orientation on different features axis and whether they belong in some opinion related community.

This pipeline will be presented in chapter 3, going into details about what these "interaction probabilities" mean and how determined the activity based features.
We chose those perform our studies on these two specific subreddits alone, as we observed they were both really popular -providing great amount of data- and their rules were neutrally oriented. Nonetheless, our pipeline could be applied easily to different subreddit/year contexts, and the training process could be applied for the same task on any other social network.

Chapter 2

State of the art

In this section I will provide a brief overview about recent works on the topics of our interests: in section 2.1 I present the state of research about polarization and echo chambers, giving an introduction about commonly used techniques and recent findings in these previous works. I will also present what are the currently open questions in this field and how our research attempt can be useful to tackle some of them.
In section 2.2 I will go into more details about the kind of models used in these task, briefly introducing state of the art techniques and then explaining the ones we will be using in our work.

2.1 Polarization and echo chambers

Given the concerns expressed by public opinion and law makers [9], the topics of polarization and echo chambers has become a thriving field in last decade's research on computational social science. The questions the scientific community is trying to answer are ones such: *How can we measure polarization?*[3], *do echo chambers actually emerge in any context?* [6], *what events might lead to polarization?* [13], *how do different social network enhance formation of echo chambers?* [5].

Echo chambers can be observed through network analysis of interactions, where segregated community shows that inter-group interactions are much more frequent that cross-group ones; echo chambers are thus characterized -and defined- by highly modular region of a network, where one's opinion might be reinforced due to repeated interactions with others' sharing the same beliefs.
To measure this modularity, one can try to estimate probabilities of interactions between a proper partitioning of the graph: if a given partition is found to share some opinion, and their probability of inter-group interaction is far higher than cross-group ones, than that partition can be considered an echo chamber.
While quantifying this phenomenon has always a margin of arbitrariness, following this reasoning a good way to find echo chambers is thus to make use of generative models: the probabilities associated with feature-feature interactions can indeed give an indication of whether an echo chamber is present [6]. In the next section I will present in more details what these kinds of model are about, what families of these are used in this field, and provide an explanation on how they work.

Measuring polarization is also not of a trivial task, because it requires to somehow "quantify opinions" of individuals [2]. Text based opinion extraction methods can be used, together with community based ones; the latter, which is the approach we follow in our work, implies finding communities which clearly share some feature and then associate each user to different feature, on the base of its relation with each community. In this case, in the context of Reddit, this is precisely what "activity based features" means. An interesting way to perform this task makes use of graph embedding strategies [3]: summarizing, each community is represented as a vector of the users in it (in our case, the communities are subreddits), then different embedding techniques are applied to estimate a latent vector representation of that community. These representations can then be worked with as it is commonly done with embedded vectors, evaluating the distance between each other, projecting them on different axis and so on. In our case the axis represented socio-demographic and partisanship leaning of each community, and the embedded vectors representation of different subreddits were taken by a previous work [3].

Using these and many other techniques, efforts in this line of research has led to interesting results about echo chambers and polarization.
It has been observed how polarization is enhanced by divisive events such as political elections [3], for example in 2016 presidential election in America. This is also the case for different specific topics that appear to resonate with popular opinion [7], and that apparently sparked debate among social networks' communities in general.

It is at this point interesting to discuss how polarization and echo chambers can be related: while the former is a measure of the actual opinion distributions, the latter is by definition a matter of their interactions. It is obvious that the two can be strictly correlated, since different interaction tendencies might lead to certain opinion dynamics and vice versa [2]. Echo chambers have indeed been associated to selective exposure[14], biased assimilation [15], and group polarization [16]. Together with polarization, echo chambers have been observed in relation to specific controversial topics, such as abortion [7] and vaccines [17].

However, it can be argued that echo chambers might not be the only cause of polarization, or at most the relation between the two is not that obvious. We can look for example at 2016 political election in America, and users' activity in Reddit at that time: while polarization was increasing [3], it has been observed how users were mostly interacting with opposite-partisanship others [6].
This would suggest that echo chambers are not to be taken for granted, and neutral grounds might actually enhance cross-group interactions; it can also suggest that echo chambers might not be the only cause for polarization. On the contrary it might be argued how the debate happening through cross-group interactions was actually the reason of it, reinforcing the beliefs of individuals not by similarity with like-minded others but by distancing from opposite-minded ones [18].

For this reason, the actual existence of echo chambers has been put into scrutiny in recent years [6][19]. Our work will follow this line of research, looking at different users' features and estimating how much each of those was impacting their habits and what kind of interactions they were leading to.

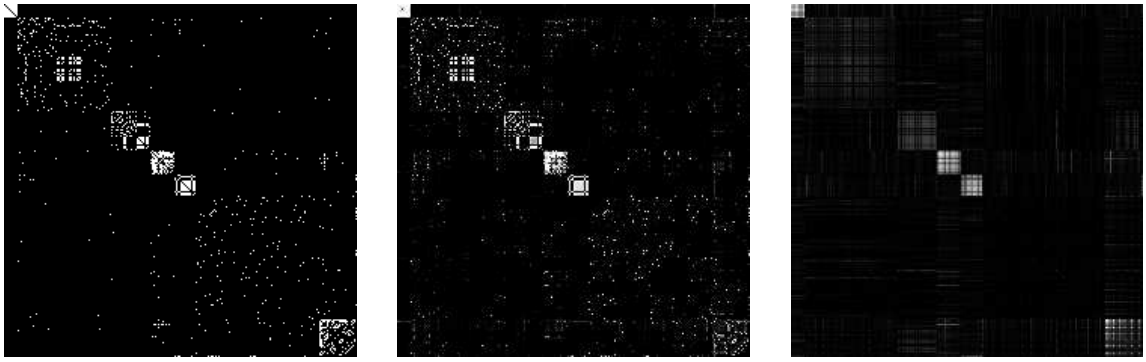

Figure 2.1: Different performances of link prediction models on a subset of the most active 234 authors of NIPS 1-17 cohautorship dataset; these pictures are taken from "Nonparametric Latent Feature Models for Link Prediction"[12]. In each of them, the rows and columns represent an author of scientific articles, and the citation from one author to another is depicted as a white pixel in the matrix. The figure on the left represent the true observed collaborations; figure on the right represents the extimated collaboration probabilities with an instance of SBM method; figure in the middle is the result of a generative model approach based on MCMC learning algorithm.

2.2 Theoretical approaches

In the field of Computational Social Science, statistics and machine learning techniques are developed to have some insight in human behavior. Some task in this field might require the use of articulated models, mixing together graph theory and machine learning [3], agent based models [2] and much more.
In general, since the research questions often results in estimating some "linking probability" between to users/group of individuals based on their features, a powerful approach can be Bayesian statistics.
In particular, the problem of estimating probabilities of a link between two users, on the base of their features, can be formulated in the realm of generative models.

Generative models are ones where one tries to estimate how the dataset is generated, assuming it follows given rules. In the context of features-features interactions this means assuming a theoretical structure of interactions (which can be referred to as the model itself) where one introduce some parametrization that regulates how these features interact; provided the features of the dataset, one then applies some strategy (a learning algorithm) to find the parametrization that maximizes the posterior.
A generative model can then be represented by a likelihood function, assigning a probability of an event/label Y (interactions in our case) to each element of the dataset X (pair of features vectors $X \equiv X_i, X_j$ in our case):

$$P_{M,h}(Y|X) \equiv P_{M,h}(Y = 1|X_i, X_j) \equiv \text{interaction between author and target} \quad (2.1)$$

Where M stands for the model's parametrization, and h the hyperparameters; *interactions* might stand for any kind of event related to the two nodes i, j, which in the simplest case is just a variable $Y = 1, 0$ of wether that interaction happened or not. In our case i will be the author of a comment, j the target, and $Y = 1$ represents the actual observation of user i responding to user j.

The goal is then to study the posterior of this model, in order to find a good parametrization that satisfy some criteria - commonly maximizing the posterior, but also being the average/median and so on. Using Bayes theorem, we are thus interested in the quantity:

$$P(X|Y,M,h) = \frac{P(Y|X,M,h)P(M,h)}{\int_M P(Y|X,M,h)P(M,h)dM} \tag{2.2}$$

Where $P(M,h)$ is the prior on the model's parameters. The advantage of Bayesian approach is to provide the possibility to introduce this prior, while the advantage in generative approach is to allow us to question: *given the observed dataset features, labels, model's prior and assuming they are related by that given likelihood, what are the parameters of that model that most likely generated the data we observed?*.
Different instances of this approach will then differ on the kind of features/labels we are using, the priors and the likelihood structure.

In this framework, the task of estimating feature-feature based interactions is often formalized through an interaction matrix W, which will define the model's parameters; this means that the likelihood probabilities are defined in some functional form like:

$$P_{W,h}(Y|X_i,X_j) \equiv P_h(Y|X_i^T W X_j) \tag{2.3}$$

In this framework, a common technique is the so called "Stochastic Block Modeling" (SBM) [20], where each individual is assigned a specific group and the interaction matrix represent the probability/amount of interactions between two pair of communities; this means that X_i, X_j are each one a vector of binary/continuous variable, where each element x_i^k represents the probability that user i belongs to community k. Variations of this approach can implement binary vectors, continuous ones/non mutually exclusive communities (i.e. each user can belong to more groups), hierarchical organization of communities and so on. The limit of this approach is the fact tht the communities found by such an approach are generally not interpretable, while our interest is looking at specific features and their cross-group interactions .

Other approaches have then been developed that make use of "Markov Chain Monte Carlo methods" (MCMC) [12], a general and versatile approach that can estimate posterior of a model[10]. Other approaches make instead use of classification algorithms, such as the Logistic regression [6], eventually processing the dataset in order to fit its functional form - since the Logit model is built to take one vector as input, training on datasets of pairs of them requires some transformation, as will be thoroughly discussed in section 3.4.3.

Markov chains Monte Carlo methods

MCMC methods are a family of algorithms that perform sampling based on the posterior distribution of the model's parameters [10]. This means that through enough samples the model can be a good estimate of the quantity:

$$P(M|D,h) = \frac{P(D|M,h)P(M,h)}{\int_M P(D|M,h)P(M,h)dM} \tag{2.4}$$

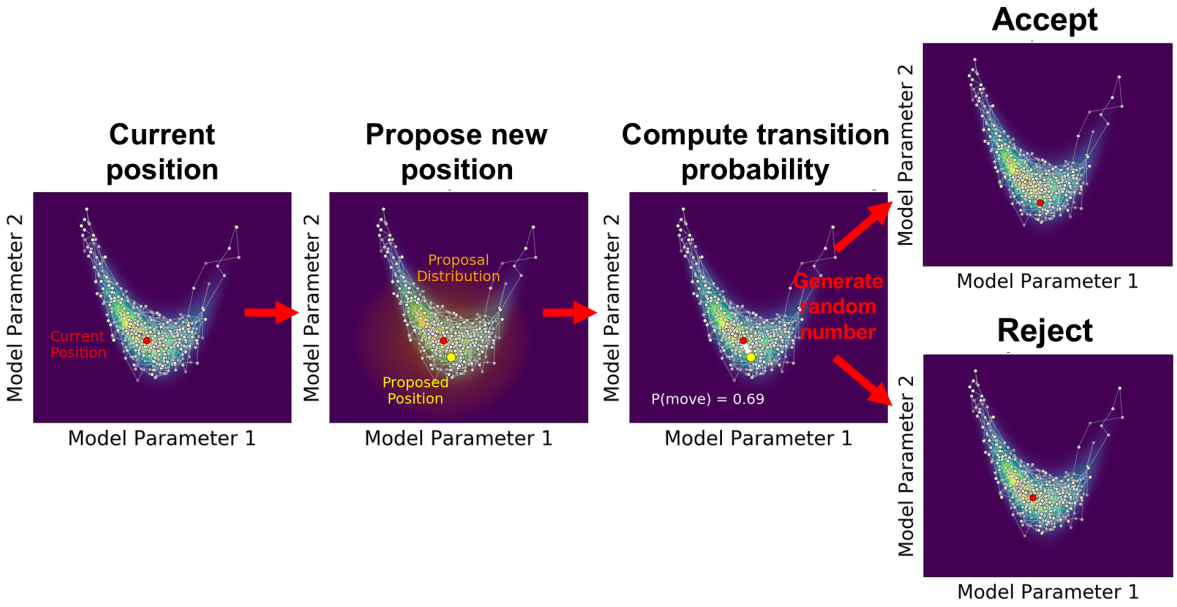

Figure 2.2: A visual representation of how each element of the chain is generated, taken from "A conceptual introduction to Markov chains Monte Carlo methods"[10]. The three steps of sampling strategy are depicted: initial proposal of the next sample through $Q(M_{i+1}|M_i)$, the computation of the transition probability $T(M_{i+1}|M_i)$ and the subsequent decision to accept or reject the proposed new sample.

where M stands for the model's parameters, h its hyperparameters and $P(M, h)$ is thus the prior on these quantities. The goal of this strategy is thus to estimate

$$D_s(M) \sim P(M|D, h) \tag{2.5}$$

where $D_s(M)$ is the density distribution of sampled values of M. In the specific case of interaction matrix estimation, $M \equiv W$ and $D \equiv \{Y, (X_n^{author}, X_n^{target})\}_{n=1,\dots,N_{edges}}$.
The sampling strategy in MCMC algorithms consists in generating chains of samples in an attempt to explore all the space of feasible parameters M, in a way such that after enough samples the distribution will converge according to 2.5.
These chains will consist in samples generated in sequence $\{M_n\}_{n=1,\dots,N_{samples}}$, extracted one after the other from a probability $P(M_{i+1}|M_+)$ that behaves like a propagator on model's parameters [10]. The process of generating these chains can be complex and the details about it are beyond the scope of our work; however, it can be useful to summarize the main steps of this process in order to understand how this model can be applied.
The transition from M_i to successive sampled values M_{i+1} is defined by a transition probability T, that decides whether to accept the sample, and a proposal distribution Q that defines a general rule of moving from a sample to the next:

$$P(M_{i+1}|M_i) \equiv T(M_{i+1}|M_I)Q(M_{i+1}|M_I) \tag{2.6}$$

This means that at the time i (after the i-st sample has been generated) the algorithm extracts the next sample from the proposal distribution Q (that can be, for example, a normal distribution centered on the current sample M_i); it then computes the

transition probability T and generate a random value $u_i \in [0, 1]$: the process is then repeated until the transition probability of this new sample is bigger than the random extracted number $T(M_{i+1}|M_i) \geq u_i$, ensuring that the chain will follow the criteria being requested by T.
The transition probability is defined as:

$$T(M_{i+1}|M_i) \equiv min\left[1, \frac{D(M_{i+1})}{D(M_i)} \frac{Q(M_i|M_{i+1})}{Q(M_{i+1}|M_i)}\right] \tag{2.7}$$

where $D(M_\tau)$ is the actual posterior being estimated by the algorithm itself, so the distribution of previous samples will progressively improve our convergence to the posterior. Figure 2.2 depicts a visual representation of this process. What is relevant for the purpose of application is that:

- The algorithm requires a reasonable choice of the proposal distribution Q in order to properly work, and different implementations of this process are actually possible changing this function.
- Contrary to other sampling based strategy, the advantage of MCMC family is that the samples are not bound to some finite region of parameters' space, so that the chains could theoretically explore it all.
- The model becomes "better at sampling" as times goes on, since the transition probability T actually depends on the posterior being estimated; this means that in use cases, is common practice to reject the initial samples, in what is generally called the "warmup phase".

Different variation of this general idea are then possible, by changing the way that the samples are generated, and the proposal distribution Q in particular. An optimization of the sampling process, providing an improvement on efficacy of the sampling, is the No-U-Turn Sampler (NUTS)[21]: the idea behind this implementation is to avoid the chains to "double back", exploring regions of parameters space "already explored enough", thus enhancing the information that each sample will carry.
Since the chains are generated through a propagator-like function, the goal of NUTS implementation is indeed to reduce autocorrelation within generated samples - namely the dependency of the sample M_{i+1} from the previous sample M_i. As a way to estimate the effectiveness of the sampling strategy, one can indeed compute the number of effective samples, being inversely proportional to the autocorrelation [10]:

$$N_{eff} = \frac{N_{sample}}{1 + \alpha} \tag{2.8}$$

where N_{sample} is the number of the required samples, and α a measure of autocorrelation in the chains. N_{eff} can thus quantify how the algorithm is effective in generating "informative" samples, and is a relevant quantity to look at when working with MCMC models. Since the NUTS algorithm is commonly regarded a well performing one among the MCMC family [21], this was the choice we made for our study.

After the training process, and after ensuring through N_{eff} that the generated samples are informative enough, the model finally gives the desired distribution $D(M)$

of the whole chains. Considering $N_{samples} = N_{warmup} + N$ one can then obtain a good estimate of the posterior by considering $D_s(M)$ the distribution of sampled parameters on the chains, removing the one of the warmup; a general rule of thumb is to divide the samples into two, using the same amount of samples for the warmup and for posterior estimate.
One can thus have:

$$D_s(M) \equiv \text{distribution of sampled parameters after the warmup} \sim P(M|D,h) \quad (2.9)$$

The relation to the actual posterior can then be evaluated looking at the convergence of distribution $D_s(M)$: if the distribution of sampled pairs has managed to converge, it means it has has converged to the actual posterior; if the distribution has not converged, then a bigger amount of samples is required. This convergence can be computed with Gelman-Rubin statistics [22], and the number of samples can subsequently be chosen in order to have significant convergency.
Once the distribution $D_s(M)$ is proven to be convergent, if the task requires choosing a single "good" parametrization, it can be assumed that the average of this distribution might be well representative for the model's parameters:

$$M_{best} \equiv < D_s(M) > = \frac{\sum_{s=1}^{N} M_s}{N} \quad (2.10)$$

where N is the number of samples after the warmup phase, and M_s the sampled parameters at iteration s. Looking at the posterior itself will tell if the average if actually a good choice (in the case that, for example, the posterior is denser on its average), otherwise different choices can be made in relation to the task.

The advantages of this approach is the great versatility in prior assumption and on the whole model's structure, together with providing useful information about the whole parameter posterior; one of the disadvantages is the complex structure of it, requiring sometimes calibration of algorithm's hyperparameters and some testing. The main drawback of this approach is however the great computational cost, scaling exponentially with increasing dimension of parameters' space [10]. This means that, thanks to its versatility, MCMC can be a good choice when no other training algorithm could be used for a given task; at the same time, better learning algorithms could be used in certain scenario, that might require much smaller computational costs without losing learning capability.

Logit model

In the use case of feature-feature interactions prediction, Logit model can be used after a transformation that maps the pair of vectors into a single one. This means that when applying the Logit model in a problem of pairs classification, one must choose a transformation of the kind:

$$T_x : (X_{author}, X_{target}) \to \tilde{X} \quad (2.11)$$

Examples of this transformation, both on our study and previous works are reported in section 3.4, while here I will only discuss how this approach is generally applied and its advantages. In order to estabilish statistical significance of the outcomes, the dataset can first be evaluated using Variance Inflation Factor [23]. As long as the features show no pathological collinearity, it is indeed possible to establish statistical significance of the estimated parameters.
The significance of each regressor's weights can indeed be evaluated in term of their p-values and the confidence intervals; for each pair of features, the reliability of the estimated weight will indeed depend not only on the actual contribution of that pair to the interaction, but also to the number of occurrences of that pair observed by the model. This is particularly relevant in the context of research since it allows to evaluate the reliability of the obtained results.
Furthermore, while the memory used to store the "expanded vector" scaled with the number of interactions in the dataset $\sim \frac{N^2}{2N} \sim N$ this method is proved to be fast in training, also allowing the use of different learning algorithms.
For being easy to use, fast, and able to provide statistical significance of results, the Logistic regression model is a common choice in current research; after different tests with either MCMC method and Logit model, it will be indeed our choice either.

Chapter 3

Methodology

Our approach followed two steps: extracting features from users' activity (presented in sections 3.1 and 3.2), then applying the learning algorithms to measure their interactions (presented in section 3.4); other strategies have also been later applied to further characterize the interactions, and extract more detailed information (namely text based methods presented in section 3.3). Since the task was of unsupervised learning, validation and interpretability of this methodology has been evaluated on each step on its own, as it is discussed in the next chapter.

3.1 Data Ingestion Pipeline

The features extraction pipeline starts from raw data of Reddit activity, and ends up with most active users' features and their interactions.
Chosen a "context" (year and a subreddit of interest), the program scans the whole Reddit data to find the "local" activity of the users, the proceed to select the most active and extract the required information.

All the featyres will be activity based, meaning it is assumed that the habits of the users will be somewhat representative of their opinion, social status, socio-demographic category. This assumption is commonly used in state of the art model [3] [6]: though the activity of each user can be specific, it is assumed therefore that a statistically relevant amount of them will lean to some obsrvable habits in their posting activity. How to observe these habits and link them to their features will be discussed in the next section.

Since Reddit is shaped in a tree-like structure, where each comment has a target the author is answering to, the interactions will instead just consist in a list of Author-Target-Sentiment of each comment in the chosen context. The sentiment is determined as described in 3.3.1, while the dataset is later limited to the interactions within selected users only.

To speed up this process, parallelization of each step has been implemented using pyspark library on python (version 3.4.7).

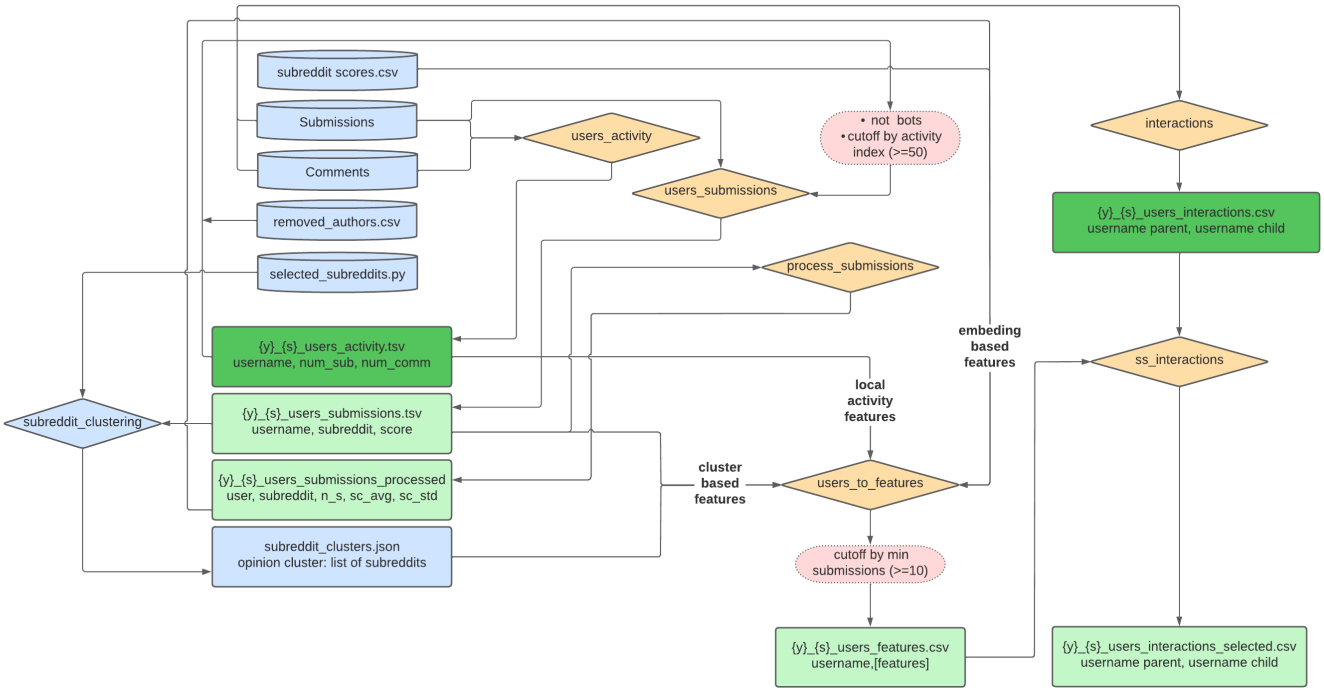

Figure 3.1: Pipeline of features extraction with mentioned different strategies. Color blue indicates that the same file/script is used for different datasets; color dark green stands for dataset dependent files; light green stands for datasets/cutoffs/preprocessing dependent files; red nodes are the cutoffs and diamond shaped ones are python scripts.
Squared shaped nodes represent files generated in the process: the first line is the name of the file (where {y}_{s} will take the values of year and subreddit of chosen context) while second line tells what data is stored in that file.

3.1.1 Users selection

Since Reddit provided a large amount of data in the subreddits of our choice -in the order of 10^7 comments in each year- we had to reduce our datasets to what we assumed were the most relevant interactions. During the pipeline different criteria of users selection are thus applied:

- Minimum local activity
- Minimum global activity
- User must not be a bot

The local activity is evaluated in the chosen context: the sum of submissions and comments of an author must be greater or equal than given cutoff C_{local}(50 and 25 will be used as cutoff's values); global activity cutoff requires instead that the author must have been active in at least a certain amount of subreddit: since the global activity is evaluated on users' submissions, this means that each selected user must have sent at

least one submission in at least C_{global} different subreddits (10 and 5 will be used as cutoff's values).
The purpose of these cutoffs is mainly to reduce noise in the dataset, since it is assumed that habits of most active users will have some most observable leaning. Furthermore, this process reduces computational costs of the whole pipeline, either in term of memory usage and required time.

To find bots we instead applied two criteria: we considered to be a bot a user for which one of the following is true:

- Author has made submissions in chosed context but no comment
- Author belongs to a list of known reddit provided bots (automoderators etc) [24]
- Author who have posted in more than 50 subreddits each month
- Author that have the string 'bot' in their name

State of the art strategy of bots detection might apply finer criteria, based on persistence of activity and repetitiveness of text and lexicon [24], while the solution we used might be considered rougher. However, since Reddit already provides a big amount of data and since it can be assumed that these criteria induce no bias in the dataset, we decided they were good enough.

After the extraction of all users activity and interactions, the values C_{global}, C_{local} have been chosen in order to have a minimum of around 10^6 interactions to train the model on. This amount ensures that even the rarest features' will be significantly represented in the dataset (these concerns will be later addressed in section 4.1).
Since we determined that the amount of total activity and users' comments distribution is context specific, different cutoffs have been applied; to ensure some consistency, the same cutoffs are applied to datasets extracted from the same subreddit even in different years (table 3.1). This data cleaning step resulted in the selection of around $5\% - 20\%$ of the total interactions, while looking only at around $2\% - 5\%$ of the users.

	N all nodes	C_{local}	C_{global}	**N nodes**	N all edges	**N edges**
2016 politics	491.903	50	10	**17.336**	16.432.846	**1.998.604**
2016 news	638.811	25	5	**27.976**	5.795.873	**1.166.076**
2017 politics	537.044	50	10	**16.559**	17.913.072	**1.689.337**
2017 news	727.100	25	5	**34.060**	7.452.411	**1.390.243**
2018 politics	626.316	50	10	**17.274**	19.515.785	**1.621.852**
2018 news	725.953	25	5	**31.997**	7.071.222	**1.221.779**
2019 politics	801.539	50	10	**18.626**	15.800.076	**1.103.759**
2019 news	685.542	25	5	**21.225**	5.922.030	**793.569**
2020 politics	1.267.545	50	10	**26.772**	25.360.849	**1.945.175**
2020 news	869.943	25	5	**29.045**	7.741.996	**1.067.614**

Table 3.1: Datasets statistics before and after applying cutoff to users' activity. *N all nodes* is the total number of users observed in that year and subreddit, while *N all edges* is the total number of interactions (comments). *N nodes* and *N edges* are the remaining users and interactions after C_{local} and C_{global} cutoffs are applied to their activity. Training will be performed on those, sub sampling up to a maximum of 10^6 edges.

3.2 Features extraction

As discussed in the next section, the model structure allow for the use of either categorical, binary and continuous features. The solution we adopted is to transform continuous variables in categorical ones, using therefore only binary features some of which will be mutually exclusive (like male or female, the leaning to left or right in political opinion and so on). This allows for more readable results and the possibility of observing non assortative behaviors in continuous features.
We used features of three kind:

- Socio-demographic status (age, gender, affluence)
- Opinion leaning (leaning left or right, and the "membership" of a certain politically oriented community like lgbt, gun owners etc.)
- Local activity and popularity

And we extracted them using three different strategies:

- Value on an axis inferred by graph embedding techniques (for the sociodemograpic status and left/right leaning); specifically each community (aka subreddit) is assigned a value on certain axis, and users' features are evaluated on the base of how much they were involved with each of them.
- Membership of a given subreddit cluster (for membership of politically oriented communities); this meant we found group of community who seemingly shared the same specific opinion, and we asked whether each user was involved in one of them.
- Direct observation (for local activity and popularity).

The direct observations gave us activity in given context (year and subreddit that the dataset was built upon) and the received upvotes, which can be used as metrics on their involvement in that community and popularity.

The other activity based features will instead be evaluated on the submissions of the users in other subreddits; it is therefore assumed that the act of writing a submission in a given subreddit in a stronger indicator than just answering to other authors. While this might be an approximation, since submissions are much less frequent than comments in Reddit, this solution is faster and might be assumed to be a way to remove noise in estimating users' features.

3.2.1 Features from embedding

These features are obtained on the base of a recent work about graph embedding and "community scoring" [3]; summarizing, each subreddit is assigned a score based on an ambedded representation computed on all Reddit users' activity, which places them on a polar axis representing some polar feature. We therefore used subreddits' score provided from their work to evaluate age, gender, affluence and left/right leaning of each user; these scores' distributions are represented in fig.3.2

These users' features are then computed as the weighted average of all their submissions in each subreddit:

$$F = \frac{N_s F_s}{\sum_s N_s} \tag{3.1}$$

Where s is a subreddit, F the feature, N_s the number of the user's submissions in that subreddit, and F_s is the score of the subreddit. This strategy makes so that if a subreddit is leaning to "male" in the gender axis (which can be interpreted as being mostly frequented by males [3]), this scoring method will assign a bigger probability of being male to a user writing lots of submissions in that subreddit.

At this point the users will have a set of continuous features of values ranging in $[-1, 1]$, representing the leaning on each axis/feature. To use these features in our model, we quantile-normalized these values among the selected users and chose to label only the ones falling in at the two extremes of the distribution. Choosing a threshold of 0.25, this means that taking for example the gender axis we will label the lowest quarter to be male, and highest quarter to be female.
The feature vector X_i of $i-st$ user, will thus be composed of a component of embedding based features:

$$X_i^{emb} = \bar{x}_i^j \qquad j = 1, ..., N_{emb} \tag{3.2}$$

where for each of the N_{emb} features, provided f_i^j quantile normalized $j-st$ feature of $i-st$ author, we obtained:

$$x_i^j = \begin{cases} [1,0] & f_i^j \leq 0.25 \\ [0,1] & f_i^j \geq 0.75 \\ [0,0] & otherwise \end{cases} \tag{3.3}$$

This decision induces balance in these features, while that might be not the case in the dataset. Nonetheless, we observed that the balance of the dataset is not relevant

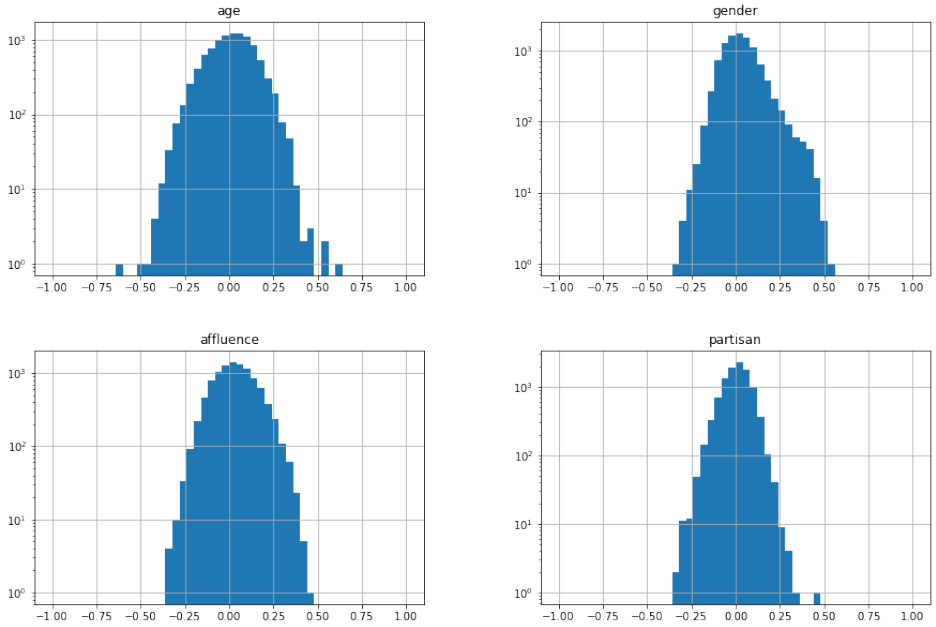

Figure 3.2: Subreddit scores distribution along polar axis with different meaning, provided by data of [3]. On x axis are the possible values of leaning (ranging in $[-1, 1]$ divided in 100 bins), on y axis the count of how many subreddits fall in that bin represented in logaritmic scale

in features matrix interaction estimation (section 4.2.3), and thus can provide a stable feature labelling strategy. The interpretation of the values on the polar embedding based axis is indeed not obvious; while a linear interpretation of the sign of these value can be considered to be an indication of the leaning (so that a negative value on that axis mean female while a positive value means male), this approach should be verified and not taken for granted. The interpretation approach followed by [3] in their work is instead looking at these values after mapping them to the distance from the average of all of them. Only then, the sign of these distances have been considered to be interpretable as the actual leaning.
In a similar manner therefore, we only looked at the quantile-normalized values and assumed the only the fact of being at the extremes can be interpreted as the actual leaning; the validity and interpretability of these features will be further addressed in cap 4.1

3.2.2 Features from clustering

As mentioned above, another strategy to extract activity based features is just looking at the participation of a user in some specific(group of) subreddit(s). While some communities' purpose is to be neutral and provide a ground for different opinion users to discuss, some others are made to be community of people with a specific shared opinion; in other cases, furthermore, some specific theme/moderation structure of a subreddit might provide favorable ground only for people who share some sort of opinion and/or value.
Since some of the communities can thus be considered polarized, we introduced clus-

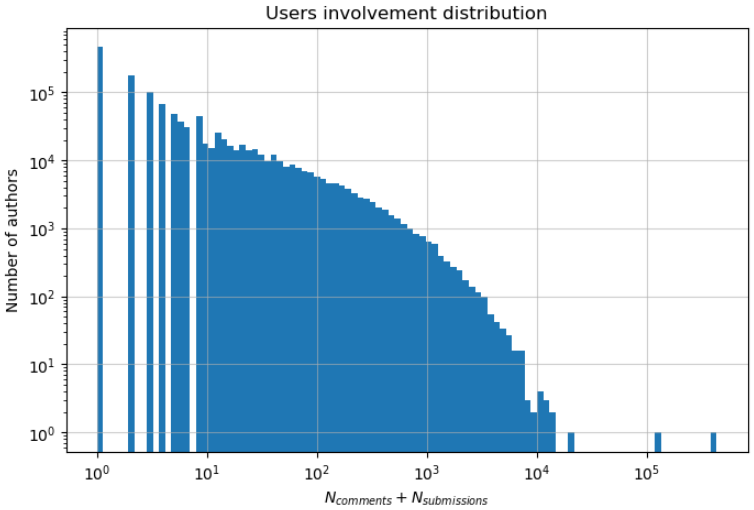

Figure 3.3: Users' involvement distribution in 2020 on subreddit 'politics'. Involvement is the total number of a user activity in that context (sum of number of its submissions and comments), and y-axis in this histograp represent the amount users with that involvement value.

tering based features in order to find groups of subreddit who supposedly share some same beliefs.

In order to do so, we first chose a set of subreddit which, by manual inspection, we thought to be insteresting from a political point of view; then we performed clustering on them, representing each subreddit with a vector of the users active in them in 2020[1].

The metric used to evaluate distance between these subreddits' vectors is Jaccard similarity, and we used DBSCAN clustering with $\epsilon = 0.94$; this strategy proved to be useful in finding communities who share some specific interest, as the clusters that emerged could be intuitively labelled as left extremist, left moderates, conservatives, guns enthusiasts, lgbt supporters and environmentalists.

The membership of each user in each community is then evaluated looking at their submissions: if a user had made at least one popular submission in a community (which received more upvotes than downvotes), it is assumed to belong in that community and therefore share their opinion.

The feature vector of $i-st$ user were thus be composed of another component of cluster based features:

$$X_i^{clust} = \bar{x}_i^k \qquad k = 1, ..., N_{clust} \tag{3.4}$$

where for each of the N_{clust} features, we obtained:

$$x_i^k = \begin{cases} 1 & if \quad i \in k \\ 0 & otherwise \end{cases} \tag{3.5}$$

so that the value of element x_i^k stands for user i belonging/sharing opinion of community k.

[1]Since the clustering is performed on 2020 activity only, it is to be assumed that the general opinion-activity tendency of Reddit users must have been mostly invariate in year of analysis (2016-2020 in our case)

mod_left	JoeBiden, neoliberal, democrats, Enough_Sanders_Spam, Pete_Buttigieg, ElizabethWarren, VoteBlue, VoteDEM, Liberal
ext_left	SandersForPresident, ChapoTrapHouse, LateStageCapitalism, WayOfTheBern, OurPresident, ToiletPaperUSA, Political_Revolution, BernieSanders, ENLIGHTENEDCENTRISM, DemocraticSocialism, socialism, DankLeft, AntifascistsofReddit, SocialistRA, ShitLiberalsSay, MarchAgainstNazis, Anarchism, bernieblindness, COMPLETEANARCHY, Socialism_101, communism, alltheleft, IronFrontUSA, Anarchy101
anti_trump	Fuckthealtright, EnoughTrumpSpam, The_Mueller, esist, MarchAgainstTrump, RussiaLago, Impeach_Trump
conservatives	Conservative
pro_gun	liberalgunowners, guns, Firearms, gundeals, GunAccessoriesForSale
lgbt_comm	asktransgender, traaaaaaannnnnnnnnns, actuallesbians, trans, transgendercirclejerk
environment	environment, climate, energy, climatechange, ClimateOffensive

Table 3.2: Clusters of opinion related subreddit, found with DBSCAN; each entry here represent the feature we associated with it, and the subreddit in the each cluster. Each of this cluster can thus be considered a community, possibly distributed in different subreddits, where the members share some sort of opinion

3.2.3 Local activity and popularity

Feature of user's involvement was given by the integer value of total number of comments and submissions in the context; feature of user's popularity was instead it's average received score, where each score is given by total upvotes minus total downvote.
Popularity has then been quantile normalized and divided in bin like we did on eq.3.3[2]. Values of activity required instead normalization by edges: since most active users were responsible of the greatest percentage of interactions, picking one comment would almost often result in picking an interaction between two users falling in the top percentage of most active. This resulted in great imbalance between observed interactions involving less active users, but was indeed resolved by weighting the quantile normalization over the number of comments.

[2]Apart from results presented in section 4.3, where we used different binning in order to have more granular information about local popularity

At the end of the process, we thus had the three different kind of feature, each one represented by a vector of binary values. The final feature vector associated to each user was then in a form of:

$$X_i = [X_i^{emb}, X_i^{loc}, X_i^{clust}] = X_i^h \tag{3.6}$$

where each element X_i^h could only take the value of $\{0, 1\}$ representing the activation of that feature for user i.

3.3 Text based analysis

To gather more details about interactions, we applied method of text analysis. Specifically, sentiment analysis was used as a metric of interactions' "quality"; topic recognition was instead performed on submission titles in order to evaluate any correlation between users' features and specific topic involvement.

3.3.1 Sentiment analysis

Using default implementation of VADER [25] (Valence Aware Dictionary and sEntiment Reasoner) we evaluated each comment's sentiment, as a metric of whether the interaction between two features is "positive" or "negative".
VADER is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media; it works on a pretrained single-word to sentiment mapping, so that the sentiment of a sentence can be evaluated as the average of the single words it is composed of.
For each interaction we then looked at author's and target's features and added sentiment value, thus mapping each feature pair to an array of measured sentiments. For each pair, overall sentiment is finally evaluated as the average of this array.

As will be discussed in 5.4, values estimated in this manner were highly context dependent and noisy; nonetheless, comparing observed average sentiment of a pair with the others can be a metric of how much users in that category are arguing or conversing. This method also provided insights about users' relation between socio demographic/opinion category and expected language, since average sentiment systematically appeard to be higher or lower for some feature.

3.3.2 Topic recognition

We also wanted to measure how much users' socio demographic classes and opinions influenced the choices of topics they got involved with. In order to do so, we used topic recognition strategy Using gensim and Spacy library.
First we looked for the topics in the context (given year and subreddit) submissions: we took the titles, lemmatized them and removed the stopwords, then filtered the corpus by keeping only the tokens which appeared at least 10 times in the dataset. Furthermore, to clean up data, we only kept unique titles (avoiding the ones that might have been due to spam/bots).

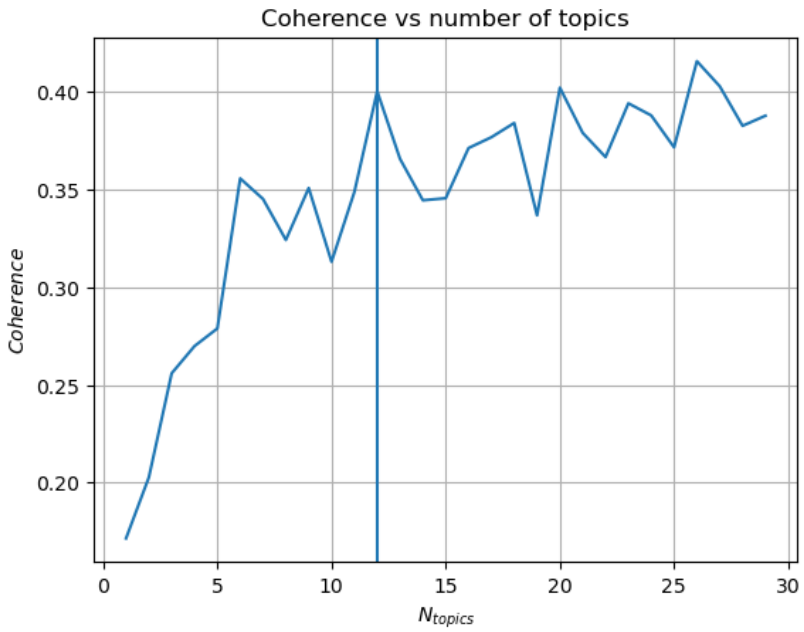

Figure 3.4: Coherence of submissions' title representation, with respect to number of topics in NMF. Value of 12 was chosen by visual inspection of this graph, searching for a value where curve abruptly slows its growth, but still provide a good $Coherence/N_{topic}$ ratio

Topics were then found applying Negative Matrix Factorization (NMF) with a proper number of given topics N_{topics}; this proper number was estabilished looking at coherence vs number of topic relation, choosing a number which provided enough coherence and a good ratio of the two. Looking at fig.3.4, a good choice of N_{topic} was found to be around 10-15 topics.

With these topics, we were able to associate one or more topic to each interaction: given the initial submission, NMF performed on its title gave us the topic of discussion, and it is assumed that every other subsequent comment in that submission will be coherent with that.

Relating each comment to a set of topics, gave us even more granular information about interactions: the question that we want to answer in this manner is "how does the topic influence interactions between certain users' categories? Specifically, for any pair of author-target features how do interactions probability/sentiment change when talking about different topics?".

This strategy was implemented in later stages of our work, in order to have a better understanding of the constant relations we found in different features' interactions (results reported in sec.5.2); in particular we were interested to see whether consistent assortativity, with respect to affluence, were due to the topic of discussion or the features themselves.

At the time of writing, this stage of our study is still a work in progress, so results of this strategy will not be discussed in this thesis.

3.4 Model

The model will estimate probability of interactions between two users, based on the features extracted as explained above. Naming X_{in}, X_{out} the features vectors of the target and author of a comment, given a model M with hyperparameters h the probability associated with the outcome Y of their interaction is defined as

$$P(Y|X_{in}, X_{out}, M, h) \equiv P_{M,h}(Y|X_{in}, X_{out}) \tag{3.7}$$

The particular functional form of $P(Y|...)$ depends on the model of choice. The hyperparameters h involved this probability (posterior) will be a subset of all the system hyperparameters H, since some of the hyperparameters are involved in training and model optimization (es prior on M) but not in the posterior $P(Y|...)$ once the model has been established.

When the outcome is binary $Y \in \{0, 1\}$ and the model aim to predict linking between two nodes, we can define

$$P_{M,h}(Y = 1|X_{in}, X_{out}) = \text{link being established between } X_{in} \text{ and } X_{out} \tag{3.8}$$

Our goal is to predict whether a pair of features is more likely to lead to an interaction; for this purpose, the "observed links" will be labelled with $Y = 1$ while the non observed ones will be generated through negative sampling and labelled with $Y = 0$.
Negative sampling will be performed generating an amount $N_{neg} = PN_{pos}$ interactions, extracting them from the distribution of observed outdegree of the nodes; P is the negative sampling proportion, which we later observed does not influence the extimated interaction matrix 4.2.3.
This strategy thus consists in comparing the observed interactions' structure against a random graph with same degree distributions on the nodes, independently of their features.

3.4.1 Structure

In our model, we have defined that:

- $X \equiv \{x_i\}_{i=1}^{i=N} \qquad tc \qquad x_i \in [0, 1]$
- $M \equiv W_0, W$
- $h \equiv k$

Where:

- X is the feature vector of a node, representing the -eventually normalized- attributes of the node
- W is a real valued $N \times N$ matrix, which will have the meaning of "probability weights"
- $W_0 \in R$ will represent the bias of the model

- $k \in R$ will be the steepness of the sigmoid we are using

Given those, the probability of a link being established from *out* to *in* is defined as:

$$P(Y = 1|X_{in}, X_{out}) \equiv P_{W_0,W,k}(Y = 1|X_{in}, X_{out}) = \sigma_{W_0,k}(X_{out}^T W X_{in}) \tag{3.9}$$

Where $\sigma_{s,k} : R \to [0, 1]$ is the sigmoid

$$\sigma_{s,k}(x) = \frac{1}{1 - e^{-k(s+x)}} \tag{3.10}$$

Using this expression, we have that

$$P(Y = 1|X_{in}, X_{out}) = [1 - e^{-k(W_0 + X_{out}^T W X_{in})}]^{-1} \tag{3.11}$$

Interactions' odds The odds Θ of the link being established from node *out* to *in* are defined as

$$\Theta(X_{in}, X_{out}) = \frac{P(Y = 1|X_{in}, X_{out})}{1 - P(Y = 1|X_{in}, X_{out})} \tag{3.12}$$

By replacing expression above, using $\xi = k(W_0 + X_{out}^T W X_{in})$ to ease up notation:

$$\Theta(X_{in}, X_{out}) = \frac{(1 - e^{-\xi})^{-1}}{1 - (1 - e^{-\xi})^{-1}} = \frac{1}{(1 - e^{-\xi})(1 - \frac{1}{1-e^{-\xi}})} = \frac{1}{(1 - e^{-\xi})(\frac{1-e^{-\xi}-1}{1-e^{-\xi}})} = \frac{1}{e^{-\xi}} = e^{\xi} \tag{3.13}$$

So

$$\Theta(X_{in}, X_{out}) = e^{k(W_0 + X_{out}^T W X_{in})} \tag{3.14}$$

The log odds, will then be:

$$ln(\Theta(X_{in}, X_{out})) = k(W_0 + X_{out}^T W X_{in}) \tag{3.15}$$

This means that the values of the interaction matrix W that our model will estimate, can be interpreted as the log odds of the interactions between any pair of features

3.4.2 Logit Model

A solution to extimate probability of interaction, is to transform the two vectors X_{out}, X_{in} in a single vector $\tilde{X}$; this is a commonly used technique for models that try to estimate the interaction between two vectors, since it allow for the use of more common and known methods such as the Logistic Regression [6] [11].
While different choiches of the transformation $Tr : (X_{out}, X_{in}) \to \tilde{X}$ can be used (more about this in the next section), a good solution in our case in to use a Logit model where the regressor's weights $\tilde{W}$ can be mapped back to the interaction matrix W.
In order to do so, we chose the transformation:

$$T_x : \tilde{X}^{n=Ni+j} = X_{in}^j X_{out}^i \tag{3.16}$$

Where the apex n corresponds to the position in that vector, and N is the number of features. T_x is invertible considering that:

$$i = floor(n/N) \qquad and \qquad j = mod(n/N) \tag{3.17}$$

so that given the regressor's weights $\tilde{W}$, the interaction matrix can be obtained as

$$W_{i,j} = \tilde{W}_{Ni+j} \tag{3.18}$$

Looking then at the problem of binary classification with logistic regression model, it is defined that

- $\tilde{X} \equiv \{\tilde{x}_i\}_{i=1}^{i=\tilde{N}} \qquad tc \qquad \tilde{x}_i \in [0,1]$
- $\tilde{M} \equiv \tilde{W}_0, \tilde{W}$
- $\tilde{h} \equiv \tilde{k}$

Where:

- $\tilde{X}$ is the feature vector of a node, representing the -eventually normalized- attributes of the node
- $\tilde{W}$ is a real valued vector of lenght $\tilde{N}$
- $\tilde{W}_0 \in R$ will represent the bias of the model
- $\tilde{k} \in R$ will be the steepness of the sigmoid we are using

Given these, the probability of $\tilde{X}$ being assigned to category $Y = 1$ is defined as

$$P_{LR}(Y = 1|\tilde{X}) \equiv P_{\tilde{W}_0,\tilde{W},\tilde{k}}(Y = 1|\tilde{X}) = \sigma_{\tilde{W}_0,\tilde{k}}(\tilde{W}\tilde{X}) \tag{3.19}$$

Where $\sigma_{s,k} : R \to [0,1]$ is the sigmoid

$$\sigma_{s,k}(x) = \frac{1}{1 - e^{-k(s+x)}} \tag{3.20}$$

Using this expression, we have that

$$P_{LR}(Y = 1|\tilde{X}) = [1 - e^{-\tilde{k}(\tilde{W}_0 + \tilde{W}\tilde{X})}]^{-1} \tag{3.21}$$

And the Logit log odds

$$ln(\Theta_{LR}(\tilde{X})) = \tilde{k}(\tilde{W}_0 + \tilde{W}\tilde{X}) \tag{3.22}$$

Together with equation 3.18 this shows that the problem of interaction probability estimation between X_{out}, X_{in} can be solved with a logistic regression; moreover, provided the same sigmoid steepness $\tilde{k} = k$ and bias $\tilde{W}_0 = W_0$, the regressor weight will be the same (and have the same meaning of log odds) as the general model discussed in the above section.

Provided this statistical structure, any model that assigns probability according to eq.3.9 will differ only by hyperparameters and learning algorithm, and different solutions can be applied to extimate the interaction matrix W. We thus decided to test two different strategy: Markov Chains Monte Carlo (MCMC) and Logistic Regression. The latter will be defined as it has been described in this section, while the former is a sample based method, which can be used to extimate not only the weights but also their posterior distributions.

3.4.3 Comparison with other models

Given a model that assign probability like 3.9 or 3.19, different models can be built in order to do so. It is interesting to compare these models from a general point of view, since this allows to compare different results obtained from different methods and can estabilish coherence between those; this has indeed been useful when we compared our model to state of the art results, obtained on the same dataset but through different methods.

Considering two vectors X_{out}, X_{in} and a model that assign the probability like eq.3.9, this can be in general transformed to a model that assign probability then like 3.19 through a transformation $T : (T_x, T_m, T_h)$ such that:

- $T_x : X_{in}, X_{out} \to \tilde{X}$
- $T_m : M \to \tilde{M}$
- $T_h : h \to \tilde{h}$

The extimated probability can then be computed as

$$P_{T_m(M),T_h(h)}(Y|T_x(X_{in}, X_{out})) \tag{3.23}$$

and an equivalence criteria between the two model can be estabilished if for any X_{in}, X_{out}, there can be a transformation such that:

$$P_{T_m(M),T_h(h)}(Y|T_x(X_{in}, X_{out})) = P_{M,h}(Y|X_{in}, X_{out}) \tag{3.24}$$

It is not trivial for this relation to hold, since the transformation might reduce dimensionality of input space and/or the two models might have been defined with too different of a parametrization/hyperparametrization. It is not my goal to discuss this topic in detail but I will focus on linear transformation that reduce the pair X_{out}, X_{in} to a single vector that will be used for training in a regression model; as discussed above, this is a solution often adopted in state of the art link prediction task [11] and is nonetheless a general case for the solution we adopted for the logistic regression.

This means choosing a transformation of the input vectors such that:

- $T_x^n(X_{in}, X_{out}) = \tilde{X}_n$
- $T_m(W, W_0) = \tilde{W}, \tilde{W}_0$
- $T_h(k) = \tilde{k}$

where the steepness of the sigmoid $k, \tilde{k}$ is the only hyperparameter (and we can assume that $k = \tilde{k}$ for simplicity), while the biases $W_0, \tilde{W}_0$ will be extimated by the models themselves.
The transformation of the vectors through T_x can thus be decomposed in n components, where n is the number of features in the transformed vector and each component has its own transformation T_x^n. Requiring it to be linear means requiring

$$T_x^n(X_{in}, X_{out}) = \tilde{X}^n = \sum_{i=1}^{i=N} \sum_{j=1}^{j=N} t_{i,j}^n X_{in}^j X_{out}^i \tag{3.25}$$

where $t^n_{i,j}$ is a $N \times N$ real valued matrix.
In this framework, the transofmation provided in 3.4.3 can be indeed written as

$$T^n_x = \{t^n_{i,j} | 1 \text{ if } i = floor(n/N) \text{ and } j = mod(n/N); 0 \text{ otherwise } \} \qquad \text{for } n = Ni+j \in [1, N^2] \tag{3.26}$$

so that:

$$t^n_{i,j} = \delta_{n,Ni+j} \qquad \text{for } n \in [1, N^2] \tag{3.27}$$

A different application of transformation 3.4.3 will be used in section 4.3, to compare our model against results obtained on state of the art works, where a non trivial transformation of the two vectors is applied.

In the broader case we can use general linear case transformation 3.4.3 to make some comparison between the models' odds. In particular, it can be shown that if the equivalence relation 3.24 holds, then the log odds will also be the same and this can provide a way to establish a mapping T_m, T_h that satisfy this equivalence. The use case of this criteria is that, once a transformation of vectors T_x is established, one can find the transformation between the two models' parameters and compare their results; this leads to the possibility of comparing the results of two different models trained separately on the same dataset, not only through comparing the probabilities they generate but also their parameters (as we show in section 4.3). Furthermore, it justified the transformation 3.4.3 we used to train the logistic regression.

If the equivalence relation 3.24 holds, then since the odds are defined as a monotone function of the probability of an event $\Theta = \frac{P}{1-P}$ it must also hold for the odds and log odds. Replacing the log odds equations 3.15 and 3.22, the equivalence relation 3.24 becomes:

$$\tilde{k}(\tilde{W}_0 + \tilde{W}\tilde{X}) = k(W_0 + X^T_{out} W X_{in}) \tag{3.28}$$

Writing vector-matrix product explicitly and using definition 3.4.3, this becomes

$$\tilde{k}(\tilde{W}_0 + \sum_{n=1}^{n=\tilde{N}} \tilde{W}_n \sum_{i=1}^{i=N} \sum_{j=1}^{j=N} t^n_{i,j} X^j_{in} X^i_{out}) = k(W_0 + \sum_{i=1}^{i=N} \sum_{j=1}^{j=N} W_{i,j} X^j_{in} X^i_{out}) \tag{3.29}$$

by switching summation order:

$$\tilde{k}(\tilde{W}_0 + \sum_{i=1}^{i=N} \sum_{j=1}^{j=N} \sum_{n=1}^{n=\tilde{N}} \tilde{W}_n t^n_{i,j} X^j_{in} X^i_{out}) = k(W_0 + \sum_{i=1}^{i=N} \sum_{j=1}^{j=N} W_{i,j} X^j_{in} X^i_{out}) \tag{3.30}$$

Since this relation must hold for every pair $X_{in} X_{out}$, this relation must indepentently hold for every component i, j of the feature matrix W

$$T^{-1}_m : \frac{\tilde{k}}{k}(\sum_{n=1}^{n=\tilde{N}} \tilde{W}_n t^n_{i,j} + \tilde{W}_0) - W_0 = W_{i,j} \tag{3.31}$$

If we assume $k = \tilde{k}$ and $\tilde{W}_0 = W_0$ then this relation is just:

$$T^{-1}_m : \sum_{n=1}^{n=\tilde{N}} \tilde{W}_n t^n_{i,j} = W_{i,j} \tag{3.32}$$

which is indeed the general case of transformations 3.4.3 and 4.3.1.

Once the vector and hyperparameter transformation T_x and T_h is provided, relation 3.32 gives a way to map a model in one other, thus allowing for the use of different learning algorithm (as we did in section 4.2 using transformation 3.4.3) or comparison between two models (as we did in section 4.3 using transformation 4.3.1)

3.4.4 Learning algorithms

In this theoretical framework we will use two different learning algorithms, then evaluate their performance in terms of outcomes, computational costs, and the ability to provide measure of statistical significance.
The first algorithm is MCMC-NUTS algorithm [21], implemented from library pyro. We used a number of samples in the order $\sim 10^2$, half of which will be used in the "warmup phase". The model used in this case is straightforward and follows the probability definition given in eq.3.9.
The secon algorithm is the Logit model, implemented from library statsmodels. The model in this case follows eq. 3.19, thus is applied to our datasets only after transforming our vector pairs according to 3.4.3.
In either the case we tested the outcomes under different hyperparametrizations k, and after finding that these changes were not relevant to the outcomes we just set the sigmoid steepness $k = 1$. MCMC algorithm also requires a prior, according to 2.4, so we set $P(W), P(W_0) \sim Gauss(0, \sigma_W)$. Again, after different testing of hyperparametrization σ_w we just noticed no relevant changes in model's behaviour, and we just set $\sigma_w = 1$ meaning the MCMC prior on the parameters is just a standard normal distribution.
The extimation of the bias W_0 will instead be left to the model, and as we saw section 4.2.3 it will be related to the negative sampling proportion.

The first thing to notice now is that, according to what we discussed in previous section and equation 3.24, *we are using two equivalent models -meaning they assign probabilities in the same functional form- and so any difference in their outcome will be due only to the different learning processes.* While the equation used for the two models is basically the same, it is indeed not granted that the two different algorithms will lead to precisely the same W, W_0; furthermore, MCMC model will give us the posterior and no single parametrization of choice, so is up to us to decide the values W, W_0 from their posterior distributions.

We thus expected these two learning algorithms to ultimately give us the same interaction matrix estimation, though at different computational costs and different form.
The computational costs were different by several order of magnitude, the MCMC requiring days or even weeks to train while Logit model requiring just some hours. After some testing, as reported in the next section, we also saw that the average of the posterior estimated with MCMC method was systematically very close to Logit outcomes, meaning that *the posterior on parameters W, W_0 was denser around its peak value, so that its average actually coincided with the maximum value.*
This is an interesting result on itself, telling us that the posterior of the interaction

matrix was "pretty regular" and that the MCMC model was actually finding the same estimate of the Logit model; this result will be further discussed in section 4 where we performed training with either the strategies and compared their results.

However, given that no major advantage was provided by MCMC algorithm at the cost of mugh bigger time required for training, we later decided to continue our research on the Logit model alone. Since the Logit model allowed us to compute confidence intervals, so that the crucial part of the posterior could be evaluated anyway, this strategy was indeed not only faster but also equally effective.

Chapter 4

Validation

The goal of the methodology we developed was to estimate and characterize interactions in a given year/subreddit. The main difficulty in this task was that the raw data of Reddit provided no information about the users, other than their activity, so each observation is based on features we ourselves needed to extract.
For this reason it is important to distinguish the two steps of our approach as two distinguished tasks themselves: features extraction/opinion mining, and the study of how these features interacted. Learning their interactions is indeed tightly related to how these features are extracted, meaning any choice made on the first (and concerningly any bias) will be reflected on the second.
In this chapter I discuss how these concerns have been addressed in our work, and the tests we performed to evaluate the reliability of our approach.

While our task was not one of opinion mining itself, we indeed needed to take great care of the features extraction process, making sure that the dataset we were building was reliable enough. We thus constantly looked at datasets' statistics (as presented in section 5.1), performed inferential observations on those (in section 3.2), and compared results of features from previous work (in section 4.3) and different strategies (for example, trough table 5.3).
It is also important to notice that either clustering and embedding are based on activity itself, so extracting features based on those might be a bit of a convoluted process; we thus had to make a step back to see what the features actually represented and how could they be interpreted. A brief summary on this topic is presented in section 4.1, and the actual intepretability of any result had been addressed any time we made an observation (as will be the case in chapter 5).

In regard of the learning process, synthetic data test has been performed to see how the learning algorithms performed on different scenario, and will be presented in section 4.2. Furthermore, to evaluate our methodology we compared our strategy with recent state of the art work (section 4.3), to see wether we found coherence in our results and to discuss how our strategy can introduce some improvements.

4.1 Features validity and iterpretation

Since the goal of this work was to explore how/if it was possible to find interesting interactions between opinion/sociodemographic characteristics, the strategies used to actually extract those very features was of course of great relevance; furthermore, as discussed in section 3.1, while at the end each feature consisted in a binary representation of belonging/not belonging to a certain class of people, different means to extract these values had been used.
For this reason it has been important to look at the features themselves, before evaluating their interactions, and establish some criteria to evaluate their interpretation and their reliability.

A useful insight about features reliability is provided by basic statistics: for each subreddit it is interesting to ask what are the odds that taking a comment in that context, the author will belong to a certain class (namely, the corresponding element of his feature vector is 1). This means that given a certain feature, one can compute:

$$\Theta_s^F = \frac{P(\text{comment in } s \text{ by user belonging to } F)}{P(\text{comment in } s \text{ by user not belonging to } F)} = \frac{N_s^F}{N_s^{tot} - N_s^F} \tag{4.1}$$

for each subreddit s. Using this criteria, while looking only at the authors selected by our cutoffs - so the most active in the context of choice-, the subreddits with highest values of Θ_s^F will represent the activity leanings of the users labelled with F.

Though we did not formalize this procedure and it is not indeed possible to quantify reliability in this way alone, in the first stage of our work this told us whether the features correlated to the meaning we intuitively expected. In later stages, the reliability will be estimated by significance provided by the Logit model, while the interpretation -particularly for the embedding based features- can in some cases only be widely determined.
Nonetheless, observation with subreddit-feature odds told us that the features could be assumed to be reliable enough: as can be seen from tables 4.1 and 4.2 the users labelled with each feature appeared to be most active in subreddits where one could expect, based on common sense. This implied that the assumption we initially made was reasonable, and that the activity tendency of the users can effectively reflect some of their sociodemographic statuses and opinions.

young	ABCaus, techcrunch, TheAbditory, ElectionPolls, ShitPostCrusaders, okbuddyretard, leagueoflegends, teenagers, Animemes, Rainbow6, forhonor, PewdiepieSubmissions, RocketLeagueExchange, pokemon, boottoobig, smashbros, Gamingcirclejerk, comedyheaven, BikiniBottomTwitter, BattlefieldV
old	RuralNewsNetwork, MarsSociety, HotZone, CashApps, MAGAs, Against_Genocide, Derfla_bookmarks, V2X, itsTooLate, domesticgunviolence, Infrastructurist, EndlessWar, sciences, AmericanPolitics, climate, Health
male	NewLegislation, PrettyOlderWomen, newretrowave, CompetitiveWH40k, WomenWithWatches, olddominionfootball, AdultContemporary, V2X, domesticgunviolence, sexygirls, WarplanePorn, NATOrussianconflict, ScienceUncensored, irredeemables, MilitaryPorn, bikinis, scandinavia, WarshipPorn, Cyberpunk, GunsAreCool
female	HotZone, Buttigieg_Graphics, Derfla_bookmarks, itsTooLate, Fuck45, TransSpace, StormComing, LGBTnews, prochoice, Infrastructurist, popheads, TwoXChromosomes, ainbow, houseplants, climate, Republican, exmormon
poor	AsianGuysNSFW, domesticgunviolence, RuralNewsNetwork, LeftCentral, CelebrityAlbums, WhiteHouseHyperReal, CompetitiveWH40k, SFandFslavegirls, Trump_Impeachment, ClimateMemes, greatfilterpodcast, gayasianspeedo, BigTitsSmoothPits, dailywire, NATOrussianconflict, thomasjefferson, Snowflake_Meltdown, NewsofSeattle, ROI
rich	Derfla_bookmarks, V2X, Infrastructurist, sciences, thedonald, psychology, EverythingScience, Republican, Braves, technology, Health, energy, HomeImprovement, Sino, wisconsin, FoodPorn, business, travel, StLouis
left	domesticgunviolence, RuralNewsNetwork, LeftCentral, CelebrityAlbums, WhiteHouseHyperReal, SFandFslavegirls, CashApps, ElectionPolls, Against_Genocide, greatfilterpodcast, Derfla_bookmarks, itsTooLate, TransSpace, Fuck45, NATOrussianconflict, alltheleft, FakeProgressives
right	dailywire, CompetitiveWH40k, olddominionfootball, ScienceUncensored, irredeemables, RocketLeagueExchange, Conservative, Republican, leagueoflegends, LivestreamFail, CringeAnarchy, SargonofAkkad, metacanada, The_Donald, 2007scape, guns, progun, Warhammer40k, Ice_Poseidon2

Table 4.1: Subreddit with higest log odds (eq.4.1) of features extracted with embedding based strategy described in section 3.2.1

mod_left	RuralNewsNetwork, CelebrityAlbums, Buttigieg_Graphics, Enough_Sanders_Spam, NewsofSeattle, NATOrussianconflict, ImpeachmentWatch, thomasjefferson, Hatari, FakeProgressives, neoliberal, Pete_Buttigieg, VoteBlue, JoeBiden, alexjones, FoxFiction, Democrats2020, ElizabethWarren, Liberal, democrats
ext_left	N_N_N, LeftCentral, NATOrussianconflict, Kossacks_for_Sanders, SFandFslavegirls, MAGAs, CannabisNewsNetwork, ElectionPolls, greatfilterpodcast, itsTooLate, FakeProgressives, alltheleft, ProgressiveActivists, COMPLETEANARCHY, Anarchism, ChapoTrapHouse, WayOfTheBern, ROI
anti_trump	RuralNewsNetwork, LeftCentral, CelebrityAlbums, MAGAs, itsTooLate, domesticgunviolence, Fuck45, NATOrussianconflict, EnoughTrumpSpam, ImpeachmentWatch, thenewcoldwar, PrettyOlderWomen, newretrowave, Impeach_Trump, MarchAgainstTrump, Fuckthealtright, AntiTrumpAlliance
conservatives	MAGAs, dailywire, Conservative, SargonofAkkad, Republican, metacanada, NASCAR, gunpolitics, ShitPoliticsSays, The_Donald, borrow, USNEWS, YangForPresidentHQ, progun, AnythingGoesNews, lakers, vancouver, TheRightCantMeme, POLITIC, moderatepolitics
pro_gun	itsTooLate, ScienceUncensored, PrettyOlderWomen, Firearms, guns, Cyberpunk, gunpolitics, progun, The_DonaldUnleashed, Judaism, books, Libertarian, California, GreenBayPackers, brasil, 3Dprinting, libertarianmeme, bayarea, wholesomememes
lgbt_comm	CelebrityAlbums, TransSpace, traaaaaaannnnnnnnnns, LGBTnews, HistoryWhatIf, ainbow, dndmemes, forhonor, circlebroke2, TwoXChromosomes, science, houston, lgbt, VoteBlue, AgainstHateSubreddits, AskALiberal, atheism, psychology, fantasyfootball
environment	CashApps, MAGAs, ElectionPolls, Against_Genocide, V2X, ClimateOffensive, ImpeachmentWatch, TransSpace, StormComing, energy, climate, environment, ClimateMemes, TechNewsToday, sciences, KochWatch, psychology, Infrastructurist, altnewz

Table 4.2: Subreddit with higest log odds (eq.4.1) of features extracted with clustering strategy described in section 3.2.2

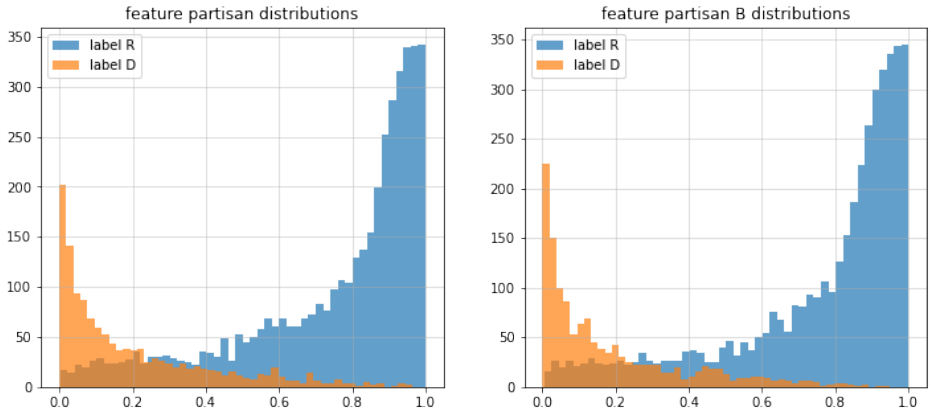

Figure 4.1: Distribution of users' political score through embedding scores, distinguishing the ones that were labelled as leaning toward left and right in [6]. As these graphs show, the two strategy are coherent and provide a good separation between left and right users; overlapping has been calculated to be over 90%. While partisan is the left/right leaning itself, partisan B represent the leaning toward left candidate (Clinton, labelled as D for democrat) and right candidate (Trump, labelled as R for repubblican); overlapping between partisan and partisan B has also been calculated to be over 90%). More details about these features in section 3.2.1 and in the original work where they are defined [3]

4.1.1 Embedding based features

Validity of the embedding based strategy discussed in 3.2.1 follows the validity of the community embedding discussed in [3]. It is important to notice that the embedding had been performed on the Reddit activity of 2012, so the provided subreddit scores represents the habits of users in that year; as discussed in their work, it is nonetheless reasonable to assume that these habits did not have significant changes in a not to long period of time, so that "the sociodemographic and opinion based habits of the users have mostly remained the same from 2012 up to 2020". To further estimate this validity we compared the Left/Right features attributed with this strategy and the Left/Right attribution with other strategies [6]. A way to extract activity based feature is indeed to find some subreddit whom orientation is known and can be taken for granted, then looking at authors active in these communities; an example of these is the subreddit "The Donald": there is a really high probability that a user of that community is indeed a Donald Trump supporter, since the subreddit is moderated in order to only allow them to post. This same strategy will indeed be the core concept to extract cluster-based features as discussed in the next section.
This comparison showed us an high overlapping between Left/Right features evaluated using this strategy and the features used in a previous work [6] (which we later specifically focus on, as will be discussed in section 4.3), further suggesting that this labelling strategy could be trusted to extract relevant information.

Regarding their interpretation, it is instead not possible to establish a strict meaning for each feature: the fact a user belongs to a certain class (as male/female, young/old and so on), must be interpreted as a higher leaning toward that class compared to the rest of the users. The classes "young/old" for example, do not establish a strict age range of that category, but a general indication that a given user leans to the same habits that are expected in that age - in other words, the user is most active in these

subreddits the embedding recognized as "young-ish" or "old-ish".

Nonetheless since the threshold make so that the "average" user is not labelled, this strategy, though not strictly interpretable, leads to a non noisy labelling according to reasonable expected behavior. This is indeed confirmed by the subreddit-features odds discussed above, which show reasonable tendency for all the embedding based feature we used.

4.1.2 Cluster based features

The cluster based features, differently from embedding based ones, are pretty much self defined; following the definition given in 3.2.2, the fact a user belongs to a class is to be interpreted as the fact he has participated in activity of a certain community.
These communities are given by the clustering algorithm, that ensures that the chosen subreddits are similar enough -in a Jaccard metric; the results reported in table 3.1 show that this method yielded reasonable clusters.
Chosing to filter out only popular users (the ones who made "popular submissions" based on the votes they received), is a way to reduce noise in the labelling process; if a user's submission resulted to be popular, it is indeed reasonable to assume he/she generally share the same opinion of the largest part of that community.

The community interpretation of these features makes sense when also compared with embedding based features, looking for example at the ones labelled as left vs moderate/extreme left and the ones labelled as right compared to conservatives: a user in one of these communities, display high probability to also be labelled at the expected political leaning. Namely users labelled as conservatives have a high probability of being labelled toward right partisanship, but not vice-versa: this means that the community of conservatives is -with good approximation- a subset of all users leaning to right, and the same will be true for users leaning toward left and the community of moderate and extremists (this can be deduced by corresponding rows and columns in tab.5.3).

One issue we encountered with this kind of features is instead about the amount of users we were able to label: among a total of around 10^4 selected users, we were able to find around 10^3 that could be labelled as "moderate left", "extremist left" and "conservatives", while only in the order 10^2 could be labelled as environmentalist, pro lgbt, pro gun etc (as can be seen in tables 5.1 and 5.2); this is coherent with the fact that a more specific opinion, such as being an environmentalist, will have smaller reach that a broader one, such as opinions generally leaning toward political left.
This has proven to be a numerical limit when it came to training, since a smaller order of users with a given feature leads to model's low statistical significance of that feature's interactions; for this reason we choose to perform training with at least 10^6 interactions, since that led to at least an order of 10^2 interactions for every pair of feature. This will be further discussed when presenting the results in [ref cap results].

Nonetheless, this strategy proved to yield really noisy results when we looked ad features' interactions: in chapter 5 it is showed how sometimes p-values can be really high, confidence intervals too wide or trend during the years too noisy. This might

be due to the fact that these features lead to no systematic interactions' tendency among the users, or possibly due to a too small dataset of these users; it might indeed be not reasonable to assume, for example, that the order of 10^2 users we found of environmentalists will be a representative enough sample for all the environmentalists in general.

For this reason, after some early results, (sections 5.1 and 5.2), in later stages of our work we decided to drop these features and focus only on embedding based ones, which proved to be more stable and reliable when it came to statistical significance. Despite we decided that they were of secondary importance for our purposes, relevant and significant information about these features' interactions can still be acquired through this strategy, maybe requiring different cutoffs to label bigger amount of users or maybe refining the clustering step defined in 3.2.2.

4.2 Test on synthetic data

In early stages of our work, the first thing we did was to test our model on synthetic data. In particular, we were interested in observing how the model behaved when changing:

- The dataset type we provided: binary, continuous and categorical variables. For "categorical" it is intended features which are mutually exclusive with other (for example, male vs female, old vs young and so on)
- Features overlapping: the amount of features that each user will have, might influence the ability of the model to estimate the interaction matric.
- The negative sampling proportion: since the ovserved interactions are only a small size in comparison to all possible ones, negative sampling is required and the proportion might influence model's behavior.
- MCMC model's priors on the interaction matrix values $P(W)$ and bias $P(W_0)$
- MCMC sigmoid's steepness k.
- MCMC convergence with respect to number of samples.

After some testing, we decided that the MCMC model's samples needed to be in the order of 10^2, which reliably provided good convergence of the interaction matrix's values. Changes in the sigmoid steepness and model's prior proved to be not so interesting so will not be further discussed; after some testing we decided to just chose $k = 1$ and $P(W_{i,j}) = P(W_0) = Gauss(0, 1)$.
The matter of overlapping is instead non trivial: if the model always trains on high overlapping features, its ability to find which one of them actually influence interactions might be compromised. This is a behavior we observed during testing and is generally a common issue one might encounter when training model which take binary vectors as input; it is also generally assumed that in the case of non pathological correlation (which can be verified measuring feature's collinearity) the model will be able to learn effective weights when enough data is provided. This means that by looking ad enough overlapping permutation, the model is able to learn a reasonable weight for each single feature independently. We did not formally studied this phenomenon but we established that "enough data" were required and decided to check variance inflation factor [23] before training.

More interesting finding are instead ones regarding data type and negative sample size, which will be discussed in the next sections.

4.2.1 Generating features

Different strategies were used to generate different data type, in order to observe model's behavior and decide which one would be most efficient. In each case we generated D pairs of independent vectors, built with some given rule of sampling; the dataset thus consisted on D interactions between these two features vectors, directed

from the first (target) to the second (author). This convention follows the fact that in a tree-like structure (as each subreddit is) the parent (tail) of an edge is the one receiving the comment, while the author is to be considered its child (head).

Fixed overlapping binary vectors First we generated binary features with fixed overlapping O, meaning we generated D pairs of vectors with N values in $0, 1$. Each vector was thus composed of permutations of O ones, making so that each permutation had the same probability $\frac{O!}{(N-O)!N!}$.
This first test allowed us to observe the "high overlapping" problem briefly discussed above: though we were not able or focused on establishing an exact relation, we can tell that the issue of "too high overlapping" can stem from the some ratio $\frac{O/N}{D} \sim$ too high, meaning it should always be possible to avoid it providing enough data D regardless of the overlapping; furthermore, real data does not have "fixed overlapping" O, and the possibility to observe low overlapping vectors' interactions vastly enhance model's ability to train effectively.

Variable overlapping binary vectors We then generated random binary vectors with variable overlapping: first the overlapping O_{user} is chosen sampling an integer values from a Poissonian distribution with given average, then a permutation of O_i ones out of the N values is sampled with probability $\frac{O_i!}{(N-O_{user})!N!}$, meaning once the overlapping is sampled each permutation will have the same probability.

Continuous variables To generate continuous-valued vectors, for each vector we just extracted a N values from a uniform distribution

$$V^i_{user} \sim Uniform(0, 1) \qquad i \in [1, N] \tag{4.2}$$

This decision well represent a situation of edge-wise normalized features, which is often the case when working with continuous variables. Though we ended up not using continuous features, it is interesting to notice that the model is technically able to train effectively either with continuous or binary variables, even when used at the same time

Binary-categorical vectors We then wanted to generate features, where different chunks of the vector represented the division in bins of some underlying feature. This is indeed a solution we used when training on, for example, the popularity of the users, where a chunk of three values in a vector represented low/average/high popularity; this means that the vector can be partitioned in some amount of smaller vectors, where only one of the values can be one and so the values are mutually exclusive.
This solution can be adopted either in cases where a bin division is required for a feature (i.e. low/mid/high or similar) or in cases of bipolar features like gender and political leaning toward left and right.

In order to study the model's behavior in these cases, we generated continuous variables according to eq.4.2, then used some binning-like rule to transform each continuous variable in its binary counterpart; this binning-like rule mapped different intervals

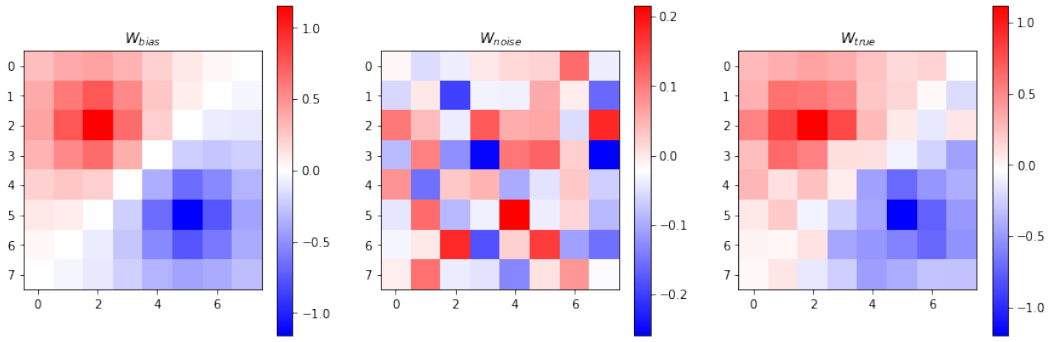

Figure 4.2: The process used to generate a "realistic" interaction matrix. W_{bias} is the matrix of the $B_{i,j}$ values introduced to simulate assortativity or disassortativity in the interactions; the noise is a matrix of i.i.d. values sampled from a Normal distribution with standard deviation of 0.1

within $[0, 1]$ to a n_N vector where at most one of the values was one. The strategies we tested were:

- Uniform binning: chosen a certain n_N for each of the N continuous variables, n_N equally sized intervals were mapped to a vector with one in corresponding position

$$for \quad i \in [1, n_N] \qquad M^N : \left[\frac{i-1}{n_N}, \frac{i}{n_N}\right] \rightarrow [\delta_{i,j}] \qquad for \quad j \in [1, n_N] \tag{4.3}$$

- Binning the extremes: chosen a threshold Tr, the values below trashold are mapped to $[1, 0]$, the values above to $[0, 1]$ and all the others to $[0, 0]$

4.2.2 Generating interactions

Given the D pairs of features vectors generated in through one of the sampling processes described above, we proceeded to simulate their interactions.
To do so, we generated a "true interaction matrix" W_{true} which is to be intended as the underlying features influence on interactions and will be the model's target. First, we generated W_{true} by just sampling each value from the same prior $P(W)$ we used in the MCMC model; this resulted in gaussian distributed random valued matrices where -by central limit theorem- most of the interactions occurred with a probability of around 0.5. While the model was effectively able to learn this matrices we introduced a "bias function" on the values of W_{true} to simulate a situation where some group of features eventually behaved assortatively or disassortatively; in other words, we introduced some structure in the interactions to simulate some more realistic scenario.

We thus built the matrices according to some law $B_{i,j}$ and added noise in the same functional form as the prior (still a gaussian, but with lower standard deviation to avoid unreasonably high or low values):

$$W_{true}^{i,j} = B_{i,j} + Gauss(0, 0.1) \tag{4.4}$$

In particular, in one of the instances we used $B_{i,j}$ central field like values, where a positive centers give higer probabilities that these features pair lead to interaction, and

a negative center simulate disassortativity. This roughly simulate a situation where some features lead to assortative behaviour, while some other lead to disassortativity. We did not explore in details many different possible compositions of W_{true} but we found this solution to be broadly representative of real case datasets. An instance of this realization is shown in fig.4.2.

It is important to notice the role of the "noise" here is to introduce some irregularity in the matrix, but the generating and training processes will make no distinction between what we called "bias" and "noise" and just learn W_{true} as it is. The role of the W_{noise} addition in thus to simulate the irregularity that can be found in real data themself, so that the target we expect to find through learning is precisely the matrix W_{true}

At this point we sampled interactions between an amount in the order of 10^6 generated paird D: each of the pair is assigned a probability by eq.3.9, then for each one of them a value of $interaction = 0, 1$ is sampled with a binomial distribution averaged on that probability. We thus have a dataset consisting of positive interactions labelled with 1, which stands to represent comments between two nodes with features in D. Subsampling is then performed on the positive interactions, and a proportion $P_{neg/pos}$ of negative ones (labelled with interaction value of 0) is taken to simulate negative sampling.

The training was then performed with an amount of $10^3 - 10^4$ positive interactions for MCMC, and $10^3 - 10^6$ for Logit model; the limitation on dataset size used for MCMC is due to the prohibitively long times it took for training. The proportion of negative sampling $P_{neg/pos}$ we used ranged from $0.5 - 10$ and will be object of further discussion in the next section

4.2.3 Observations

Our goal at this point is to observe wether the models were able to learn the interaction matrix W_{true}, and study what consistency we could expect when training on real data. The results were really positive since either the model appeared to be able to effectively learn with low to no approximation, and do so consistently regardless on the data type, dataset size (as long as they were "enough") and $P_{neg/pos}$ balance.

The metric to evaluate performance were ROC and PRC score, evaluated both on training dataset and out of sample; the best case scenario we aimed for and achieved, was that of $W_{model} \simeq W_{true}$, while ROC/PRC scores should be close to the maximum that could be achieved on each dataset.

Interaction matrix extimation The first thing to notice is that the dataset and interaction matrix we used, did set a limit to the model's ability to predict link: since each interaction is sampled from a binomial distribution centered in $\sigma(W_0 + X_{out} W_{true} X_{in})$, there is intrinsic randomness in the dataset itself.
In real case scenario this is to be interpreted as the feature's inability to accurately predict an interaction outcome, meaning each interaction can be driven by factors outside the features we are considering; this is often the case when studying interaction matrix on social media, since many factors beside opinions and sociodemographic classes will

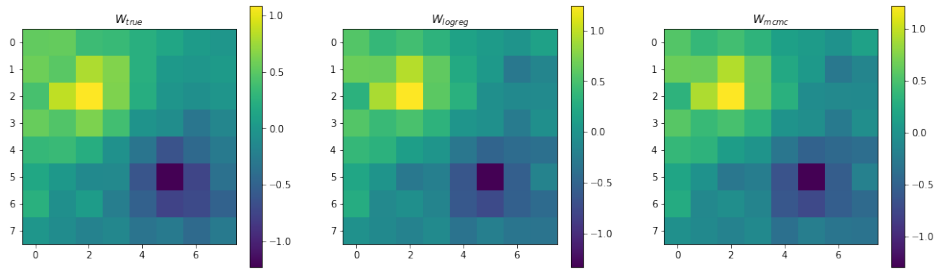

Figure 4.3: The results on the training performed on 10^4 and $P_{neg/pos} = 1$, where the true interaction matrix is W_{true} in fig.4.2. As can be seen, the trained interaction matrices were basically the same as the true one

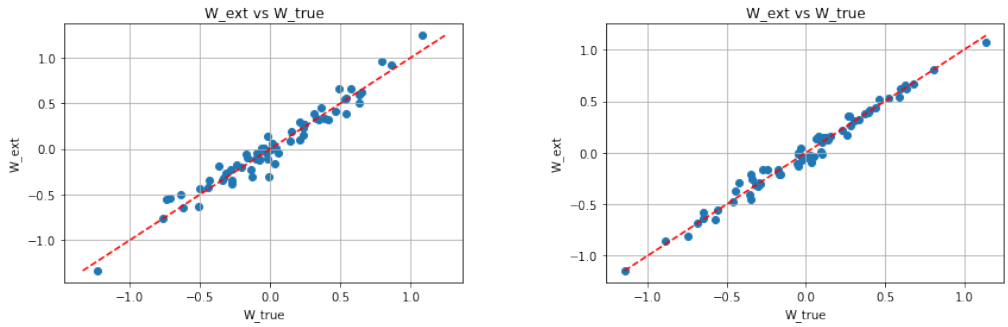

Figure 4.4: Comparison between W_{true} (on x axis) and model's estimated W_{ext} (y axis); on the left the feature matrix estimated through Logit model, on the right the one estimated with MCMC

be relevant -as for example the timing, the underlying friendship relations, the topic of discussion and so on.

When using our model on real data this meant our features might not have been highly explanatory, and thus we can not expect to accurately predict each individual interaction; in this sense, our method's estimates are to be considered mostly unable to individually predict each interaction, while possibly good at finding statistical tendency of large group of users' habits. On synthetic data, the low explanatory capability of the features is simulated by how close to $0, 1$ are the probability assigned by our W_{true} interaction matrix.
When evaluating the performances of our model, this means that the best possible outcome (and superior limit in model's prediction capability) is given by ROC and PRC curves calculated on the probabilities assigned by W_{true}. As can be seen in fig.4.7 this is indeed the case in our tests, since we found the ROC and PRC curves of our models to be consistently really close or even overlapping with those curves.
To further indicate model's effectiveness at learning W_{true} fig.4.4 shows how close are indeed the extimated interaction weights to the true ones.

In sample vs out of sample performances As figure 4.7 shows, ROC and PRC metrics "out of sample" are as good as the "in sample" ones. This is due to the fact that the model trains on an amount of data around $10^3 - 10^5$ bigger than its parameters, and would suggest that overfitting can not occour and validation tests might not be

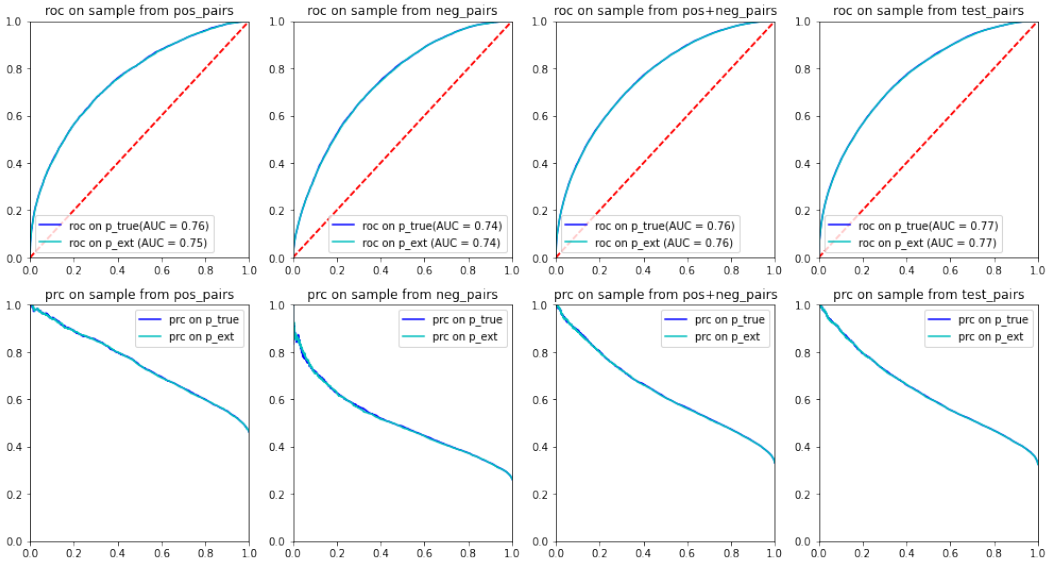

Figure 4.5: ROC and PRC curves on positive interactions (pos pairs), negative interactions (neg pairs), both of them and on pairs outside the training dataset (test pairs). These graphs shows model's inability to achieve the maximum possible performance (discussed in section 4.2.3), regardless of bein in/out of samples and of the true interaction outcome

required.
The only thing to look with more care when training on real data, is that different interactions might belong to the same author and/or same target, thus carrying same information and effectively reducing dataset size; this means that for not enough big of a dataset the model could actually overfit individual authors activity, especially the ones who are most active (having up to $\sim 10^4$ interactions in each year).

To avoid overfitting the activity of most active users, we observed that ROC auc score seems to converge to a dataset dependent minimum with higher dataset sizes. For example, when using real data, training the model with 10^3 interactions could give ROC auc of around 0.6, falling to around 0.55 for dataset of $10^4 - 10^5$ and remaining at these values for bigger datasets; this would imply that "node-wise overfitting" might occur when dataset are smaller than $10^4 - 10^5$ interactions. Without further discussion on this phenomenon, we considered our reference of training on 10^6 interactions to be good enough to prevent any kind overfitting.

Negative sampling and model's bias An important factor in model training is the proportion $P_{neg/pos}$ negative pairs with respect to positive pairs; commonly, values in the range of the unit are used in this kind of task. We were interested in studying how changing the negative sampling size could influence the models' outcomes: in particular, we expected it to only induce changes in models' biases while the feature matrix whould remain the same. This means we expected the balance of the dataset to be represented by the models' bias.

The model's ability to extimate the interaction matrix can be verified measuring model's ROC auc, while the relation bewteen $P_{neg/pos}$ and W_0 was meadured empirically. We thus trained the model with $P_{neg/pos}$ ranging from 0.5 to 5.0, and repeated

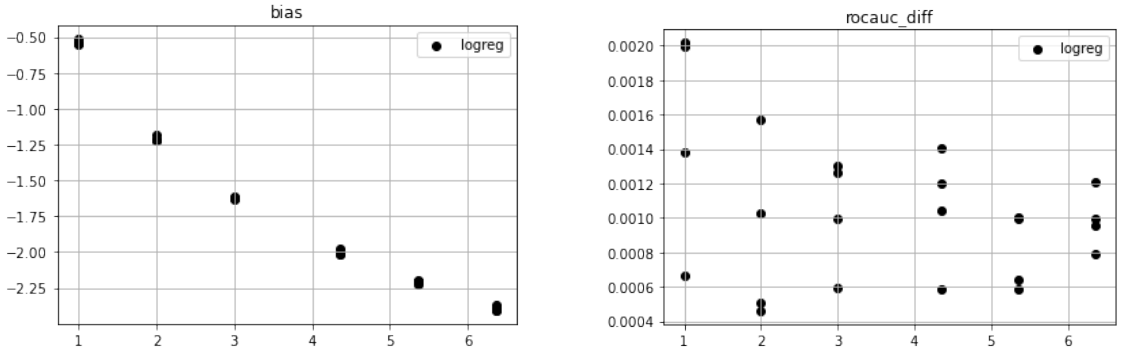

Figure 4.6: On either graphs, the x axis represent the chosen proportion of negative sampling $P_{neg/pos}$; on the left the y-axis represents the values of estimated bias W_0, while on the right the ROC auc of the corresponding model (in term of distance from the maximum possible ROC auc, which is the one achieved by true probabilities). These graphs are evaluated on Logistic regression model alone

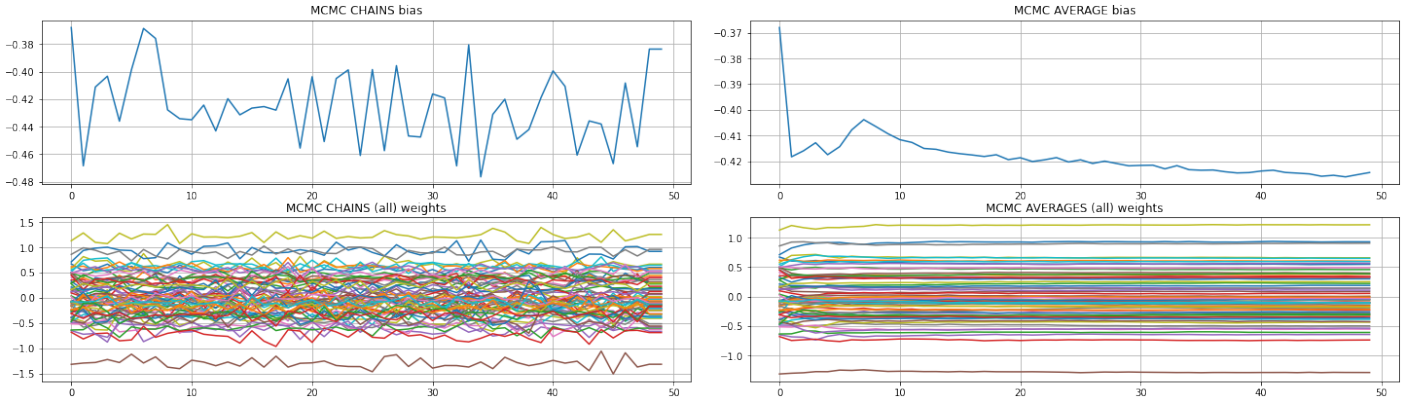

Figure 4.7: Sample chains generated by MCMC process (above), and their average (below). The values of convergence of these parameters are the ones of interaction matrix in fig.4.3, where similarity with Logit estimated interaction matrix is evident.

the process 5 times for each value. The results are plotted on fig. 4.6.
As that figure shows, the negative sampling proportion indeed only effect the bias of the model: while the ROC auc remains basically stable throughout this process, the bias consistently changes with the value of $P_{neg/pos}$.

These results suggest the models' extimated interaction matrices manage indeed to capture information about the interactions themselves, and are not influenced by dataset balance.

MCMC vs Logit model Since we trained the model with two different strategies, we were able to compare these two in term of computational cost and effectiveness. While we continued to use either the model in the next stage of our work, we observed the same effectiveness for the two models at the cost of much higer computational cost by MCMC model.

We observed that the MCMC chains' averages rapidly converged to the same values extimated by Logistic regression, as can be seen in figures 4.7 and 4.4. The speed at

which these averages converge is an indication of how localized is the posterior, since the MCMC model will be sampling with that distribution.
Together with the equivalence of these average and the Logit estimate (shown in fig.4.4), we could tell that the posterior was indeed highly dense around its average value, which also happened to be the maximum value of the likelihood.

The MCMC model nonetheless provided us a full estimate of the posterior, which we could verify was indeed regular and did not presents bimodal structure or significant local maximum other than its peak. Another example of this regularity is presented in the next section when training on real data.
Either the models, furthermore, provided ways to statistically evaluate our results: MCMC model's convergence can be estimated with Gelman-Rubin convergence index [22] (which is conveniently provided by pyro library) while chi-squared and confidence interval measure can be applied to Logistic regression model (which is conveniently provided by statsmodel library).

For all these reason we considered the Logit model to be a more effective model to learn the interaction matrix, since at the cost of much bigger computational costs MCMC model did not provide much more meaningful information. We thus decided to use MCMC only in first stage of the training, with small sub-sample of the dataset, just to have a rough estimate of the posterior and ensure no bimodal or pathological structure was found.
Nonetheless, knowing that the average of the prior coincided with the maximum of the likelihood told us that the logistic regression weights are a meaningful choice to extimate the interaction matrix.

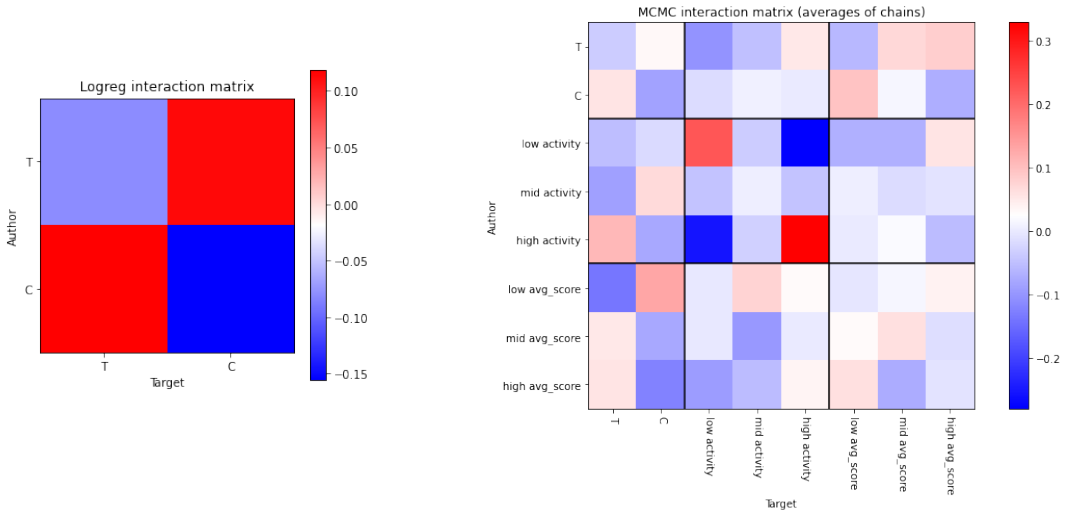

Figure 4.8: Interaction matrices obtained with on data from "No Echo in the Chambers of Political interactions on Reddit"[6]. Figure on the left shows the results using only Logit model and only political labels ("C" stands for Clinton supporter, "T" Trump supporter); figure on the right was instead obtained using MCMC model, considering also features of involvement (activity) and popularity (score). From the top left partition of the second matrix we can see how the disassortative behavior is persistent when using these confounding factors; Overall, all features interaction are coherent with the expectation given by [6].

4.3 Comparison with state of the art results

In the following stage of our work, before training the model on data generated through the pipeline discussed in section 3.1, we looked how it performed on previously collected data and studied if yielded expected results. We thus compared it with work done in 2020 [6] where there was observed to be a non-assortative interaction between opinion based features, where left leaning users interacted more with right leaning ones and vice versa.
The dataet will consists of users' interactions in subreddit 'politics' during 2016, year of American presidential elections, together with their political label (stating whether a user is a Trump or a Clinton supporter) and measures of their activity on that subreddit.

We were interested to see if our model found the same results of this no political-partisan echo chamber, together with assortative and disassortative behavior of activity and popularity respectively; not only we indeed found these same kind of interactions but we also provided a model which was more intuitive and more effective than the previous method, since it allowed the study of more interaction at the same time and thus the use of more confounding factor.

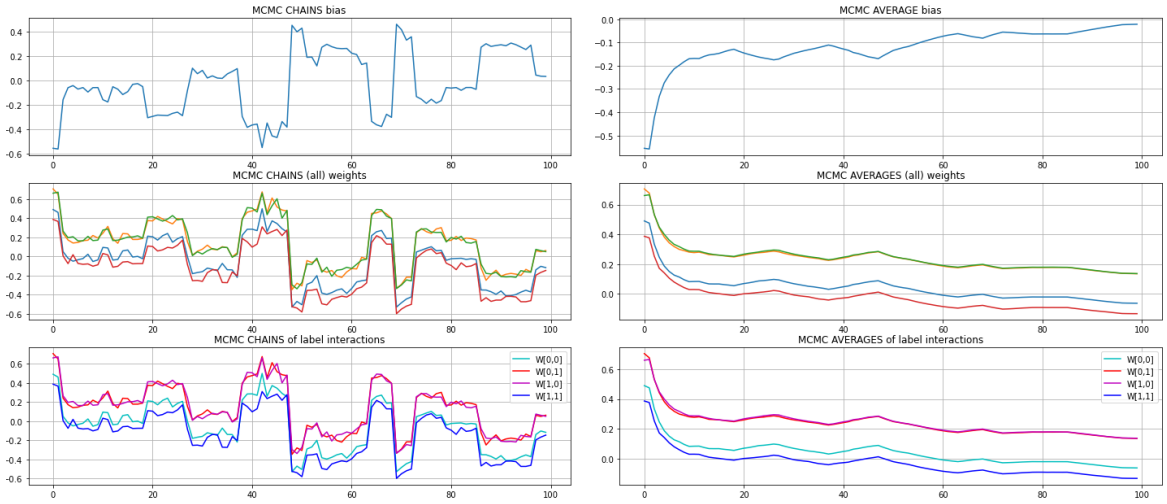

Figure 4.9: MCMC chains and their average when training on political labels alone. Gelman-Rubin statistic was around 1.2 for each of these parameters, meaning the convergence was decent but more samples might have been required - as can be seen from the graphs on the right. Nonetheless the average displayed clear disassortative behavior, and since this training was performed on a relatively small subset of dataset this is indeed coherent with our expectations

4.3.1 Models' equivalence

The author's data we used in this process was then the same that was used in that reference work [6]. To compare the model we looked ad the estimated log odds in the two, which could be computed through different processes.
The reference work performed one of the transformation discussed in 3.4.3, then training a Logit model and estimating the log odds with the regressor parameter.

The equivalence of this process can thus be established through eq.3.24, looking at the transformation they applied to user's data. In general the data used in that work consisted in vectors of three elements: the fist and second were two mutually exclusive values in $0, 1$ indicating leaning toward left and right respectively; the third was one of the confounding factor.

- $X^1_{in,out} \in \{0, 1\}$ chosen so that the first component of the feature vector is 0 if the user is a Trump supporter, 1 if a Clinton supporter
- $X^2_{in,out} \in \{0, 1\} = \neg X^1_{in,out}$
- $X^3_{in,out} \in [0, 1] =$ normalized confounding factor

When preparing the data for the Logit mode, the transformation used on such composed pairs of users' vectors was $T : (t^1, ..., t^6)$ where the first 3 elements represented interactions between political labels:

- $t^1 : \tilde{X}^1 = X^1_{out}$
- $t^2 : \tilde{X}^2 = X^1_{in} xor X^1_{out}$
- $t^3 : \tilde{X}^3 = X^1_{out} and (X^1_{in} xor X^1_{out}) = X^1_{out} and \neg X^1_{in}$

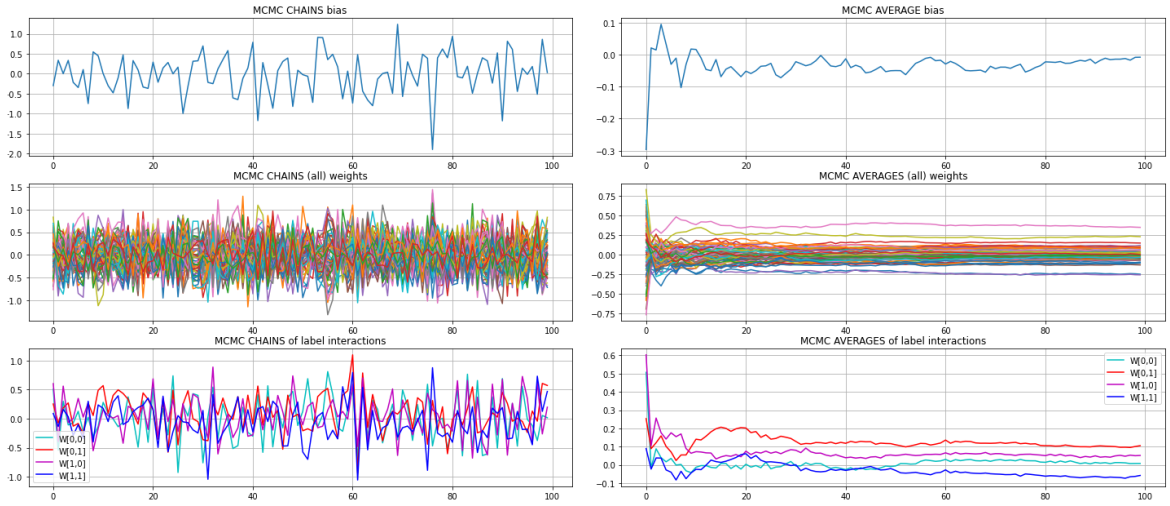

Figure 4.10: MCMC chains and their average when training on both political labels and confounding factors (users' activity and popularity). The first and second lines of graphs shows the chains on model's bias W_0 and interaction matrix W. The last line of graph shows, among all these values, the ones regarding interactions of political labels - which against show systematic disassortative behavior. The final values of these chains are the same reported in right figure 4.8

Using the second component of the feature vectors, these can be written as linear compositions:

- $t^1 : \tilde{X}^1 = X^1_{out}$
- $t^2 : \tilde{X}^2 = (X^1_{in} and X^2_{out}) or (X^2_{in} and X^1_{out})$
- $t^3 : \tilde{X}^3 = X^2_{in} and X^1_{out}$

The other transformations were interactions bewteen the two user's confounding factors, and can be written as:

- $t^4 : \tilde{X}^4 = X^3_{out}$
- $t^5 : \tilde{X}^5 = X^3_{in}$
- $t^6 : \tilde{X}^6 = X^3_{out} - X^3_{in}$

This transformation T can't be written as a matrix (unless using $X^4_{in,out} = 0$). At most, it can be decomposed in a tensor-like transformation (for the first 3 elements of transformed vector) and other functions such that:

- $t^1 = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
- $t^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

- $t^3 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$
- $t^4 : \tilde{X}^4 = X^3_{out}$
- $t^5 : \tilde{X}^5 = X^3_{in}$
- $t^6 : \tilde{X}^6 = X^3_{out} - X^3_{in}$

Using the eq.3.32 obtained before, where we set $k = \tilde{k}$ (same sigmoid steepness) and separating the linear-like transformation from the rest of the components'transformation we obtain

$$\begin{gathered} W_0 + \sum_{i=1}^{i=N} \sum_{j=1}^{j=N} W_{i,j} X^j_{in} X^i_{out} = \\ \tilde{W}_0 + \sum_{n=1}^{n=3} \sum_{j=1}^{j=3} \sum_{i=1}^{i=3} \tilde{W}_n t^n_{i,j}(X_{in}, X_{out}) + \tilde{W}_4 X^3_{out} + \tilde{W}_5 X^3_{in} + \tilde{W}_6 (X^3_{out} - X^3_{in}) \end{gathered} \tag{4.5}$$

Calling $\beta_n \equiv \tilde{W}_n$ the regressor's weights used in Logit model, and $s_{in} \equiv X^3_{in}$,$s_{out} \equiv X^3_{out}$ the confounding factor values of the target and author respectively:

$$\begin{gathered} W_0 + \sum_{i=1}^{i=3} \sum_{j=1}^{j=3} W_{i,j} X^j_{in} X^i_{out} = \\ \beta_0 + \beta_1 (X^1_{out} X^1_{in} + X^1_{out} X^2_{in}) + \beta_2 (X^1_{out} X^2_{in} + X^2_{out} X^1_{in}) + \\ \beta_3 (X^2_{out} X^1_{in}) + \beta_4 X^3_{out} + \beta_5 X^3_{in} + \beta_6 (X^3_{out} - X^3_{in}) \end{gathered} \tag{4.6}$$

which gives the correspondence of the regressor's weights used in reference work (right side of the equation) and the interaction matrix we will compute following our strategy. More specifically, when computing the log odds for possible feature pairs, this equation tells us correspondence between the two models for each possible case:

X^1_{out}	X^1_{in}	$ln(\Theta_{This})$	$ln(\Theta_{Ref})$
0 (T)	0 (T)	$W_0 + W_{2,2} + s_{in} s_{out} W_{3,3} + s_{in} W_{2,3} + s_{out} W_{3,2}$	$\beta_0 + \beta_4 s_{out} + \beta_5 s_{in} + \beta_6 (s_{out} - s_{in})$
0 (T)	1 (C)	$W_0 + W_{2,1} + s_{in} s_{out} W_{3,3} + s_{in} W_{2,3} + s_{out} W_{3,1}$	$\beta_0 + \beta_2 + \beta_4 s_{out} + \beta_5 s_{in} + \beta_6 (s_{out} - s_{in})$
1 (C)	0 (T)	$W_0 + W_{1,2} + s_{in} s_{out} W_{3,3} + s_{in} W_{1,3} + s_{out} W_{3,2}$	$\beta_0 + \beta_1 + \beta_2 + \beta_3 + \beta_4 s_{out} + \beta_5 s_{in} + \beta_6 (s_{out} - s_{in})$
1 (C)	1 (C)	$W_0 + W_{1,1} + s_{in} s_{out} W_{3,3} + s_{in} W_{1,3} + s_{out} W_{3,1}$	$\beta_0 + \beta_1 + \beta_4 s_{out} + \beta_5 s_{in} + \beta_6 (s_{out} - s_{in})$

Where (T) stands for Trump supporter, and (C) stands for Clinton supporter. When no confounding factor is used, these equivalences become just

X^1_{out}	X^1_{in}	$ln(\Theta_{This})$	$ln(\Theta_{Ref})$
0 (T)	0 (T)	$W_0 + W_{2,2}$	β_0
0 (T)	1 (C)	$W_0 + W_{2,1}$	$\beta_0 + \beta_2$
1 (C)	0 (T)	$W_0 + W_{1,2}$	$\beta_0 + \beta_1 + \beta_2 + \beta_3$
1 (C)	1 (C)	$W_0 + W_{1,1}$	$\beta_0 + \beta_1$

Where the relations between the two different models' parameters is now evident, and can be used -after training- to indeed compare their results.

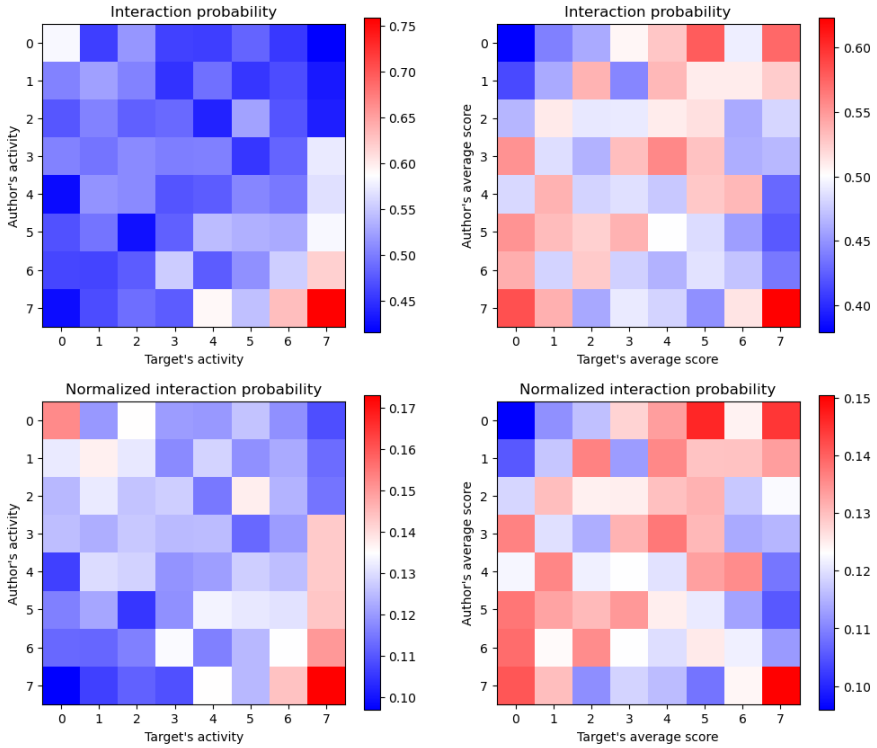

Figure 4.11: The two matrices on top shows the probability that and interaction will be observed $P(Y = 1|C_a^i, C_t^j)$, taking an author and a target whose activity/popularity measure fall in bins i and j respectively. The second row shows the same matrices normalized by rows $\frac{P(Y=1|C_a^i,C_t^j)}{\sum_k P(Y=1|C_a^i,C_t^k)}$, representing the probability that taking a comment of an author whose feature falls in i bin, the target's feature will fall in j bin. These pair of matrices show the expected results (which are more evident in the normalized ones): assortativity with respect to activity, and disassortativity with respect to popularity.

4.3.2 Coherence with expected results

We expected the model to find three different results:

- Disassortative behavior for political labels interactions
- Assortative behavior for users' activity
- Disassortative behavior for users' popularity

First we trained our models on a sample of $\sim 10^4$ interactions, using political labels alone. Either MCMC model or Logreg model show evident disassortative behavior in interactions, as can be seen in fig.4.9.
It is to be taken into account that due to long time of the MCMC model's training these early tests were performed on small datasets alone, so our observations and expectations regarded qualitative behavior more than precise quantitative observations.

We later trained the model on measures of users' activity and popularity, the number of written comments and the average score these comments managed to gather. To do so we quantile-normalize these features over edges occurrences ("edge-wise normalization"), meaning we had a uniform distribution of the activity and popularity

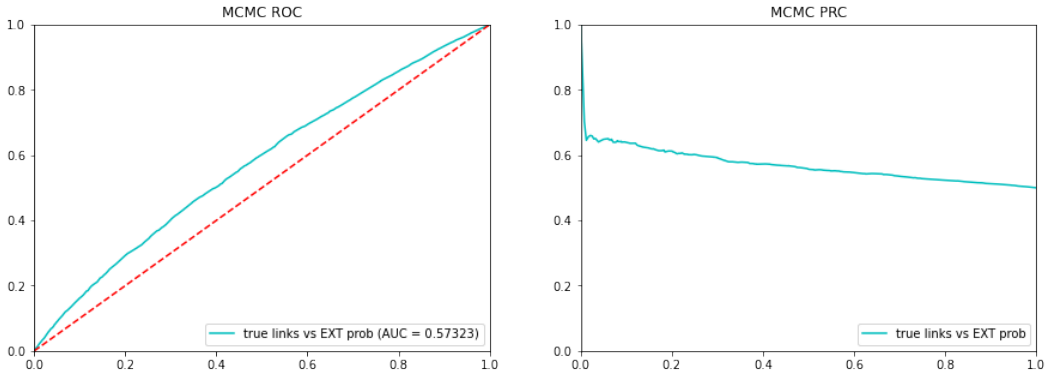

Figure 4.12: The ROC and PRC measure on MCMC model's estimated interaction'probabilities. Low values of the ROC auc tells us that the model can't be considered good in prediction task itself; this is nonetheless typical of link prediction task based on limited amount of features, meaning these features can't be predictive enough to make accurate estimation on individual interactions. As it is discussed above, this strategy instead aim at mostly finding statistical leanings in users' habits, which are for example the assortative or disassortative behavior and/or no-echo chamber phenomenon discussed in this section

measure associated with each occurrences over the dataset; this strategy, compared with normalization over nodes' occurrences gave more relevance to lower values of observed activity and popularity of the users.

We then mapped these values in 8 bins according to eq.3.3, and trained the model on each one of these measure; the results are shown in fig.4.11. The first of matrices, representing interactions based on users' activity, indeed display assortative behavior; the third column of matrix, representing interactions based on users' popularity in that subreddit, instead display disassortative behavior.

Finally, we trained the model using political labels, activity and popularity altogether. Each feature was treated as it was just described, the only difference being we divided activity and popularity feature in 3 bins (representing high/mid/low popularity or activity), to reduce MCMC training time but still have interpretable results. Fig.4.13 shows how each the feature behaved as observed shortly before, since the diagonal partitions of the interaction matrix show indeed the disassortative-assortative behavior we expected (either in MCMC model's and Logit's model extimate. Results of logistic regression model are not reported here since the found interaction matrices were the basically same as the MCMC found, according to our findings on synthetic data.

In addition, the interaction matrix represented interactions between political labels and confounding factors, as for example the fact that Trump supporter are apparently more prone to respond to more active user.

In table above we reported the statistical measures on the model's parameter estimate; particularly r-hat stands for Gelman-Rubin statistics and tells us a good convergence is reached, being really close to 1 for each parameter. N_{eff} stands instead as the MCMC model efficacy in exploring the parameters' space: the higer it is the more the chains had been able to extract informative sample on that space [10], further implying the good convergence we achieved.

	mean	std	median	n_eff	r_hat
s	-0.01	0.53	0.03	156.07	1.00
w[0,0]	0.01	0.41	-0.00	275.68	0.99
w[0,1]	0.11	0.33	0.09	170.94	0.99
w[0,2]	-0.05	0.30	-0.11	107.67	0.99
w[0,3]	-0.05	0.36	-0.10	202.61	0.99
w[0,4]	0.12	0.29	0.10	134.23	1.00
w[0,5]	-0.13	0.29	-0.17	110.42	1.00
w[0,6]	0.04	0.35	0.02	155.20	0.99
w[0,7]	0.05	0.30	0.03	94.98	0.99
w[1,0]	0.05	0.37	0.03	230.10	0.99
w[1,1]	-0.06	0.35	-0.07	181.70	1.00
w[1,2]	-0.00	0.36	-0.01	180.00	0.99
w[1,3]	0.10	0.42	0.06	158.49	1.00
w[1,4]	-0.05	0.33	-0.04	193.07	1.01
w[1,5]	0.15	0.33	0.17	216.84	0.99
w[1,6]	-0.06	0.41	-0.05	123.07	0.99
w[1,7]	-0.11	0.31	-0.11	324.36	0.99
w[2,0]	-0.07	0.36	-0.03	267.27	0.99
w[2,1]	-0.03	0.33	-0.03	94.12	0.99
w[2,2]	0.23	0.30	0.21	86.38	0.99
w[2,3]	-0.05	0.32	-0.08	205.24	1.00
w[2,4]	-0.25	0.32	-0.24	104.32	1.01
w[2,5]	-0.04	0.30	-0.07	112.25	1.00
w[2,6]	0.03	0.30	0.00	131.97	0.99
w[2,7]	-0.07	0.30	-0.09	171.89	1.01
w[3,0]	-0.03	0.32	-0.06	233.96	0.99
w[3,1]	-0.02	0.37	-0.02	351.02	0.99
w[3,2]	-0.04	0.37	-0.03	95.67	0.99
w[3,3]	-0.01	0.31	-0.02	106.03	0.99
w[3,4]	-0.03	0.38	-0.02	110.64	0.99
w[3,5]	0.06	0.34	0.05	194.33	0.99
w[3,6]	-0.05	0.28	-0.07	140.56	0.99
w[3,7]	-0.01	0.32	-0.01	348.74	0.99
w[4,0]	0.08	0.38	0.11	208.21	0.99
w[4,1]	-0.01	0.39	-0.03	121.47	0.99
w[4,2]	-0.26	0.36	-0.25	195.58	1.00
w[4,3]	-0.05	0.37	-0.01	145.72	1.01
w[4,4]	0.35	0.32	0.32	125.92	1.02
w[4,5]	-0.02	0.34	-0.02	355.03	1.00
w[4,6]	0.02	0.32	0.02	153.35	0.99
w[4,7]	0.04	0.27	0.04	143.24	1.01
w[5,0]	-0.11	0.36	-0.14	190.34	0.99
w[5,1]	0.09	0.30	0.06	122.09	0.99
w[5,2]	-0.05	0.33	-0.11	214.25	0.99
w[5,3]	-0.00	0.29	-0.04	75.60	1.00
w[5,4]	0.01	0.33	-0.02	132.27	0.99
w[5,5]	0.01	0.36	0.01	147.56	0.99
w[5,6]	0.00	0.30	0.01	234.97	0.99
w[5,7]	0.04	0.31	-0.00	88.45	1.02
w[6,0]	0.02	0.33	0.00	135.62	1.04
w[6,1]	-0.01	0.34	0.03	72.47	1.02
w[6,2]	-0.10	0.35	-0.09	151.44	1.00
w[6,3]	-0.05	0.35	-0.08	78.95	1.00
w[6,4]	-0.00	0.39	-0.00	157.55	0.99
w[6,5]	0.09	0.31	0.07	97.67	0.99
w[6,6]	0.07	0.30	0.05	108.45	0.99
w[6,7]	-0.05	0.29	-0.07	100.90	1.00
w[7,0]	0.06	0.44	0.05	172.39	1.01
w[7,1]	-0.07	0.41	-0.06	134.00	1.00
w[7,2]	0.03	0.29	-0.01	198.07	1.00
w[7,3]	-0.03	0.34	0.00	115.10	0.99
w[7,4]	-0.10	0.36	-0.09	198.04	0.99
w[7,5]	0.08	0.36	0.10	575.01	0.99
w[7,6]	-0.01	0.29	0.01	453.06	0.99
w[7,7]	0.00	0.31	0.01	308.68	0.99

Table 4.3: Statistics over MCMC estimated posteriors over each parameter; In particular are here reported averages, standard deviation and Gelman-Rubin statistics (r_hat). This table also shows how median value is often really close to average value (hinting a denser posterior around its center), and MCMC efficacy in exploring parameter space. N_{eff} stands for number of effective samples, to compare with $N_{samples} = 100$ which we chose: higher values tells MCMC efficiency in exploring the posterior

Without going into more details about the results themselves (which are already well documented in the reference work we are comparing with [6]), we thus concluded that our model successfully managed to capture the expected results, further providing some additional information about confounding factor and with an easily interpretable framework.

4.3.3 Improvements

Comparing our strategy with the one used in the reference work [6], some improvement has been achieved in different ways.
First of all our strategy of building the feature vector appears more natural and easier to work with compared to the transformation applied in 4.3.1. This not only means that the preprocessing from feature to training can be more straightforward, but also allows to work with the features altogether instead of one confounding factor at a time.

As a consequence, more information can be acquired about interactions between many different features; in particular, being interested in some feature in particular, following this strategy one can asses the consistency of an hypothesis when more confounding factors are involved. In this case, looking ad political label (feature of interest) and activity/popularity measures (confounding factors) we have been able to establish that the disassortative behaviour of political opinion based interaction is not a phenomenon due to activity of popularity of users, being instead apparently an intrinsic motivation that drive users's interactions

In conclusion, this method thus provides improvements not only on usability and interpretability of the model but also on more information provided about interactions we are observing.

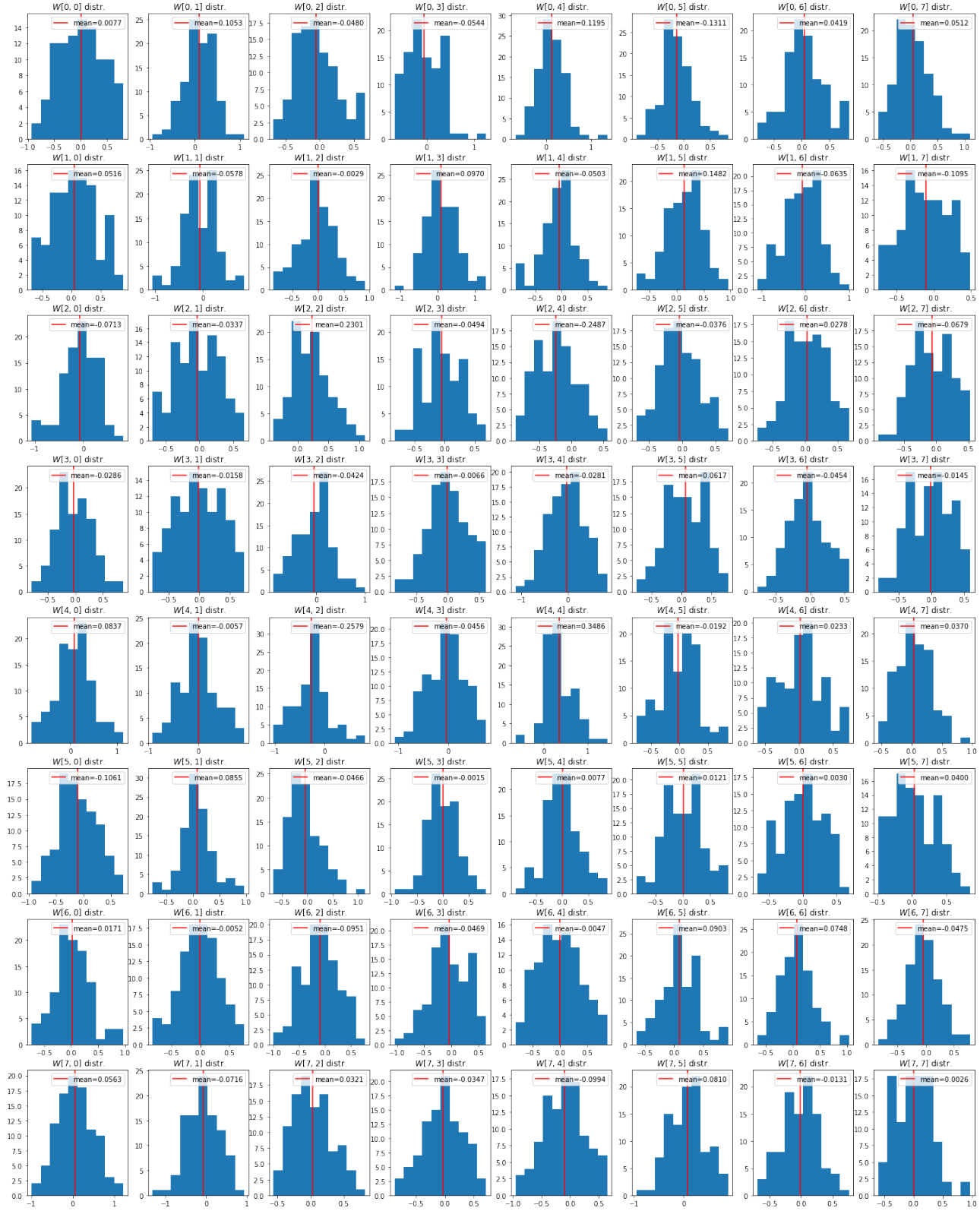

Figure 4.13: Posterior estimated distribution with MCMC, for every single pair used in testing against previously known results (section 4.3. These distribution can roughly show how the whole posterior itself is indeed approximately denser around its average, with pretty regular center field like topology and no other local maximum to stand off.

Chapter 5

Results

We focused our research on interactions observed yearly between 2016 and 2020, on 'news' and 'politics' subreddits separately; dataset size, number of users and interactions are the ones reported in tab.3.1. We observed interactions based on sociodemographic classes and opinions/communities, extracted through the pipeline described in section 3.1.
We learned interaction matrices on each of these datasets with $P_{neg/pos} = 1$ negative sampling proportion, using Logistic regression alone since MCMC learning process proved to be computationally intensive for these amount of data.

Results from interaction probability estimation can then be combined with observed average sentiment, to have more of a specific information about how each feature influence users' leaning.
Looking at these values from different perspectives, as the ones presented in next sections, will then provide an overall understanding of what users' activity is driven by.

5.1 Features statistics

After the preprocessing, the first thing we did was to evaluate some statistics on each dataset; this was useful not only as a result itself, but also as a way to calibrate the model.

VIF index was indeed measured to be around $VIF(F^i) \sim 1$ for all the features in the model, which told us that collinearity was low and Logit estimate would have been consistent. Feature co-occurrences and edges features pair co-occurrences was also useful for deciding dataset sizes, since we established the Logit model was required at least $\sim 10^2$ edges for each observable pairs - counting both actually observed ones and the ones generated from negative sampling.
Specifically table 5.1 shows that least frequent features are present in around 10^2 nodes, while table 5.2 shows that least frequent features pairs are present in indeed around 10^2 edges. This is the case from specific opinion features from clustering, like belonging to LGBT community, to pro gun community or being an environmentalists; as it is to be expected, belonging to a specific community is generally represented less than having some sociodemographic status, by some order of magnitude.

	young	old	male	female	democrats	conservative	poor	rich	low activity	high activity	(c) mod left	(c) ext left	(c) anti trump	(c) conservatives	(c) pro gun	(c) lgbt comm	(c) environment
young	6690	0	2087	1109	757	2734	3397	382	1999	1198	485	980	113	105	73	105	13
old	0	6688	1288	2341	2590	994	318	3663	1418	2058	728	887	339	196	222	17	309
male	2087	1288	6694	0	808	2744	1924	1619	1779	1579	477	940	176	173	333	15	36
female	1109	2341	0	6692	2878	714	1120	2157	1719	1623	539	762	180	71	57	148	119
democrats	757	2590	808	2878	6696	0	1910	1432	1484	1934	1093	1911	609	24	129	158	217
conservative	2734	994	2744	714	0	6691	1601	1917	1870	1377	228	338	42	407	262	13	37
poor	3397	318	1924	1120	1910	1601	6694	0	1867	1418	645	1559	314	151	159	162	59
rich	382	3663	1619	2157	1432	1917	0	6692	1506	1780	451	520	148	134	131	10	167
low activity	1999	1418	1779	1719	1484	1870	1867	1506	6638	0	425	860	151	161	162	65	67
high activity	1198	2058	1579	1623	1934	1377	1418	1780	0	6678	913	1222	380	160	170	63	205
(c) mod left	485	728	477	539	1093	228	645	451	425	913	2462	651	267	99	75	30	120
(c) ext left	980	887	940	762	1911	338	1559	520	860	1222	651	4130	412	100	146	92	176
(c) anti trump	113	339	176	180	609	42	314	148	151	380	267	412	1012	31	33	17	74
(c) conservatives	105	196	173	71	24	407	151	134	161	160	99	100	31	635	28	4	26
(c) pro gun	73	222	333	57	129	262	159	131	162	170	75	146	33	28	683	7	17
(c) lgbt comm	105	17	15	148	158	13	162	10	65	63	30	92	17	4	7	264	5
(c) environment	13	309	36	119	217	37	59	167	67	205	120	176	74	26	17	5	463

Table 5.1: Each value in this table represent number of observed nodes possessing both row and column feature. Features labelled as "(c)" are the ones obtained from clustering method.

	young	old	male	female	democrats	conservative	poor	rich	low activity	high activity	(c) mod left	(c) ext left	(c) anti trump	(c) conservatives	(c) pro gun	(c) lgbt comm	(c) environment
young	47748	148168	65195	84946	119029	63045	53255	117664	17915	255830	77681	80221	36365	12483	7280	2185	46769
old	64568	278025	103739	143374	208888	99018	80009	207454	26324	446095	137354	139836	68855	24482	12865	3255	92585
male	54568	188623	79523	103302	147947	75230	63200	146303	20716	319677	96415	99839	46176	15821	9091	2498	59841
female	56070	213392	83400	117109	163172	80959	64970	164999	22112	350400	105991	105939	52204	18071	9810	2752	69724
democrats	63440	246490	96810	131870	191071	91161	78009	184668	24835	406611	125307	130279	60155	21742	11618	3101	79234
conservative	49345	177475	71671	95796	137220	69909	56732	138249	19255	296322	89827	92342	44917	14769	8350	2320	58818
poor	52329	178317	75008	97819	141101	71701	62063	137263	19826	303468	92492	96667	43922	15661	8749	2496	57087
rich	57603	234621	89421	124420	177639	85761	67733	179425	23554	379813	116427	117655	58176	19610	10499	2758	77694
low activity	13994	51594	20142	27096	40501	19959	15948	41473	5609	85279	26623	27348	13496	3762	2246	729	17981
high activity	151803	578153	230844	311103	443220	219242	183391	434040	58804	957646	289949	297409	140429	52332	28137	7335	183598
(c) mod left	30828	111590	46395	61137	90158	42175	38402	81958	11871	188837	58946	65482	26217	10239	5249	1489	34504
(c) ext left	41528	152971	62634	82529	123483	57563	52298	113258	16692	258701	83900	87252	37620	13789	7410	2038	48324
(c) anti trump	11625	53835	19339	26339	39998	19106	15176	39303	4845	85272	26496	27507	13872	5140	2400	549	18730
(c) conservatives	5587	23804	8455	12247	18109	8307	7148	17368	2269	38810	11830	12530	6298	2027	993	273	8231
(c) pro gun	6055	22704	9186	11876	17100	8935	7473	16832	2332	37616	10984	11653	5593	1933	1560	295	7144
(c) lgbt comm	2364	6399	3063	4016	5242	2940	2644	5082	824	11632	3277	3567	1462	556	334	149	1817
(c) environment	4802	25859	8341	12374	18920	8124	6485	18381	2120	39960	12827	12948	6611	2216	1095	262	9230

Table 5.2: Co-occurrences in observed edges. Each value represent the number of observed edges where the author has the row feature and target has column feature.

It is also of interest to observe features correlations: table 5.3 shows probability that a user selected among the most active ones in each dataset will share a pair of features. In particular, this gives an overview of relations between sociodemographic categories and possible opinions. The results found in this way will be context dependent, as they are observed only on most active users of a given subreddit.
Found probabilities actually met what we expected by common sense: as an example, conservatives people have higher chances of being old, gun enthusiast of being male, LGBT supporter to be female. These values will not thus not only give us insights about our features extraction strategy, but also contribute to general overview of userbase in a given context and their tendencies.

	young	old	male	female	democrats	conservative	poor	rich	low activity	high activity	(c) mod left	(c) ext left	(c) anti trump	(c) conservatives	(c) pro gun	(c) lgbt comm	(c) environment
young	0	0	31	16	11	40	50	5	29	17	7	14	1	1	1	1	0
old	0	0	19	35	38	14	4	54	21	30	10	13	5	2	3	0	4
male	31	19	0	0	12	40	28	24	26	23	7	14	2	2	4	0	0
female	16	34	0	0	43	10	16	32	25	24	8	11	2	1	0	2	1
democrats	11	38	12	42	0	0	28	21	22	28	16	28	9	0	1	2	3
conservative	40	14	41	10	0	0	23	28	27	20	3	5	0	6	3	0	0
poor	50	4	28	16	28	23	0	0	27	21	9	23	4	2	2	2	0
rich	5	54	24	32	21	28	0	0	22	26	6	7	2	2	1	0	2
low activity	30	21	26	25	22	28	28	22	0	0	6	12	2	2	2	0	1
high activity	17	30	23	24	28	20	21	26	0	0	13	18	5	2	2	0	3
(c) mod left	19	29	19	21	44	9	26	18	17	37	0	26	10	4	3	1	4
(c) ext left	23	21	22	18	46	8	37	12	20	29	15	0	9	2	3	2	4
(c) anti trump	11	33	17	17	60	4	31	14	14	37	26	40	0	3	3	1	7
(c) conservatives	16	30	27	11	3	64	23	21	25	25	15	15	4	0	4	0	4
(c) pro gun	10	32	48	8	18	38	23	19	23	24	10	21	4	4	0	1	2
(c) lgbt comm	39	6	5	56	59	4	61	3	24	23	11	34	6	1	2	0	1
(c) environment	2	66	7	25	46	7	12	36	14	44	25	38	15	5	3	1	0

Table 5.3: Correlation between features, shown as a percentage and approximed to its closest integer value. Each value is the probability that if a user has the feature indexed by row, it will also have the one indexed by column

5.2 Interactions' probability

As a mean to establish statistical significance of our results, for each parameter in each matrix the p-value and confidence interval is calculated; to clear observed data we thus first applied two criteria:

- $p - value \leq 0.05$, which ensures statistical significance itself.
- the two extremes of the confidence intervals are of the same sign, which suggest assortative/disassortative behavior can be considered statistically significant.

Filtering the interaction matrices value by which at least one of the two was true, we obtained a representation of interactions cleared up of noise and/or non significant interactions; these results are shown in fig.5.1.

It can be seen how the interactions involving features obtained from clustering often appear to be statistically non significant, possibly noisy due to the small amount of nodes belonging to those classes (around $\sim 10^2 - 10^3$). Nonetheless, many elements in each matrix come with some statistical significance meaning the model is mostly capable of extracting some relevant information on the dataset.

5.2.1 Context dependent similarities

Before focusing on specific interaction values, it can be interesting to ask how these matrices differ one another, to see if something can be said about year and subreddit dependent habits in general. This whould gives some insight about how interactions can be related to the specific context.

We thus applied different matrix distance measures to assess similarity between each of them. In particular, we measured element-wise squared difference, element-wise counting of sign differences, and either applied these metrics to the non fildered matrices

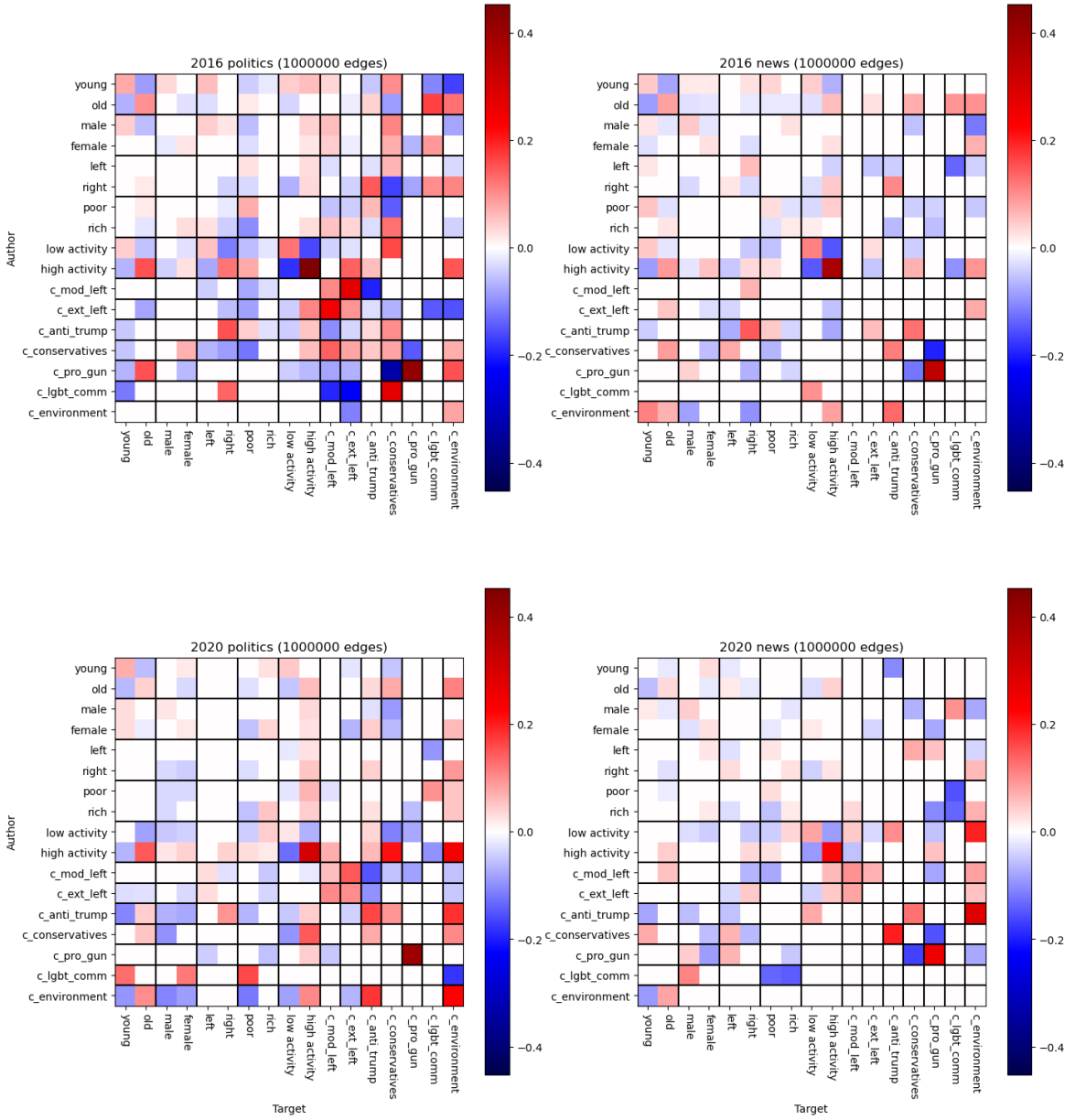

Figure 5.1: Interaction matrices on 2016 and 2020 datasets, where non statistically significant values (according to criteria defined in section 5.2)are represented in white. Features extracted through the clustering strategy are the ones with prefix 'c_'

or on the significant values alone. Most insightful are specifically the measures of how many values had different signs among the statistically significant ones, since they effectively counted qualitative different behavior among two datasets. Nonetheless, each of these measure actually showed similar trends as the one in fig 5.4.

These measures told us that higer similarities could be generally found in interaction matrices evaluated on same subreddit (which lead to the "grid like" shapes that can be seen in fig.5.4), and this effect is apparently more relevant than proximity in time: interactions in the same subreddit but different years appear to be more similar than interactions on the same year but different subreddit. This would indeed suggest

	2016 politics	2016 news	2017 politics	2017 news	2018 politics	2018 news	2019 politics	2019 news	2020 politics	2020 news
2016 politics	0	19	19	22	20	23	22	11	15	16
2016 news	19	0	15	7	18	1	18	6	13	5
2017 politics	19	15	0	15	6	17	9	13	8	13
2017 news	22	7	15	0	17	3	18	2	13	5
2018 politics	20	18	6	17	0	11	6	11	3	13
2018 news	23	1	17	3	11	0	17	1	8	6
2019 politics	22	18	9	18	6	17	0	16	2	13
2019 news	11	6	13	2	11	1	16	0	13	5
2020 politics	15	13	8	13	3	8	2	13	0	9
2020 news	16	5	13	5	13	6	13	5	9	0

Table 5.4: The values in this table represent the number of values that differ on sign between each pair of interaction matrices. Lower values (depicted with darker reds) are thus representative of more similar matrices

	2016 politics	2017 politics	2018 politics	2019 politics	2020 politics
2016 politics	0	19	20	22	15
2017 politics	19	0	6	9	8
2018 politics	20	6	0	6	3
2019 politics	22	9	6	0	2
2020 politics	15	8	3	2	0

	2016 news	2017 news	2018 news	2019 news	2020 news
2016 news	0	7	1	6	5
2017 news	7	0	3	2	5
2018 news	1	3	0	1	6
2019 news	6	2	1	0	5
2020 news	5	5	6	5	0

Table 5.5: The same values of table 5.4, ordered by year and divided by subreddit

that on the dataset we studied, users' habits were highly determined by the specific community they were participating in.

Looking at these values by dividing the two subreddits' datasets, time dependent similarities become more evident. First thing that can be noticed is how interactions in 2016 on subreddit politics, and in 2020 on subreddit news appear to be significantly more different from the others; this might be interpreted by the fact that in each of these context, relevant external factors could have higly influenced users' behaviors. 2016 was indeed a year of presidential elections, and it has been observed how that situation sparked strong debated in Reddit community [6] if not on the world in general; the same can be said for 2020, when the first outburst of Coronavirus probably fueled discussion and interest in the 'news' subreddit.
Intuitively then, in could be said that activity in these datasets strongly correlated with strong external events, though more research on more subreddits and years would be required to further study this phenomenon.

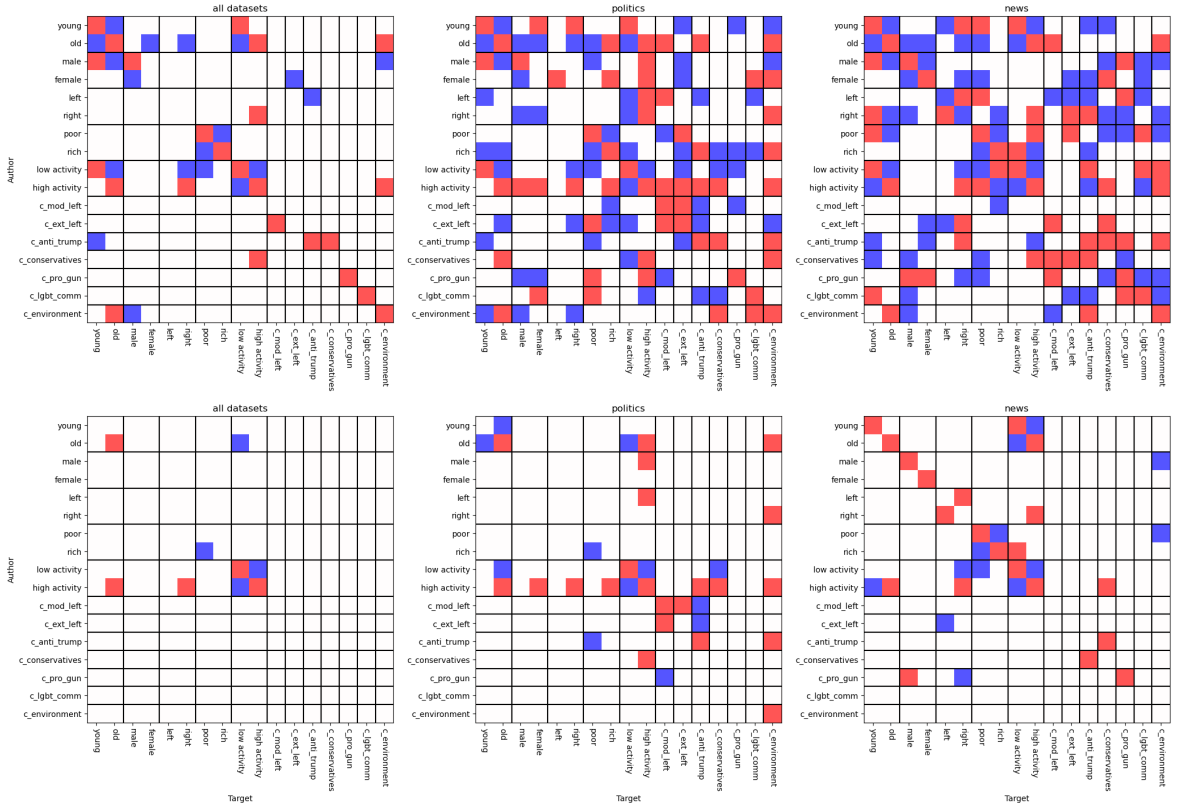

Figure 5.2: Interaction matrices' values that exhibited persistent assortativity (red) or disassortativity (blue) on different groups of datasets. Matrices above are obtained without any requirement on each value statistical significance, while the ones below are filtered by requiring that these values must have statistical significance in each observed year (according to criteria defined in section 5.2).

Additionally, a slight gradient can be seen in fig.5.5 (particularly on 'politics') where similarities appear to be higer the more years are closer, meaning there could be some time dependent trend underlying users' habits.

Unfortunately, observations on longer time spans and bigger amount of datasets in general would be required to assess any of these hypothesis with more certainty; nonetheless these observations meet common intuitions of what one could expect on relations between context and social habits.

5.2.2 Common trends

Given the subreddit appear to exhibits context specific trends, it is interesting to ask whether some features' interactions might be constant over all years. Moreover, if there are some which remains of same sign among all datasets in general.
To evaluate this, we can ask which one of the statistically significant interaction matrices' values remain of same sign in a certain group of dataset; using the two criteria defined above, this gives high statistical to observed features interactions. This request can be relaxed by just asking that the sign of a value of those matrices remains the same regardless of their p-values or confidence intervals: this strategy provides less certainty but can still capture interesting leanings in interactions.

The results of these two different strategy are represented in fig.5.2.

Overall constants Common interactions among all dataset appear only on a very limited amount of features' pairs; as the first matrix in fig.5.2 shows, the interactions which remains the same on all datasets are mostly the ones related to assortative behavior with respect to users' activity - meaning overall, very active users tend to interact more with other active ones, and new ones interact more with new ones.
Another constant trend in all datasets appear to be the tendency of people "older than the average" to interact more with users of same age, and the tendency of "richer than average" users to have little interactions with users of minor affluence. These two habits will be discussed in further details when we later focus on these features.
Interactions between sociodemographic features and activity ones might instead be due to collinearity between the two (as can be seen in tab.5.3), and are then less informative.

Relaxing the request of all values satisfying statistical significance, other trends emerges: particularly, looking at cluster based features we can see assortative (if not segregative) behavior when it comes to opinion/community, with guns enthusiast, LGBT community and environmentalist mostly interacting with users from the same community.
Lastly, community of anti Trump (as comments' author) seems to be in general interacting with conservatives (as comments' target), hinting at some debate that might be happening between the two.

Constants on politics subreddit To remove outliers, we ignored 2016 politics data and observed constants over years between 2017 and 2020. Other than the interactions related to activity, these datasets show interesting interactions when it comes to cluster based features - which we discussed can be interpreted as opinions and/or belonging to a certain community.
Specifically, interactions between left moderate and left extremist appeared to be systematically frequent, and the same appeared to be the case for anti trumpists; this would suggest tendency in assortative behaviors driven by opinion, which might be due to different arguments of conversation that each community leans toward.
Furthermore, constant behavior appear with respect to environmentalists: looking at rightmost column of that matrix, it appears that environmentalist are with high probability targeted by comments of older people than average and/or leaning right and/or participating in anti trump community.

Relaxing the statistical significance request, assortative behavior by opinion appears more evident (looking at diagonal values of bottom righ part of the matrix) together with more information about less frequent features' interactions. As an example, LGBT community now appears more active toward females (either as target or as author) conservatives toward older people (as author).

Constants on news subreddit Differently from politics datasets, interactions on news subreddit appears to be systematically driven by sociodemographic features: the top left part of the matrix, which display interactions between embedding based features, shows indeed significan assortative behavior by age, gender and affluence. Since

users are not in general able to recognize each others' sociodemographic status, these results on themselves might be interesting for further research, asking whether assortativity happens because of different arguments being talked about or are driven by other reasons.

Interestingly, non assortative behavior is instead shown with respect to political opinion and partisanship: interactions between left-wing and right-wing are indeed more frequent between members of opposite partisanship (as can be seen from third partition on matrix's diagonal), and the same goes for anti Trump community and conservatives. On the other side, interactions within left-sided communities (which were present in politics' datasets) here appear to be not consistent.
This can be interpreted as the context and subreddit format inducing more debate within users, thus enhancing interactions between opposite sided opinions and communities; not only it is interesting to notice that these datasets exhibits no echo chamber, and even disassortative behavior, but to also point out that this behavior can be context dependent.

Relaxing the significance assumption, in top right matrix in fig.5.2 each of these behavior is more evident (also negative values on sociodemographic values appear to be constants) and some information is gathered about smaller communities: without going into much details about each of them, looking at last three columns and rows we can see segregative behavior of small communities, mostly interacting within each other. Each one of them also exhibit some constant behavior regarding the sociodemographic class they address to: pro guns mostly writing to male, LGBT community to young and females, environmentalists to older people.

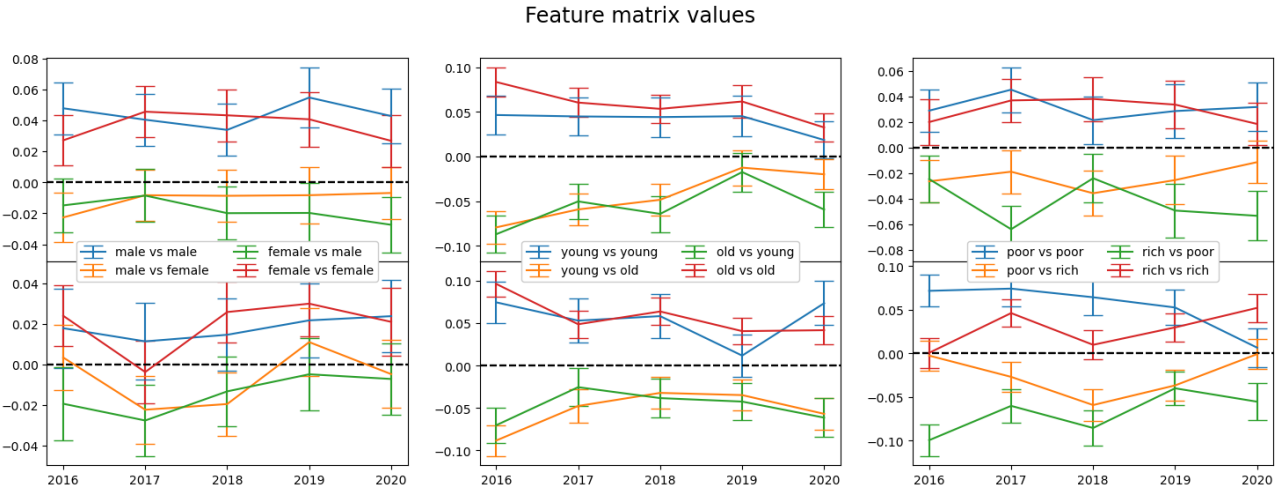

Figure 5.3: Feature matrix values regarding interactions of sociodemographic features, together with their confidence intervals. Graphs on top shows interactions on subreddit 'news', ones below on 'politics'.

5.3 Yearly trends

To see wether these interactions showed some time dependencies, and to see our results more clearly, we looked at these interaction matrices values year by year together with their confidence intervals.
Plots obtained in this way clearly showed that even in cases where confidence intervals were wider, some tendencies could be found for many features' interactions.

5.3.1 Sociodemographic features

Sociodemographics features in particular, showed persistent and significant assortative behavior. In section above we already observed this behavior on news datasets, though with less consistency this also appear to be true on politics datasets: the interaction odds of each feature pair interaction is firmly higher for same feature pairs than opposite feature pairs.

Though this seems intuitive, it is not obvious to provide an interpretation for this, since users do not generally know each other. A possible interpretation is that this behavior might be due to different topics that users get involved in; if this is true, the fact that these behavior appear to be more evident in news datasets could be explained by the fact that this different subreddit might provide a broader range of topics. This hypothesis will be later tested by analysis of correlation between users involvement in topics and their sociodemographic features.

5.3.2 Political opinion features

Behavior based on opinion and communities displayed wider variety, suggesting that socio-demographic status is instead a more important factor when it comes to users decisions of who to interact with. Nonetheless, as can be seen from fig.5.4, interesting

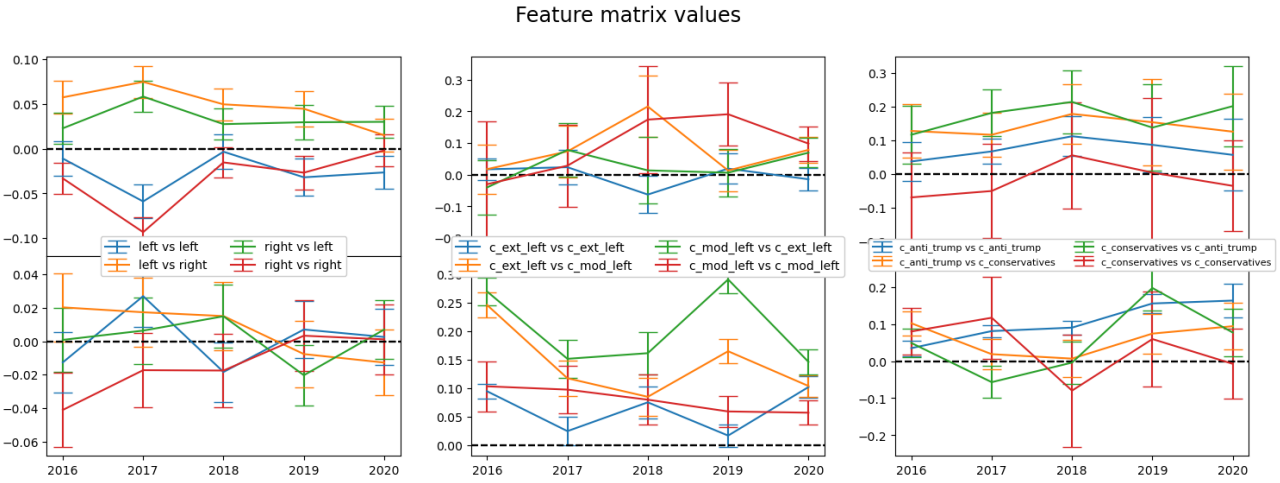

Figure 5.4: Feature matrix values regarding interactios of opinion based features, together with their confidence intervals. Graphs on top shows interactions on subreddit 'news', ones below on 'politics'.

tendencies can be observed.

First of all, none of the observed politic opinion features displayed any meaningful echo-chamber like behaviour: in no case, interactions between same-opinion users appeared to be significantly and consistently more probable than interactions between opposite opinion users. While we observed in section above that communities appeared to behave in a segregative manner, this is not true when looking at politic opinions: in this sense, it might be observed that segregative behavior are observed community-wise, and mixed behavior are observed opinion-wise.

As fig.5.4 shows, when some trend is present, it is always disassortative. As we observed before, partisan based interactions are indeed disassortative on news datasets; this was also observed in politics datasets but only on some specific year - for example in 2016, which indeed is consistent with what we did in section 4.3 and with what was observed in previous works [6]. Furthermore, in politics datasets extremist left appeared interacting more with moderate left and vice-versa, which could be interpreted in a inter-community disassortativity.
To confirm disassortativity on news datasets, it can be seen that also the community of anti trumpist and conservatives appear to behave in that manner.

As is the case for sociodemographic based features, further research can be done to observe whether these trends could be topic-related; nonetheless, it is interesting to notice that assortativity is never present on those features and thus no opinion based echo chamber was observed.

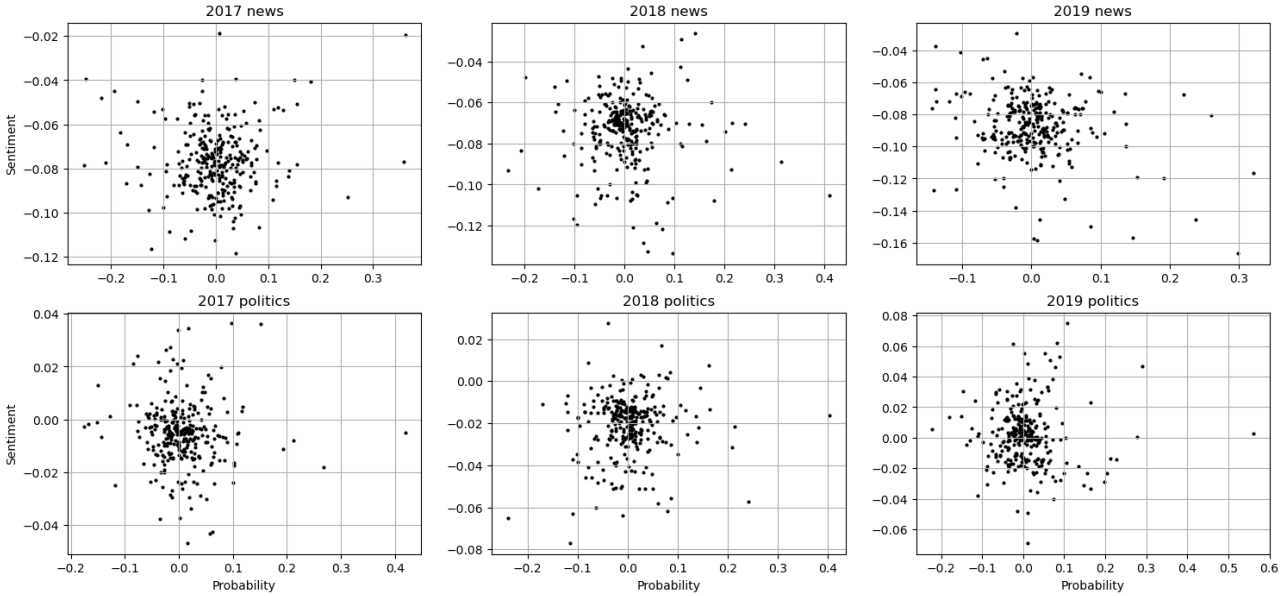

Figure 5.5: Relation between sentiment and probability, for interactions of every pair in different datasets. While horizontally centered on *probability* = 0, these figures shows high variance on these relations with no particular trend to be found.

5.4 Interactions' sentiment

Method described in section 3.3.1 provides an average sentiment for all possible feature pair: a value above 0 should indicate averagely positive interactions, and vice versa. However, these values appeared to be higly context dependent and eventually noisy, so meaningful information can be observed only when comparing these values between each other.

A sample of these results are reported in fig.5.6. The context dependency is given by the fact that for each year, in a given subreddit, these values move together; this whould imply there is a certain bias in each dataset, induced by something that apparently catalyzed overall sentiment toward certain values.

Nonetheless, in some cases there appear to be some interesting constant relation between these values: as the right-most graph in fig.5.6 shows, interactions between users labelled as rich are apparently the most positive ones, while interactions between users labelled as poor are the most negative; furthermore richer users' comments appear to be generally more positive then poorer's ones, hinting that language can be correlated with the features we found.
Something of the same fashion can be said about young-old average sentiment, where older people have a generally more positive language either on news subreddit or politics; interestingly, comment from old to young appear to be the most positive in subreddit 'politics', while old to old are the most positive in 'news'.

It is furthermore interesting to compare the sentiment trends with probability trends: while we found that assortativity was consistent in interactions of sociodemographic features, these interactions are not always positive. As an example, comments of low affluence users toward each other appear to be probable (as is shown in fig.5.6)

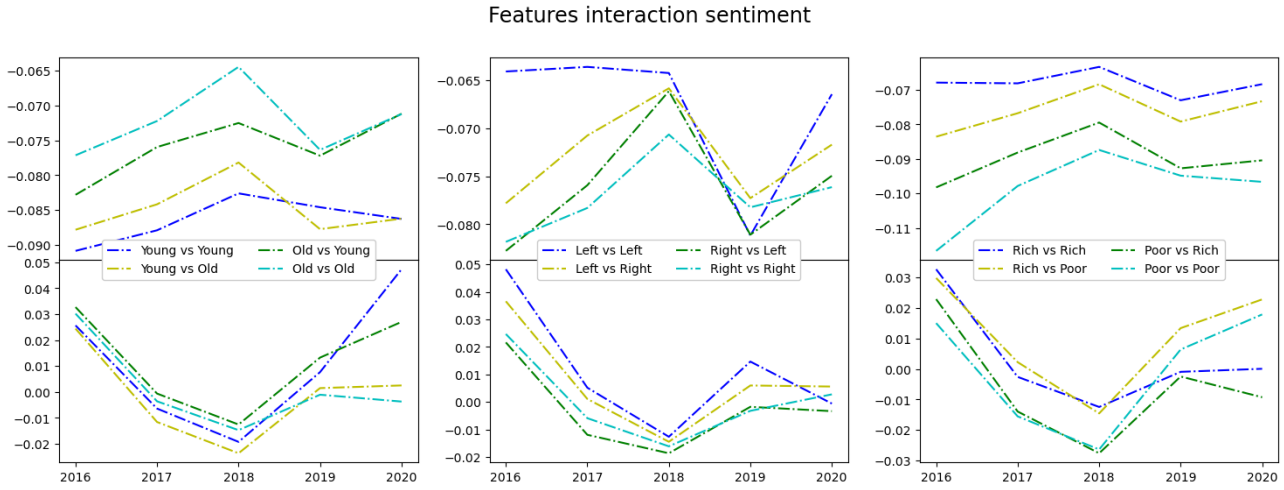

Figure 5.6: Interactions' average sentiment between users labelled with given features (First one in legend is author's feature, the other the target's); graphs above show these values measured on subreddit 'news', while ones below in subreddit 'politics'.

but indeed also more negative than the average.
When looking at a scatter plot of sentiment and probability of interactions for each pair, no clear trend can indeed be observed; this means that interactions can be driven either by a positive or negative sentiment, and no significant difference can be observed. By looking at fig.5.5, it could be argued that maybe a slight leaning toward negative sentiment/higer probability could be observed, which whould reinforce the narrative of interactions in the web being largely driven by hate and arguments [18]; however, since lot of noise was observed for either probabilities and sentiment estimate, more data and more thorough investigation would be required to corroborate these observations.

This method unfortunately provided higly noisy results on most of the features pairs, especially on features from clustering. This might imply that opinions are not a strong predictor of an interaction's sentiment, but more opinion specific data would be required to say if that is the case.

A meaningful result about opinion based interactions is instead given by average sentiment with respect to partisanship: as middle graphs in fig.5.6 show, sentiment in assortative interactions are more positive than sentiment of interactions with opposite partisanship users.
When looking at this result together with interaction matrices values (in fig.5.4) it is evident that while no echo chamber is observed, and activity is even disassortative in subreddit 'news', averagely negative sentiments tells us that opposite sided users are probably arguing and debating.

Chapter 6

Conclusion

Our study has provided insightful information about users' interaction tendencies on our datasets, and the model has proven to be reliable and interpretable.
Regarding our features extraction process, we managed to characterize users on different perspectives, working with either sociodemographic and opinion based features at the same time. Regarding the training process itself, we managed to find a simple yet fast and powerful method to see how these features interacted.

While the MCMC approach was too slow to be a viable solution, it was useful to observe how the posterior on the interaction matrix was shaped, proving that the Logit model was a powerful enough strategy. In combination with the proof of their equivalence, discussed in section 3.4.3, this motivated us to rely on the Logistic Regression alone, with the knowledge that no useful information was lost. The latter came with the advantage of p-values estimate and confidence intervals on the found interaction matrices, so that we were able to assess the statistical significance of our results.
The main advantage of our approach was being able to work with many features altogether, meaning each result on any feature-feature interaction could be considered stable with respect to many confounding factors. Together with the possibility to evaluate statistical significance, our approach thus provides great confidence when it comes to evaluate reliability of our observations.

The main finding in our work is that, in a place where there was observed to be significant polarization in users' opinions [3], no echo chamber was observed; in the context of subreddit "news" (and the same being true for the dataset "politics", although less consistently) we even found the opposite to be true: with high statistical significance (section 5.2) we found that users were interacting mainly with opposite-minded people, instead of the assortative-like behavior one would expect in echo chambers.
Providing an improvements with respect to previous works [6], this behavior seems to be indeed stable with respect to many other confounding factors, since the disassortative behavior with respect to partisanship could be observed when taking into consideration also sociodemographic characteristics and the possible involvement in specific opinion based communities. This result is in contrast with the narrative of social networks being a cause of echo chambers, and the hypothesis that polarization is mainly due to these.
The evaluation of interactions' sentiment in relation to the features, as discussed in

section 5.4 also suggests that these cross-group interactions, on the base of their partisanship, might consist in some sort of debate between opposite view. For this reason, in our case, we can make the hypothesis that a possible cause of polarization might be the repeated "negative" interaction with individuals who support different opinions. This is consistent with recent findings arguing how opinion reinforcement might be due to exposure to the opinion counterpart [18].

Our contribution on this topic is thus consolidating the narrative of polarization being due to more of a complex set of factors, not only echo chambers, obtaining this result with significant confidence. Furthermore, it appears that social networks do not always generate echo chambers, at least when interactions occur on neutral grounds - such as "politics" and "news" subreddits are- and that debate could be an important driving factor of users' social habits.
Looking at the whole set of features, a more complex scenario can indeed be observed:

- Interactions based on partisanship were mainly disassortative.
- Sociodemographic features consistently drove assortative interactions.
- People belonging to certain communities behaved in a segregative-like manner (although since we used only limited amount of data regarding these features, results related to community involvement did not come with high statistical significance).
- Interaction tendencies are context dependent

The first element of this list was indeed what led us to our observations regarding echo chambers. The last one is discussed in section 5.2.1 and suggests that "conventions" in a given context (in our case, subreddits' rules and topic) might play a relevant role in determine how individuals behave in social context.

On the matter of sociodemographic features, we found that users were leaning toward interactions with people of the same people age, gender and affluence. While appearing intuitive, it is not obvious why this actually happened: an hypothesis is that different topics could attract users with different sociodemographic extraction. To verify this hypothesis, we are performing text bases topic extraction on our dataset, trying to find correlations between features and users involvement; this process is introduced in section 3.3, but still a work in progress at the time of writing.

Regarding the features that represent users' involvement in different communities (and supposedly share some specific opinion, as discussed on 3.2.2), we observed how users belonging to certain communities were mostly interacting within each other. That is the case, for example, of communities like gun enthusiasts, lgbt supporters and environmentalist, as can be seen from the diagonal values in the matrices of 5.2. While this might suggest some sort of topic related echo chamber, these features were observed in only $\sim 10^2$ users among our whole dataset; for this reason we could not neither exclude that a bias was present in our dataset, nor the model was able to find find statistical consistency on the training results. This finding, nonetheless, would suggest that some interesting research could be performed in the matter of how echo chambers shape up in relation to different opinions: assortative behavior was here

indeed observed not on "wide" opinions (like left and right leaning) but on a set of more specific ones.

The latter is indeed one of the limitation of our approach: to have a broader set of features, it is not granted that each community/opinion is well enough represented in our dataset. Furthermore, since we needed to extract each feature from raw data, each feature comes with concerns about its interpretation and possibly some ambiguity; this problem is addressed in section 3.2 where it is discussed how this limitation can be solved, or at least taken into account. Another limitation is due to the fact that, as observed in section 5.2.1, the interactions are context dependent: while we studied and compared datasets coming from two different subreddits, it would thus be interesting to apply our model to many more, and see which of our findings remains consistent in different contexts, and which one might be related to it. The whole preprocessing and training processes, however, could readily be applied to different subreddits and year, so this one could be an interesting point for further research.

In conclusion, despite the difficulties we encountered and hints of further work to be done, with our work we depicted a complex and interesting picture of how people behave in online social interactions. The results on political based echo chambers are interesting on their own, providing some insight for the debate of the role of social networks in favoring debate instead of echo chambers.
Furthermore, looking at this picture as a whole and taking this work as an exploratory approach to behavior in general, we saw how the variety of observed interactions can suggest many other questions for further research. While any work on social interactions will be always arguably subject to interpretations and ambiguity, testing different methodology, validating the process and carefully evaluating the consistency of any finding -as we just did with our work- has proven to be an effective approach in the difficult task of understanding human behavior.

List of Figures

2.1 Different performances of link prediction models on a subset of the most active 234 authors of NIPS 1-17 cohautorship dataset; these pictures are taken from "Nonparametric Latent Feature Models for Link Prediction"[12]. In each of them, the rows and columns represent an author of scientific articles, and the citation from one author to another is depicted as a white pixel in the matrix. The figure on the left represent the true observed collaborations; figure on the right represents the extimated collaboration probabilities with an instance of SBM method; figure in the middle is the result of a generative model approach based on MCMC learning algorithm. 7

2.2 A visual representation of how each element of the chain is generated, taken from "A conceptual introduction to Markov chains Monte Carlo methods"[10]. The three steps of sampling strategy are depicted: initial proposal of the next sample through $Q(M_{i+1}|M_i)$, the computation of the transition probability $T(M_{i+1}|M_i)$ and the subsequent decision to accept or reject the proposed new sample. 9

3.1 Pipeline of features extraction with mentioned different strategies. Color blue indicates that the same file/script is used for different datasets; color dark green stands for dataset dependent files; light green stands for datasets/cutoffs/preprocessing dependent files; red nodes are the cutoffs and diamond shaped ones are python scripts. Squared shaped nodes represent files generated in the process: the first line is the name of the file (where {y}_{s} will take the values of year and subreddit of chosen context) while second line tells what data is stored in that file. 14

3.2 Subreddit scores distribution along polar axis with different meaning, provided by data of [3]. On x axis are the possible values of leaning (ranging in $[-1, 1]$ divided in 100 bins), on y axis the count of how many subreddits fall in that bin represented in logaritmic scale 18

3.3 Users' involvement distribution in 2020 on subreddit 'politics'. Involvement is the total number of a user activity in that context (sum of number of its submissions and comments), and y-axis in this histograp represent the amount users with that involvement value. 19

3.4 Coherence of submissions' title representation, with respect to number of topics in NMF. Value of 12 was chosen by visual inspection of this graph, searching for a value where curve abruptly slows its growth, but still provide a good $Coherence/N_{topic}$ ratio 22

4.1 Distribution of users' political score through embedding scores, distinguishing the ones that were labelled as leaning toward left and right in [6]. As these graphs show, the two strategy are coherent and provide a good separation between left and right users; overlapping has been calculated to be over 90%. While partisan is the left/right leaning itself, partisan B represent the leaning toward left candidate (Clinton, labelled as D for democrat) and right candidate (Trump, labelled as R for repubblican); overlapping between partisan and partisan B has also been calculated to be over 90%). More details about these features in section 3.2.1 and in the original work where they are defined [3] 34

4.2 The process used to generate a "realistic" interaction matrix. W_{bias} is the matrix of the $B_{i,j}$ values introduced to simulate assortativity or disassortativity in the interactions; the noise is a matrix of i.i.d. values sampled from a Normal distribution with standard deviation of 0.1 . . 39

4.3 The results on the training performed on 10^4 and $P_{neg/pos} = 1$, where the true interaction matrix is W_{true} in fig.4.2. As can be seen, the trained interaction matrices were basically the same as the true one 41

4.4 Comparison between W_{true} (on x axis) and model's estimated W_{ext} (y axis); on the left the feature matrix estimated through Logit model, on the right the one estimated with MCMC 41

4.5 ROC and PRC curves on positive interactions (pos pairs), negative interactions (neg pairs), both of them and on pairs outside the training dataset (test pairs). These graphs shows model's inability to achieve the maximum possible performance (discussed in section 4.2.3), regardless of bein in/out of samples and of the true interaction outcome 42

4.6 On either graphs, the x axis represent the chosen proportion of negative sampling $P_{neg/pos}$; on the left the y-axis represents the values of estimated bias W_0, while on the right the ROC auc of the corresponding model (in term of distance from the maximum possible ROC auc, which is the one achieved by true probabilities). These graphs are evaluated on Logistic regression model alone 43

4.7 Sample chains generated by MCMC process (above), and their average (below). The values of convergence of these parameters are the ones of interaction matrix in fig.4.3, where similarity with Logit estimated interaction matrix is evident. 43

4.8 Interaction matrices obtained with on data from "No Echo in the Chambers of Political interactions on Reddit"[6]. Figure on the left shows the results using only Logit model and only political labels ("C" stands for Clinton supporter, "T" Trump supporter); figure on the right was instead obtained using MCMC model, considering also features of involvement (activity) and popularity (score). From the top left partition of the second matrix we can see how the disassortative behavior is persistent when using these confounding factors; Overall, all features interaction are coherent with the expectation given by [6]. 45
4.9 MCMC chains and their average when training on political labels alone. Gelman-Rubin statistic was around 1.2 for each of these parameters, meaning the convergence was decent but more samples might have been required - as can be seen from the graphs on the right. Nonetheless the average displayed clear disassortative behavior, and since this training was performed on a relatively small subset of dataset this is indeed coherent with our expectations 46
4.10 MCMC chains and their average when training on both political labels and confounding factors (users' activity and popularity). The first and second lines of graphs shows the chains on model's bias W_0 and interaction matrix W. The last line of graph shows, among all these values, the ones regarding interactions of political labels - which against show systematic disassortative behavior. The final values of these chains are the same reported in right figure 4.8 47
4.11 The two matrices on top shows the probability that and interaction will be observed $P(Y = 1|C_a^i, C_t^j)$, taking an author and a target whose activity/popularity measure fall in bins i and j respectively. The second row shows the same matrices normalized by rows $\frac{P(Y=1|C_a^i,C_t^j)}{\sum_k P(Y=1|C_a^i,C_t^k)}$, representing the probability that taking a comment of an author whose feature falls in i bin, the target's feature will fall in j bin. These pair of matrices show the expected results (which are more evident in the normalized ones): assortativity with respect to activity, and disassortativity with respect to popularity. 49
4.12 The ROC and PRC measure on MCMC model's estimated interaction'probabilities. Low values of the ROC auc tells us that the model can't be considered good in prediction task itself; this is nonetheless typical of link prediction task based on limited amount of features, meaning these features can't be predictive enough to make accurate estimation on individual interactions. As it is discussed above, this strategy instead aim at mostly finding statistical leanings in users' habits, which are for example the assortative or disassortative behavior and/or no-echo chamber phenomenon discussed in this section 50

4.13 Posterior estimated distribution with MCMC, for every single pair used in testing against previously known results (section 4.3. These distribution can roughly show how the whole posterior itself is indeed approximately denser around its average, with pretty regular center field like topology and no other local maximum to stand off. 53

5.1 Interaction matrices on 2016 and 2020 datasets, where non statistically significant values (according to criteria defined in section 5.2)are represented in white. Features extracted through the clustering strategy are the ones with prefix 'c_' 57
5.2 Interaction matrices' values that exhibited persistent assortativity (red) or disassortativity (blue) on different groups of datasets. Matrices above are obtained without any requirement on each value statistical significance, while the ones below are filtered by requiring that these values must have statistical significance in each observed year (according to criteria defined in section 5.2). 59
5.3 Feature matrix values regarding interactions of sociodemographic features, together with their confidence intervals. Graphs on top shows interactions on subreddit 'news', ones below on 'politics'. 62
5.4 Feature matrix values regarding interactios of opinion based features, together with their confidence intervals. Graphs on top shows interactions on subreddit 'news', ones below on 'politics'. 63
5.5 Relation between sentiment and probability, for interactions of every pair in different datasets. While horizontally centered on $probability = 0$, these figures shows high variance on these relations with no particular trend to be found. 64
5.6 Interactions' average sentiment between users labelled with given features (First one in legend is author's feature, the other the target's); graphs above show these values measured on subreddit 'news', while ones below in subreddit 'politics'. 65

List of Tables

3.1 Datasets statistics before and after applying cutoff to users' activity. *N all nodes* is the total number of users observed in that year and subreddit, while *N all edges* is the total number of interactions (comments). *N nodes* and *N edges* are the remaining users and interactions after C_{local} and C_{global} cutoffs are applied to their activity. Training will be performed on those, sub sampling up to a maximum of 10^6 edges. . . . 16

3.2 Clusters of opinion related subreddit, found with DBSCAN; each entry here represent the feature we associated with it, and the subreddit in the each cluster. Each of this cluster can thus be considered a community, possibly distributed in different subreddits, where the members share some sort of opinion 20

4.1 Subreddit with higest log odds (eq.4.1) of features extracted with embedding based strategy described in section 3.2.1 32

4.2 Subreddit with higest log odds (eq.4.1) of features extracted with clustering strategy described in section 3.2.2 33

4.3 Statistics over MCMC estimated posteriors over each parameter; In particular are here reported averages, standard deviation and Gelman-Rubin statistics (r_hat). This table also shows how median value is often really close to average value (hinting a denser posterior around its center), and MCMC efficacy in exploring parameter space. N_{eff} stands for number of effective samples, to compare with $N_{samples} = 100$ which we chose: higher values tells MCMC efficiency in exploring the posterior 51

5.1 Each value in this table represent number of observed nodes possessing both row and column feature. Features labelled as "(c)" are the ones obtained from clustering method. 55

5.2 Co-occurrences in observed edges. Each value represent the number of observed edges where the author has the row feature and target has column feature. 55

5.3 Correlation between features, shown as a percentage and approximed to its closest integer value. Each value is the probability that if a user has the feature indexed by row, it will also have the one indexed by column 56

5.4 The values in this table represent the number of values that differ on sign between each pair of interaction matrices. Lower values (depicted with darker reds) are thus representative of more similar matrices . . . 58

5.5 The same values of table 5.4, ordered by year and divided by subreddit 58

Bibliography

[1] A. Edelmann, T. Wolff, D. Montagne, and C. A. Bail, "Computational social science and sociology," *Annual Review of Sociology*, vol. 46, no. 1, pp. 61–81, 2020.

[2] C. Monti, G. D. F. Morales, and F. Bonchi, "Learning opinion dynamics from social traces," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ACM, aug 2020.

[3] I. Waller and A. Anderson, "Quantifying social organization and political polarization in online platforms," *Nature*, vol. 600, 2021.

[4] N. Gozzi, M. Tizzani, M. Starnini, F. Ciulla, D. Paolotti, A. Panisson, and N. Perra, "Collective response to media coverage of the covid-19 pandemic on reddit and wikipedia: Mixed-methods analysis," *J Med Internet Res*, vol. 22, p. e21597, Oct 2020.

[5] M. Cinelli, G. D. F. Morales, A. Galeazzi, W. Quattrociocchi, and M. Starnini, "The echo chamber effect on social media," *Proceedings of the National Academy of Sciences*, vol. 118, no. 9, p. e2023301118, 2021.

[6] D. F. M. Gianmarco, M. Corrado, and S. Michele, "No echo in the chambers of political interactions on reddit," *Scientific Reports*, vol. 11, 2021.

[7] K. Garimella, G. D. F. Morales, A. Gionis, and M. Mathioudakis, "Political discourse on social media: Echo chambers, gatekeepers, and the price of bipartisanship," 2018.

[8] D. Baldassarri and A. Gelman, "Partisans without constraint: Political polarization and trends in american public opinion," *American Journal of Sociology*, vol. 114, no. 2, pp. 408–446, 2008.

[9] M. Zuckemberg, "Building global community," 2017.

[10] J. S. Speagle, "A conceptual introduction to markov chain monte carlo methods," 2019.

[11] C. Monti and P. Boldi, "Estimating latent feature-feature interactions in large feature-rich graphs," 2017.

[12] K. Miller, M. Jordan, and T. Griffiths, "Nonparametric latent feature models for link prediction," in *Advances in Neural Information Processing Systems* (Y. Bengio, D. Schuurmans, J. Lafferty, C. Williams, and A. Culotta, eds.), vol. 22, Curran Associates, Inc., 2009.

[13] K. Garimella, G. D. F. Morales, A. Gionis, and M. Mathioudakis, "Quantifying controversy on social media," *Trans. Soc. Comput.*, vol. 1, jan 2018.

[14] J. T. Klapper, "The effects of mass communication.," 1960.

[15] C. G. Lord, L. Ross, and M. R. Lepper, "Biased assimilation and attitude polarization: The effects of prior theories on subsequently considered evidence.," *Journal of personality and social psychology*, vol. 37, no. 11, p. 2098, 1979.

[16] F. Baumann, P. Lorenz-Spreen, I. M. Sokolov, and M. Starnini, "Modeling echo chambers and polarization dynamics in social networks," *Physical Review Letters*, vol. 124, no. 4, p. 048301, 2020.

[17] A. Cossard, G. D. F. Morales, K. Kalimeri, Y. Mejova, D. Paolotti, and M. Starnini, "Falling into the echo chamber: the italian vaccination debate on twitter," 2020.

[18] C. A. Bail, L. P. Argyle, T. W. Brown, J. P. Bumpus, H. Chen, M. F. Hunzaker, J. Lee, M. Mann, F. Merhout, and A. Volfovsky, "Exposure to opposing views on social media can increase political polarization," *Proceedings of the National Academy of Sciences*, vol. 115, no. 37, pp. 9216–9221, 2018.

[19] A. Guess, B. Nyhan, B. Lyons, and J. Reifler, "Avoiding the echo chamber about echo chambers," *Knight Foundation*, vol. 2, pp. 1–25, 2018.

[20] T. P. Peixoto, "Bayesian stochastic blockmodeling," nov 2019.

[21] M. D. Hoffman and A. Gelman, "The no-u-turn sampler: Adaptively setting path lengths in hamiltonian monte carlo," 2011.

[22] D. Vats and C. Knudson, "Revisiting the gelman-rubin diagnostic," 2018.

[23] Wikipedia contributors, "Variance inflation factor — Wikipedia, the free encyclopedia," 2022. [Online; accessed 27-May-2022].

[24] "List of active bots, provided by reddit."

[25] C. Hutto and E. Gilbert, "Vader: A parsimonious rule-based model for sentiment analysis of social media text," *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 8, pp. 216–225, May 2014.