Party Prediction for Twitter

Kellin Pelrine[1,2], Anne Imouza[1,2], Zachary Yang[1,2], Jacob-Junqi Tian[1,2], Sacha Lévy[1,2], Gabrielle Desrosiers-Brisebois[2,3], Aarash Feizi[1,2], Cécile Amadoro[3], André Blais[3], Jean-François Godbout[2,3], Reihaneh Rabbany[1,2]

[1]McGill University
[2]Mila - Quebec AI Institute
[3]Université de Montréal

Abstract

A large number of studies on social media compare the behaviour of users from different political parties. As a basic step, they employ a predictive model for inferring their political affiliation. The accuracy of this model can change the conclusions of a downstream analysis significantly, yet the choice between different models seems to be made arbitrarily. In this paper, we provide a comprehensive survey and an empirical comparison of the current party prediction practices and propose several new approaches which are competitive with or outperform state-of-the-art methods, yet require less computational resources. Party prediction models rely on the content generated by the users (e.g., tweet texts), the relations they have (e.g., who they follow), or their activities and interactions (e.g., which tweets they like). We examine all of these and compare their signal strength for the party prediction task. This paper lets the practitioner select from a wide range of data types that all give strong performance. Finally, we conduct extensive experiments on different aspects of these methods, such as data collection speed and transfer capabilities, which can provide further insights for both applied and methodological research.

Introduction

Inferring the political orientation of social media users is a common first step in a wide range of studies. Multiple works on understanding the impact and the spread of misinformation consider the political ideology of users (Osmundsen et al. 2021; Lawson and Kakkar 2022) in different contexts to determine, for example, how political ideology can predict amplifying fake news originating from trolls (Badawy, Lerman, and Ferrara 2019). Another critical application of party prediction is for measuring political conflicts in online societies (Green et al. 2020; Conover et al. 2011), and for devising strategies to mitigate polarization's harms (Voelkel et al. 2022; Ruggeri et al. 2021; Bail et al. 2018; Warner and Villamil 2017). Indeed, numerous studies have confirmed that partisan animosity and polarization has increased in several countries, such as the United States (Tucker et al. 2018), Canada (Johnston 2019), the United Kingdom (Hobolt, Leeper, and Tilley 2021), and France (Boxell, Gentzkow, and Shapiro 2022), with researchers identifying social media as one of the root cause of this phenomenon (Kubin and von Sikorski 2021; Banks et al. 2021; Kim and Kim 2019; Heiss, von Sikorski, and Matthes 2019; Törnberg 2022). However, drawing robust conclusions about the impacts of political ideology and how it changes requires that researchers correctly identify the partisan preferences of social media users. While this is a crucial and extensively studied topic, there is currently no widely accepted or definitive method for predicting an individual's political party affiliation based on their online activity.

In this paper, we discuss how existing methods for predicting political party affiliation of social media users are generally based on several different types of signals, including content (e.g., text of tweets: Conover et al. 2011; Rheault and Musulan 2021; Colleoni, Rozza, and Arvidsson 2014. Or linked media: Luceri et al. 2019; Törnberg 2022), relationships (e.g., followership: Barberá 2015; Gu et al. 2016), and interactions (e.g., retweeting: Xiao et al. 2020; Badawy, Ferrara, and Lerman 2018). Most of the previous methods use a combination of these features, as summarized in Table 1 and continued with Table 10 in the supplementary material; they are also normally evaluated on original datasets associated with broader studies. The collection process and resulting data difficulty vary widely, with different types of users (e.g., politicians vs. general public), diverse filters, and often entirely different time periods and topics studied. This means that assessing the performance of these models in the literature is challenging, as there is usually no empirical comparison done on the performance of the utilized methods versus other existing approaches or baselines. Although this can be expected from larger systems, sub-optimal or uncertain party prediction performance can have a big impact on the downstream results and overall conclusions.

We address these shortcomings by conducting a comprehensive survey of existing party prediction methods, followed by a thorough evaluation through an extensive array of experiments. The goal of this comparison is to find the most effective and practical methods, so we go beyond mere *accuracy* and consider also *coverage* and *cost*. Coverage measures how many users we can classify with each method, whereas cost focuses on how much resources are required to acquire the relevant data for this particular method. For example, requiring only tweet stream vs. full followership network: the latter is commonly used, but it requires more extensive data collection while not performing better than less data intensive alternatives, as we discuss in the experiments.

In order to conduct this analysis, we collected a rich dataset of approximately 14,000 Twitter users who discuss American politics during the time period directly before and after the 2020 US election. This dataset includes 8 distinct signals which enables a unified comparison to test the performance of different methods. We also propose our own approaches based on label propagation, GCN (Graph Convolutional Networks), GAT (Graph Attention Networks), HAN (Heterogeneous Graph Attention Networks), and RoBERTa (a transformer-based language model), and show that these methods deliver strong performance and can do so from a wide variety of data types. In summary, the main contributions are:

- To provide a thorough comparison of different methods for party prediction, including the four current state-of-the-art methods as well as thirty-four newly proposed methods which rely on different combinations of text, activity and relation signals.
- To show that our proposed label propagation on retweet activities has a strong accuracy, coverage and speed, while its data acquisition is also the most efficient. The coverage of this method can be improved by combining more signals and/or using more complex methods.
- To open up new options and data types for applied practitioners and new insights for future methodological research. For example, in our experiments on Canadian data the followership information was not available, hence the party prediction was performed using methods that rely on retweets and mentions. Having more options like these will be increasingly important as Twitter data access becomes more uncertain and some data types become inaccessible or more costly to collect.

Ultimately, our experiments should help researchers better understand the myriad of factors involved in this task, such as which data types are most informative and applicable, how to structure interaction data, and differences between the accounts of general users and those of politicians. Our code and supplementary materials are available at https://github.com/anonymouspartyprediction/partypred/.

Background and Related Work

Party prediction is a common approach used to classify social media users according to their partisan affiliations. In the United States, for example, users can either be supporters of the Democratic or the Republican parties; in some cases, users can also be classified according to their ideology, as liberals or conservatives. Although extensively used (Lawson and Kakkar 2022; Badawy, Lerman, and Ferrara 2019; Green et al. 2020; Conover et al. 2011; Voelkel et al. 2022; Ruggeri et al. 2021; Bail et al. 2018; Warner and Villamil 2017), there is no widely accepted consensus among scholars on the right approach to conduct this classification task. Most existing methods have been applied to Twitter data due to both the ease of access for researchers, as well as its mainstream use for political discourse. In this study, we also use Twitter data to compare methods for party prediction. However, the conclusions could transfer to similar media platforms.

There is currently a broad range of estimation techniques applied for party prediction on Twitter. Some party prediction approaches use media outlets shared by the users to infer partisan affiliation (Rheault and Musulan 2021; Luceri et al. 2019; Stefanov et al. 2020; Badawy, Ferrara, and Lerman 2018). Others rely on the structure of networks to infer partisan leanings (Barberá et al. 2015; Colleoni, Rozza, and Arvidsson 2014; Pennacchiotti and Popescu 2011; Gu et al. 2016; Xiao et al. 2020; Havey 2020; Wojcieszak et al. 2022), while some rely instead on other types of network activities, such as retweets and mentions (Conover et al. 2011; Rheault and Musulan 2021; Luceri et al. 2019; Pennacchiotti and Popescu 2011; Gu et al. 2016; Badawy, Ferrara, and Lerman 2018; Xiao et al. 2020; Jiang et al. 2021). Finally, several approaches focus on the content of messages, by either looking at the text or the hashtags used to represent a partisan side, like #Democrats or #Republican (Conover et al. 2011; Rheault and Musulan 2021; Colleoni, Rozza, and Arvidsson 2014; Pennacchiotti and Popescu 2011; Rodríguez-García, Herranz, and Unanue 2022; Mou et al. 2021; Fagni and Cresci 2022). Many works combine more than one feature to make their predictions (Makazhanov and Rafiei 2013; Boutet, Kim, and Yoneki 2012; Wang et al. 2017; Mou et al. 2021), while a few rely exclusively on a single feature in their party prediction model (e.g., Barberá et al. 2015).

We summarize some of these key methods proposed in the literature for party prediction in Table 1, based on what signals they use to infer the user's ideology as well as the distribution of the users they classified. For the former, we consider whether they use (column 2) media outlets (shared by the user in their messages) (3) network activities (who the user interacts with through retweets and mentions) (4) network relations (who the user follows) and (5) content (hashtags and keywords in their messages). We also look into the difficulty of users classified, by reporting the (6) type of user (politician or public), (7) number of users and (9) our estimated difficulty as explained below. Methods selected for inclusion in Table 1 represent a broad range of estimation techniques, and generally report high levels of prediction accuracy. Even so, these levels of accuracy fall between 66% and 97%, meaning that there is a wide range of classification error, depending on which approach is used. In the supplementary material we have a similar table for additional notable related works which self-report lower accuracy levels.

For comparing party prediction methods, it is important to consider if the classification task was performed within a subsample of "politicians" or the general "public"; that is, accounts from well-known politicians and parties (see for example Rheault and Musulan 2021; Barberá et al. 2015; Gu et al. 2016; Xiao et al. 2020), or from a broader set of Twitter users, which can be harder to classify. Even when classification extends to the public, the way these users are sampled could impact the difficulty of the classification task (e.g., sampling only hyper-partisan or super active users which are easier to classify). Therefore, to aid comparison, we have provided an overall level of difficulty for each method's test

Methods	Accuracy	Media	Activity	Relation	Content	Type	Size	Code	Difficulty
Conover et al. (2011)	95		✓		✓	Public	1,000		Medium
Barberá (2015)	78			✓		Both	42,008	✓	**Hard**
Rheault and Musulan (2021)	91	✓	✓		✓	Poli.	505	✓	Easy
Luceri et al. (2019)	89	✓	✓			Public	38,000		Medium
Colleoni et al. (2014)	79			✓	✓	Public	10,551		Medium
Pennacchiotti and Popescu (2011)	89		✓	✓	✓	Public	10,338		Medium
Stefanov et al. (2020)	83	✓	✓		✓	Public	806		Medium
Gu et al. (2016)	66		✓	✓		Both	1,200		Medium
Badawy et al. (2018)	91	✓	✓			Public	29,000		Medium
Xiao et al. (2020)	**96**		✓	✓		Both	20,811	✓	**Hard**
Preoţiuc-Pietro et al. (2017)	**97**			✓	✓	Public	13,651	✓	Medium
Rodríguez-García et al. (2022)	82				✓	Poli.	12,600		Easy
Mou et al. (2021)	80			✓	✓	Poli.	735	✓	Easy

Table 1: Survey methods to predict party affiliation. Here we report for each paper: Accuracy, whether they used Media outlets, Network Activities (retweets and mentions), Network Relation (followership), and Content (words and hashtags); as well as if they consider Public, Politicians (Poli.) or both, their test size in terms of number of users on which the ideology is inferred, if their code is available, and finally the level of difficulty based on the type and size of data (elaborated in the text).

data by investigating their sampling approach. However, this level of difficulty is approximate and cannot give a precise, fine-grained comparison. This motivates our empirical analysis where we compare methods on the same dataset.

Almost all of the previous methods are tested on a sample of users who discuss political topics (i.e., datasets selected from policy-related keywords or hashtags or tweets of known politicians). However some apply additional filters (e.g., selection on polarizing keywords, user activity levels, or user location) as a factor that could potentially change the difficulty level. In our analysis, models tested exclusively on politicians are assigned an *"easy"* difficulty level since these types of users are known to be easier to classify (Cohen and Ruths 2013; Rodríguez-García, Herranz, and Unanue 2022). A *"medium"* level of difficulty implies that the method applies to a broader set of users from the general public, and contains more complex features (Conover et al. 2011; Luceri et al. 2019; Colleoni, Rozza, and Arvidsson 2014; Pennacchiotti and Popescu 2011; Stefanov et al. 2020; Badawy, Ferrara, and Lerman 2018; Gu et al. 2016). An example here would be Stefanov et al. (2020) who classify general public users based on the ideological leanings of the news articles they share on Twitter. For the classification task to be *"hard"*, the test size must also be large and validation must be done on both the public and an elite group of politicians. Two papers fall into this category (Barberá et al. 2015; Xiao et al. 2020), which we use later on in the experiments as baselines. Notably, the method proposed in Barberá et al. (2015), based on followership, is one of the most commonly used measure to estimate the levels of partisanship of users at the individual level (Brady et al. 2017; Jost et al. 2018; Pennycook et al. 2021). The overall accuracy of 78%, obtained when classifying a set of approximately 42,000 users whose party registration records were available, is not as high as other methods that combine more than one type of features. But this is a much harder classification task because there are few restrictions on the users in the dataset.

In our experiments, we examine these two models evaluated on "hard" data (highlighted in bold in the table), and also the approaches of Preoţiuc-Pietro et al. (2017) and Liu et al. (2022), given their performance levels, the availability of the code and that they rely on different features and data types. In particular, the method in Preoţiuc-Pietro et al. (2017) focuses on the use of a limited set of ideologically loaded political words. Because Preoţiuc-Pietro et al. (2017) test their model on different types of users, the general accuracy of the model varies greatly, ranging from 62.5% to 97.2%. The model performs best when the data is limited to users who follow politicians' account (either Democrats or Republicans). The POLITICS approach by Liu et al. (2022) also uses a content-based model to predict users' ideologies. Their Twitter user classification accuracy is low, under 50%, so they are not included in Table 1. But they only tested on 3-way classification (left, right, and center), which is difficult to compare with approaches tested on 2-way (most common) or other classification tasks. We hypothesized that this model would have strong performance on a two-way Twitter classification task (and as shown later, this is borne out in our experiments).

We explain how each of these four baselines are implemented in the supplementary material.

Methodology

Here, we first discuss how we sampled the data to compare different party prediction methods, and then present our own alternative approaches, which achieve on-par or better performance and can be used based on the type of signal available.

Data Collection

We curated 3 datasets, summarized in Table 2 and discussed in further detail below. In this table, the users are the authors of the posts, and the activity columns represent the total occurrences of their respective activity within these posts. For example, if the numbers of posts and retweets were the same, it would imply that every post was a retweet. The relations (friends and followers) were collected separately from the posts.

Dataset	Count		Activity				Relation			
	Tweets	Users	Hashtag	Mention	Retweet	Quote	Friends		Followers	
US Public	5.804M	14,112	1.535M	9.863M	4.397M	420K	15.452M	(3.700M)	10.619M	(4.050M)
US Politicians	156K	995	162K	212K	83K	54K	1.724M	(863K)	33.424M	(9.342M)
CA Public	11.361M	1.114M	-	15.128M	8.438M	-	-	-	-	-

Table 2: Statistics on the collected datasets, including the tweets, the activity extracted from them, and the relations of the users.

Politicians Data We collected all tweets, retweets and replies from 995 public and personal Twitter accounts of the United States representatives (433), senators (99), as well as vice presidential and presidential candidates (8) using Twitter's Search API.[1] We call this the US Politicians dataset. This data is collected from 2020-08-01 to 2021-01-17.

US Public Data We first collected around 1% of real-time tweets using Twitter's streaming API, that included one of the following US election related keywords: [JoeBiden, DonaldTrump, Biden, Trump, vote, election, 2020Elections, Elections2020, PresidentElectJoe, MAGA, BidenHarris2020, Election2020], from 2020-10-01 to 2021-02-28 . This created a dataset (not shown in table) with approximately 350 million tweets and 20 million users. Out of these, we sample 20 thousand users from those with keywords indicating their party affiliation in their profile description: [conservative, gop, republican, trump, liberal, progressive, democrat, biden]. To retrieve all the information for these selected users (the 1% collection process means missing many of a user's tweets), we then used a combination of the normal and academic Twitter APIs, to retroactively collect the full posts and activities of the 20K users during this period. Twitter does not allow access to tweets of deleted, protected, or suspended users. Thus, we were able to successfully retrieve the tweets of about 14k users.

Canadian Public Data We also collected data on discussion of the 2021 Canadian election, from 2021-07-31 to 2021-10-21. Similar to the US, we used a list of keywords (available in supplementary material) and collected 1% of real-time tweets containing those keywords. Unlike the US, we did not retroactively retrieve *all* tweets for the users in our sample, and examine more limited data types. It provides an illustration of some challenges one might face in applications, as well as additional experiments with a multi-party context.

Additional Data We further collect relation and like data on the users in our US datasets, as explained below and summarized in Table 2.

Relations We collected friends (accounts a given user follows) and followers for all accounts available at collection time. For US Public, followers were retrieved during May 2022 and friends between mid July and mid August 2022. For US Politicians, both friends and followers were retrieved during January 2021. Because some users have an extreme numbers of friends or especially followers that is not feasible to retrieve, we capped the total retrieved per user at 5,000 friends and followers for normal users. For politicians we capped followers at 100,000 and retrieved friends fully (maximum friends a politician has in our data: 135,389). Given multiple users can follow the same user, we also report the number of unique friends and followers in each dataset in parentheses in Table 2.

Likes We collected tweets liked by users in our US Public dataset in April 2022. We capped retrieval at 1000 maximum per user, leading to 5,260,616 total likes.

Data Labeling

To prepare our datasets for evaluation of different methods as well as for training our proposed approaches, we next need to label some of our users.[2] We first identify users with "Republican", "Democrat" keywords in their profile description. For "Republican" we use: [conservative, gop, republican, trump]. For "Democrat" we use: [liberal, progressive, democrat, biden]. We consider users as *potential* "Republican" ("Democrat") if the description contains at least one of the Republican (Democrat) identifiers and does not include any of the Democrat (Republican) identifiers. The rest of the users remain as "unknown".

Manual Labeling From this set, two political science graduate students who have studied US politics manually classified 1000 general public Twitter users from each party by looking at their biographical information. Their agreement was .796 Cohen Kappa. To further improve the quality, we asked a third political science student to resolve the disagreement cases. Finally, we dropped 70 cases that could not be determined (e.g., profiles that indicated someone apolitical or independent).

Weak Labeling In order to have a larger labeled dataset for training, we classify the rest of the users as Democrats and Republicans based on the full description they provide on their *user profile*. In particular, we use our manual labels to train a *profile classifier* to generate weak labels. This created a training set with noisy labels ($\sim 96\%$ accurate) that is used to train the classifiers introduced in the Proposed Methods section below, which do not rely on the user profiles and are based on activity and/or relations.

Before training this profile classifier, we first put aside approximately one third of users which have all the necessary

[1]Some Members of Congress have more than one social medial account (e.g., one personal and one official account). In this case, we collected information for all of the relevant accounts.

[2]We discuss here the process for US users, which was more complex. Canadian users were labeled with a similar process through the manual labeling step, and then those manual labels were used directly (further details are available in the supplementary material).

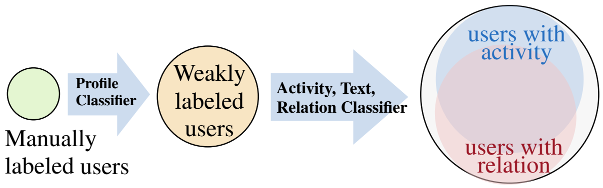

Figure 1: Starting from a small set of manually labeled users, we use the profile classifier to expand to a larger set of users (our US Public dataset), and then use that to build classifiers that can classify any user with text, activity, and/or relations.

data types to run every approach—our main test set. Then on the remainder we finetuned a RoBERTa-large (Liu et al. 2019) language model to predict the party each user is closest to from their profile description. We report the results in Table 3.

Dataset	Counts		Accuracy
	Rep.	Dem.	
US Politicians	101	66	98.8%
US Public	6,355	7,757	96.0%

Table 3: Number of users with explicit party/ideological keywords in their profile description, labeled by actual/manual label where available otherwise profile label classifier (on the left). Corresponding performance of our profile label classifier based on manually labeled sample of public users and actual politician parties (on the right).

We should highlight that the profile classifier, although accurate, cannot be effectively applied to users without explicit party indicators in their biographical information. This motivates methods that classify users based on other signals, e.g., who they follow or the text of their tweets. We illustrate the process of building from the small set of manually labeled users to increasingly general users in Figure 1. We next discuss how we go from users with profile classifier labels to more general ones.

Proposed Methods

Our data consists of graphs (activity and relation) and text (tweets). Here, we explain the preprocessing and models used for each data modality, namely: Graph Convolutional Networks (GCN) (Kipf and Welling 2017) a simple graph neural network model; Graph Attention Networks (GAT) (Veličković et al. 2018) a somewhat more complex one; Heterogeneous Graph Attention Networks (HAN) (Wang et al. 2019b) a heterogeneous graph one; Label Propagation (Zhu and Ghahramani 2002) a non-neural graph method based on "propagating" label information to neighboring nodes in the graph; and RoBERTa (Liu et al. 2019) a bidirectional transformer-based language model. These methods themselves are not new, but they have not been previously tested for this task and/or with the range of data types we examine.

Graph Data We structure the relationships between users as graphs, where the nodes are users and edges represent interactions between them. We consider two perspectives on which interactions link people together:

- Direct links: person i mentions or retweets person j. This corresponds to the graph's adjacency matrix A, a (0,1)-square matrix, where $A_{ij} = 1$ if j links to i.
- Projected (indirect) links: persons i and j both mention the same person k, even though i and j may not mention each other directly. Formally, this corresponds to the projected adjacency matrix $A' = A^T A$ and represents co-activity.

Direct links are intuitive, however, we find empirically that projected links are more informative for our GCN approach. Therefore, aside from a supplementary experiment that explicitly tests this, all our *GCN* models use the projected graph which represent similarity of activities between users. On the other hand, we see empirically that the direct graph performs better than the projected one for *GAT*, *HAN*, and *Label Propagation*. So in these cases, we always report the performances on the direct graph, again excepting the experiment explicitly investigating this.

GCN Our next step here is to construct one embedding per user and interaction type—i.e., a learned vector representing each user's realized interactions of that type (e.g., retweet, followership). We use the titular GCN (Kipf and Welling 2017), which is a simple yet powerful graph representation learning model. We generally use a single-layer version that is semi-supervised, with an unsupervised link prediction task and a supervised node classification task using profile classifier labels on training nodes.

This model supports using node features as well as the graph structure. We conducted preliminary experiments on using text embeddings for this, but found it did not improve performance. This parallels (Xiao et al. 2020), which likewise found that node features did not help their GNN-based model for party prediction. We instead use a uniform random vector with dimension 100 to initialize the embeddings.

To combine embeddings from different data types, we first train separate, independent GCNs on each individual input type, producing one embedding per user per GCN. We can then mix and match these data types as desired by concatenating the embeddings of the same user from different GCNs, before passing the combined embedding to a classifier. Here we use a random forest model to make a final prediction. In this way, we compare each data type individually and different combinations of them over multiple experiments.

GAT Our usage of GAT is similar to GCN, except we only use node classification (no link prediction) to train it. We follow the architecture and implementation of Huang et al. (2020).

HAN Our implementation of HAN is similar to GAT, but here we combine multiple graph types into a heterogeneous graph. Specifically, we combine the retweet, mention, quote, and hashtag graphs together into a single graph with those four edge types. We use the Deep Graph Library (DGL) implementation of HAN[3] which in turn is based on the original

[3]https://github.com/dmlc/dgl/tree/master/examples/pytorch/han

from Wang et al. (2019b). We match the hyperparameters to our GAT models.

Label Propagation This approach (Zhu and Ghahramani 2002) "propagates" labels between connected nodes (users). It is a more classical, non-neural method, and rather than an embedding, it directly produces a prediction. We use the semi-supervised version of this algorithm, i.e., seeding train nodes with profile classifier labels. Unlike the preceding approaches, because there are no embeddings, it is not straightforward for Label Propagation to incorporate multiple data types. However, if the data type is chosen carefully, this simple method achieves excellent performance with very high computational efficiency as shown in Tables 4 and 5.

Text Data Here, we use RoBERTa (Liu et al. 2019), a transformer-based (Vaswani et al. 2017) language model built on the foundation on BERT (Devlin et al. 2018), with changes to the pretraining process to improve performance. We examine two versions, RoBERTa-base (125M parameters) and RoBERTa-large (355M parameters), and consider both an untuned version using out-of-the-box embeddings classified by a random forest, and tuning them on training set users to classify tweets according to the tweet author's label, and making a final author (user) level prediction by majority vote.

Further Information Additional details on how these approaches were implemented are available in the supplementary material under "Supplementary Implementation Information."

Experiments

We begin by illustrating the necessity of accurate party predictions through a simple experiment to examine the impact on a downstream task. Next, we move to detailed comparison of the different methods and signals for the party prediction task.

The Impact of Party Prediction

We consider measuring political conflict, with a simple approach from the literature (Conover et al. 2011; Waugh et al. 2011): network modularity, corresponding to the intuition that better separated clusters reflects more *political polarization*.

We examine the measurement at different accuracy levels by taking our retweet graph data, restricted to users with manual labels, and simulating different accuracy levels by randomly swapping a percentage of the labels. The box here represents the range of reported accuracies in the literature from Table 1. We see that the accuracy has a large effect, with the lower end of reported accuracies producing roughly half the modularity of the upper end. Consequently, low or unknown accuracy can lead to seriously misleading measurements and conclusions. This is also a simple, nearly direct measurement—the effects on more complex approaches and downstream tasks could be even more problematic. With this motivation in mind, we next evaluate different methods to get the best predictive performance.[4]

[4]Because our data is relatively balanced and errors on each class matter equally in most applications, we continue to report accuracy as the performance metric. We also examined two versions of F1 (macro and weighted averaging) and found they yield similar results and conclusions.

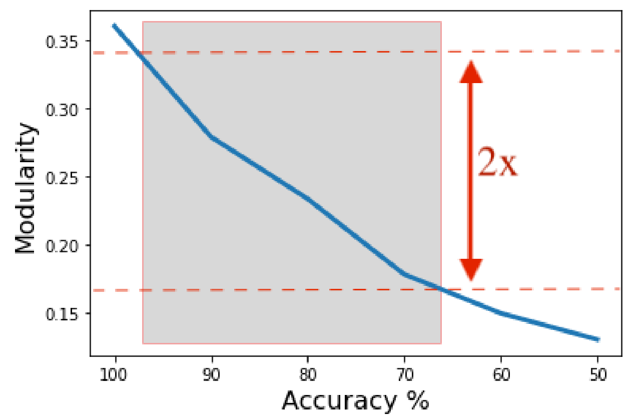

Figure 2: Imperfect accuracy has a big impact on polarization measurement, i.e., the downstream task. The box represents the range of accuracies reported in the literature (Table 1).

Comparing Accuracy of Methods

To compare different approaches equitably, we need a set of users that are active in all the particular ways that the different methods require. For instance, to compare Barberá (2015) (which is based on follow relations) with Preoţiuc-Pietro et al. (2017) (which is based on tweet text), we need users that both follow others and post tweets. We therefore require that all users in the test set have all the interaction types needed for every method we consider, to hold the test data constant across every approach. This requires filtering for users which have all 7 interaction types considered; it also requires that they follow a political elite user (e.g., politician) to apply the Tweetscores implementation of Barberá (2015) (we relax these conditions in subsequent experiments to include more cases in our analysis). In total 655 of our manually labeled users meet this criteria. We repeat our experiments 10 times with a random sample of 40% of these users as the test set (training each time on the fully separate profile-classifier-labeled users), and report mean and standard deviation. From the results reported in Table 4, we draw three main conclusions:

1. **Testing different approaches on the same users, like here, is necessary for clear comparisons.** For example, if judging only by the number reported in their original paper, Barberá (2015) would perform over 15 percentage points worse than in our experiments. Depending on which result is being considered, none of which are close to the findings reported here, Preoţiuc-Pietro et al. (2017) would perform anywhere from slightly better to over 30 points worse. The approach with one of the worst results for Twitter party prediction reported in the literature, POLITICS (Liu et al. 2022), turns out to have the best performance of all.

Type	Method	Accuracy	Rank
SOTA	Barberá (2015)	96.4 ± 0.6	15
	POLITICS (2022) Untuned	94.7 ± 0.9	24
	POLITICS (2022) Finetuned	97.8 ± 0.9	1
	Preoţiuc-Pietro et al. (2017)	92.4 ± 0.9	28
	TIMME (2020)	OOM	-
Text	RoBERTa-base Untuned	86.7 ± 1.0	35
	RoBERTa-base Finetuned	97.5 ± 1.0	3
	RoBERTa-large Finetuned	97.6 ± 0.8	2
Activity	Label Prop. Retweet	97.2 ± 0.5	4
	Label Prop. Mention	91.0 ± 0.9	30
	Label Prop. Quote	95.7 ± 0.8	20
	Label Prop. Hashtag	92.6 ± 0.7	26
	Label Prop. Like	95.7 ± 0.7	20
	GCN Retweet (RT)	96.2 ± 1.2	18
	GCN Mention (@)	85.6 ± 1.7	36
	GCN Quote (QT)	89.4 ± 1.1	32
	GCN Hashtag (#)	88.1 ± 1.8	34
	GCN Like	89.0 ± 1.6	33
	GCN RT+QT	96.3 ± 0.7	16
	GCN RT+QT+@+#	96.3 ± 0.7	16
	GAT Retweet (RT)	96.9 ± 0.4	6
	GAT Mention (@)	88.1 ± 1.3	34
	GAT Quote (QT)	93.5 ± 1.0	25
	GAT Hashtag (#)	91.4 ± 1.3	29
	HAN RT+QT+@+#	90.9 ± 8.4	31
Relation	Label Prop. Friend	96.5 ± 0.6	13
	Label Prop. Follow	96.1 ± 0.8	19
	GCN Friend	96.5 ± 0.7	13
	GCN Follow	92.5 ± 1.2	27
	GCN Friend+Follow	95.2 ± 1.1	23
	GAT Friend	96.7 ± 0.4	10
	GAT Follow	95.4 ± 0.8	22
Combined	GCN All	96.7 ± 0.7	10
	GCN All but Like	96.7 ± 0.7	10
	GCN All but Like+Follow	96.8 ± 0.8	8
	GCN RT+QT+Friend	96.9 ± 0.8	6
	GCN RT+Friend	97.1 ± 0.7	5
	GCN QT+Friend	96.8 ± 0.8	8
	GAT All but Like	95.7 ± 0.8	20

Table 4: Classification accuracy of different methods. Rank is shaded from blue (best) to orange (worst).

2. **Our relatively simple approaches deliver state-of-the-art performance.** 9 out of the top 10 approaches are ones we propose for the task and inputs here, with only a small gap between them.

3. **One can achieve strong performance with many different types of data.** There are methods from every category within less than 1.5% of the best method, meaning that one can get a strong prediction regardless of the category of data one has access to. Conversely, if deciding what data to collect, one can choose an efficient, scalable option (discussed in detail in subsequent experiments).

Besides those main conclusions, we note that TIMME produced out of memory errors on this scale of data. Additional discussion of our implementation, which uses the original study's code, is available in the supplement. We also note that HAN has a high standard deviation because one of the 10 runs gave very poor performance (66.0% accuracy). However, even with this run excluded, the average would be 93.6 ± 1.5, which would rank 25th. Finally, we note that although POLITICS Untuned is much better than RoBERTa-base Untuned, the gap after fine-tuning is slight. These are similarly-sized models—in fact the former is based on the latter. It seems most of the benefit of the special training used to produce POLITICS is erased when finetuned to this task, which is troublesome since finetuning seems necessary to get the best performance.

	Runtime (s)		
Method	Retweet	Friend	Text
Label Propagation	1	2	-
GCN, Direct Graph	483	1,871	-
GCN, Projected Graph	2,682	5,799	-
GAT, Direct Graph	754	9,195	-
GAT, Projected Graph	2,374	4,276	-
POLITICS (2022) Finetuned	-	-	24,370
RoBERTa-base	-	-	23,479
RoBERTa-large	-	-	55,701

Table 5: Runtime for 1 run of several approaches.

Comparing Cost of Methods

Here, we report the cost of methods in terms of computational time of the models as well the time required to retrieve the data they need.

Computation Time Table 5 shows runtimes (on 1 RTX8000 GPU) for several types of approach. We see that label propagation is multiple orders of magnitude faster than the GNN approaches. Considering label propagation also achieves very good accuracy in all our experiments, this is a very strong approach.

We also note that the direct graph is generally faster than the projected graph, except for the friend graph with GAT. The number of nodes of the projected graph is fixed at the number of users of interest; only the edge density changes, and that cannot go beyond the complete graph. The direct graph though can scale arbitrarily in both number of nodes and edges. Thus at moderate graph sizes it seems the density of the projected graph requires more time to process than having more nodes with the direct graph. But with the larger friend graph (direct version has over 1.1 million nodes), the crossover point is reached for GAT and the projected version becomes faster. Finally, we saw previously that the text-based approaches (RoBERTa and POLITICS) had strong performance, but they require much longer to train and finetune than any other option. So their applicability may be situational. Without that finetuning, their performance is very weak.

Data Retrieval Time Another critical factor for efficiency is how long it takes to retrieve the data. The limiting factor here is generally the Twitter API, which imposes caps per 15 minutes. The differences are summarized in Table 6.[5]

[5]Future changes to the Twitter API since this study was conducted may affect the exact numbers here. We advise future researchers to use the analysis methodology here as a guide that can be adapted to their individual data retrieval constraints.

We see that Likes[6] have a very harsh limit. Their predictive performance is also ordinary, therefore this data type is not advised unless one is already retrieving them for another purpose. Relations[7] (i.e. friends and followers) can be retrieved in much greater number per request, 5000, leading to a ten times larger total than Likes. They do have good performance in our experiments. However, each request can only retrieve a single user. So if one has a large number of users with few relations each, the small 15 requests per 15 minutes may be a binding factor. We show this in practice in the following experiment. Finally, the limits for tweets[8]—and therefore activity contained within tweets (i.e. retweets, mentions, quotes, and hashtags)—are the most generous. Even more can be retrieved if using the general streaming API instead of retrieving tweets for particular users. Tweets also have strong coverage of users, discussed in the appendix. Since we also have strong methods for this type of data (e.g. label propagation on retweets), it may often be the optimal strategy.

Data Type	# per R.	R. per 15m	Total per 15m
Tweets	200	900	180,000
Likes	100	75	7,500
Relations	5,000	15	75,000

Table 6: Retrieval limits per request (R.) and per 15 minutes for different data types

Comparing Politicians & Public

In this experiment, we consider both learning from and predicting the party of politicians. Learning from politicians has a theoretical advantage because their parties are officially known, removing the need for any additional labeling process. Meanwhile, on the evaluation side, they provide an additional set of data with even more definitive labels than expert-labeled general public users.

We show results in Table 7. Here the first three columns consider the task of predicting party affiliation for the general public, and compare training on the public itself, politicians, or both. The remaining three columns examine the same training scenarios but evaluated on the politicians themselves. Some additional implementation notes are given in the supplement.

Focusing first on the different interaction types (rows), we see that the best performance for politicians comes from the Friend relation (trained on politicians). This suggests politicians are especially consistent in their friends—i.e., who they follow. Considering the current strong partisan divisions in the US (Iyengar et al. 2019; McCarty, Poole, and Rosenthal 2016), this agrees with the intuition that politicians may avoid following people from the opposite party in order to prevent an appearance of mixed loyalties. Meanwhile, again when trained on politicians, quote performs the worst. This could be due to politicians frequently using quotes not only to support fellow members of their party but also to rebut opponents, making this relation more challenging to learn from in this context.

Considering next the differences between datasets (columns) and results overall, performance on US Politicians is not uniformly better than on US Public, at least if the training data includes politicians (fifth column, highlighted). There is also a small but consistent improvement to predicting the public purely from adding politicians to the graph, without using their labels (first column, highlighted, compared with Table 4). Adding their labels in fact tends to hurt public prediction (third column). This may be because politicians are very well connected in the graphs, and some politicians may be targets of opposition discussion to an extent that it helps to allow the opposition label to more easily propagate through them.

We also see that training on one user type and predicting another (second and fourth columns) can still yield satisfactory results, even though it's more challenging than training and predicting on the same type of user (first and fifth columns, highlighted). For example, one can train on politicians and get over 95% accuracy on the general public using retweets, friends, or follows. This partially matches prior literature that there is some performance lost transferring from one type of political user to another (Cohen and Ruths 2013), but the extent of the loss is much smaller. Thus, future data and methods along these lines may enable practitioners to learn strictly from the accounts of politicians, without needing labels on the general public that are much more difficult to obtain.

Classifying Canadian Users

For our final experiment, we turn to Canadian politics. This illustrates challenges one might face in applications. First, the data types we had available are limited; for instance, we do not have friend or follower data. Second, we do not have all each users' tweets, only the ones collected in the 1% real-time keyword-based sample. Third, due to time and computation limits, we examine a smaller selection of methods.

In addition, unlike the US where two parties dominate, Canada has a multi-party system. Based on the labeled data available, we examine 5-way classification [parties: Green Party (GPC), New Democratic Party (NDP), Liberal Party (LPC), Conservative Party (CPC), People's Party (PPC)]. For this and the above reasons, the task is significantly more difficult.

We report results in Table 8. The test set is the same regardless of input data type used, similar to Table 4. We see a large drop in performance compared to US data. Approximately 10% of this can be explained due to the finer-grained 5-way classification task: if we group parties into relatively left (GPC, NDP, LPC) and right (CPC, PPC), performance increases by over 10%, e.g. Label Propagation RT+@ gives 87.2% performance. The rest is likely due to differences in the amount of data collected per user and general differences between US and Canadian politics.

[6]https://developer.twitter.com/en/docs/twitter-api/tweets/likes/api-reference/get-users-id-liked_tweets

[7]https://developer.twitter.com/en/docs/twitter-api/v1/accounts-and-users/follow-search-get-users/api-reference/get-followers-ids

[8]https://developer.twitter.com/en/docs/twitter-api/v1/tweets/timelines/api-reference/get-statuses-user_timeline

Test Data		Public			Politicians		
Training Data		Public	Politicians	Both	Public	Politicians	Both
Activity	Label Prop. Retweet	**97.9** ± 0.9	95.7 ± 1.3	97.7 ± 1.1	94.5 ± 1.1	97.1 ± 0.9	96.2 ± 1.1
	Label Prop. Mention	92.0 ± 1.5	58.2 ± 1.5	90.3 ± 1.7	62.6 ± 2.1	95.8 ± 1.5	83.5 ± 1.3
	Label Prop. Quote	96.7 ± 1.1	67.8 ± 2.1	95.6 ± 1.1	77.4 ± 2.7	92.4 ± 1.0	89.1 ± 2.4
	Label Prop. Hashtag	92.6 ± 0.9	82.1 ± 1.5	92.9 ± 0.9	74.3 ± 3.1	93.5 ± 2.3	92.0 ± 2.5
Relation	Label Prop. Friend	97.5 ± 1.1	96.5 ± 1.0	97.5 ± 1.1	97.3 ± 0.9	**99.7** ± 0.3	98.9 ± 0.5
	Label Prop. Follow	96.6 ± 1.1	97.4 ± 0.7	96.9 ± 1.0	98.7 ± 0.6	99.5 ± 0.4	99.4 ± 0.4
GCN All		97.0 ± 0.7	91.6 ± 2.1	97.2 ± 0.6	89.8 ± 1.3	97.3 ± 1.5	95.7 ± 1.7

Table 7: Learning from and predicting politician party affiliations. We see that politicians are slightly easier than the public. Transfer from politicians to public hurts performance slightly but is still very viable here with some approaches.

Type	Accuracy
RoBERTa-base Untuned	65.3
Label Prop. Retweet	72.6
GAT Retweet	73.4
Label Prop. RT+@	76.0
GAT RT+@	76.5

Table 8: Accuracy on Canadian users. Differences in the task and available data make this problem much harder, but one can still obtain useful results.

We tested here a different approach to combining data types: simply adding edges in the graph for both retweet and mention interactions (denoted in the table "RT + @"). This enables the use of approximately 25% more labeled train users than retweet alone, and seems to improve performance. We also see that GAT performs slightly better than label propagation. This may indicate that it can deliver benefits on this more challenging data. Nonetheless, the performance of label propagation remains competitive.

Future Work

Our experiments showed disappointing results from methods that combined many different data types, such as TIMME, HAN, and various GCN versions. This suggests that there might be a gap in scalable methods that leverage heterogeneous graphs and/or graphs plus text to deliver superior performance in this domain. Future work testing more recent GNN methods, both homogeneous ones such as GIN (Xu et al. 2019), which might better combine graph plus text, and heterogeneous ones such as (Hu et al. 2020), which additionally combine multiple types of graphs, would be productive. Different ways to incorporate text into these GNN models, such as different ways to produce user embeddings, could also warrant testing. Finally, entirely new methods specialized for this domain that combine multiple data types, especially ones that take scaling and data availability concerns into account, could be valuable as well.

Concurrent work done by Törnberg (2023) found that ChatGPT is effective in labeling the partisanship of Twitter messages from politicians. However, since this study focused exclusively on politicians, the performance of this approach might differ sharply on the general public (Cohen and Ruths 2013). So a comparison on general public users with some of the better-performing methods in our results, such as RoBERTa and label propagation on retweets, would be informative and help understand if and when ChatGPT labeling provides good value for this task.

In our experiments we focused on partisan users, but independent and apolitical users are also key groups in many analyses. It is more challenging to obtain ground truth labels for these users, but nonetheless this is an area for future work with many potential applications. More broadly, applying uncertainty quantification methods to party prediction might help account for less partisan users and produce more fine-grained analysis.

In this work, we focused on Twitter data, as it has been a central platform for political discourse, had good data availability, and is the subject of numerous studies in the literature. With recent content policy changes at Twitter, future data availability has become more uncertain. On the one hand, changes in API access may force researchers to reconsider what data types they use for party prediction, which adds to the value of this work. At the same time, future work extending our analysis to other platforms, such as Facebook or Reddit, would also be valuable. Similarly, we focused on US politics, which is one of the most common countries studied in the literature, as well as Canadian politics as an example of a multiparty parliamentary system. The majority of the insights here should transfer to other democracies, but further testing and adaptation to their unique characteristics would be productive to enable better party prediction, and in turn downstream research, on a more global scale.

Conclusion

Although party prediction is a foundational part of many research projects, in reviewing the literature, we found that (1) it was very challenging to compare the different methods used and that (2) they are often applied without thorough validation. To solve this problem, we first provided a survey of work on this task. This survey highlights not only the reported metrics, but also the data used, which varies widely and is a critical component of the evaluation for this task. Next, we selected state-of-the-art models from this survey and tested them on a consistent dataset we collected. Our results provide both quantitative evidence of the difficulty of comparing approaches based on the literature, and the missing thorough comparison. We also contributed multiple approaches of our own, which we studied and validated through extensive experiments, yielding insights along the way that can help further research in this area. We showed these approaches are competitive with and often out-perform state-of-the-art methods, while opening up new data types and options for practitioners. We also provided practical information about their coverage and efficiency. Our label propagation approach, particularly on retweets (or retweets plus mentions), is especially promising, with both strong performance exceeding the literature and comparable to our other best-performing approaches, and a very large efficiency (and consequently, scalability) gain compared to other approaches. This is an important finding since the common belief in the literature suggests one needs the followership graph for this task, which is much more costly to collect. Our results suggest that not only can the retweets graph be a proxy for followership relations, it in fact provides a more predictive signal for users' party affiliation. Furthermore, we also find that label propagation can be used to learn from politicians (with readily available party affiliations) to make predictions on the general public, without too large of a performance drop. Therefore, we recommend label propagation as a first tool to try for studies that need party affiliation on Twitter. We hope these experiments will provide a foundation for both applying party prediction for downstream research, and development of improved methods, such as on other platforms, or in other democracies.

Acknowledgements

This work was partially funded by the CIFAR AI Chairs Program and by the Centre for the Study of Democratic Citizenship (CSDC). The first author is supported by funding from IVADO and by the Fonds de recherche du Québec.

References

Babuschkin, I.; Baumli, K.; Bell, A.; Bhupatiraju, S.; Bruce, J.; Buchlovsky, P.; Budden, D.; Cai, T.; Clark, A.; Danihelka, I.; Fantacci, C.; Godwin, J.; Jones, C.; Hemsley, R.; Hennigan, T.; Hessel, M.; Hou, S.; Kapturowski, S.; Keck, T.; Kemaev, I.; King, M.; Kunesch, M.; Martens, L.; Merzic, H.; Mikulik, V.; Norman, T.; Quan, J.; Papamakarios, G.; Ring, R.; Ruiz, F.; Sanchez, A.; Schneider, R.; Sezener, E.; Spencer, S.; Srinivasan, S.; Wang, L.; Stokowiec, W.; and Viola, F. 2020. The DeepMind JAX Ecosystem.

Badawy, A.; Ferrara, E.; and Lerman, K. 2018. Analyzing the digital traces of political manipulation: The 2016 russian interference twitter campaign. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 258–265. IEEE.

Badawy, A.; Lerman, K.; and Ferrara, E. 2019. Who falls for online political manipulation? In *Companion Proceedings of The 2019 World Wide Web Conference*, 162–168.

Bail, C. A.; Argyle, L. P.; Brown, T. W.; Bumpus, J. P.; Chen, H.; Hunzaker, M. F.; Lee, J.; Mann, M.; Merhout, F.; and Volfovsky, A. 2018. Exposure to opposing views on social media can increase political polarization. *Proceedings of the National Academy of Sciences*, 115(37): 9216–9221.

Banks, A.; Calvo, E.; Karol, D.; and Telhami, S. 2021. # polarizedfeeds: Three experiments on polarization, framing, and social media. *The International Journal of Press/Politics*, 26(3): 609–634.

Barberá, P. 2015. Birds of the same feather tweet together: Bayesian ideal point estimation using Twitter data. *Political analysis*, 23(1): 76–91.

Barberá, P.; Jost, J.; Nagler, J.; Tucker, J.; and Bonneau, R. 2015. Tweeting From Left to Right. *Psychological Science*, 26: 1531 – 1542.

Boutet, A.; Kim, H.; and Yoneki, E. 2012. What's in your tweets? I know who you supported in the UK 2010 general election. In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 6, 411–414.

Boxell, L.; Gentzkow, M.; and Shapiro, J. M. 2022. Cross-Country Trends in Affective Polarization. *The Review of Economics and Statistics*, 1–60.

Bradbury, J.; Frostig, R.; Hawkins, P.; Johnson, M. J.; Leary, C.; Maclaurin, D.; Necula, G.; Paszke, A.; VanderPlas, J.; Wanderman-Milne, S.; and Zhang, Q. 2018. JAX: composable transformations of Python+NumPy programs.

Brady, W. J.; Wills, J. A.; Jost, J. T.; Tucker, J. A.; and Van Bavel, J. J. 2017. Emotion shapes the diffusion of moralized content in social networks. *Proceedings of the National Academy of Sciences*, 114(28): 7313–7318.

Bright, J. 2016. Explaining the emergence of echo chambers on social media: the role of ideology and extremism. *arXiv preprint arXiv:1609.05003*.

Chen, Z. 2015. Mass ideology-based voting model. In *2015 IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC)*, 871–877. IEEE.

Cohen, R.; and Ruths, D. 2013. Classifying political orientation on Twitter: It's not easy! In *Proceedings of the International AAAI Conference on Web and Social Media*, volume 7, 91–99.

Colleoni, E.; Rozza, A.; and Arvidsson, A. 2014. Echo chamber or public sphere? Predicting political orientation and measuring political homophily in Twitter using big data. *Journal of communication*, 64(2): 317–332.

Conover, M. D.; Gonçalves, B.; Ratkiewicz, J.; Flammini, A.; and Menczer, F. 2011. Predicting the political alignment of twitter users. In *2011 IEEE third international conference on privacy, security, risk and trust and 2011 IEEE third international conference on social computing*, 192–199. IEEE.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *CoRR*, abs/1810.04805.

Eady, G.; Paskhalis, T.; Zilinsky, J.; Bonneau, R.; Nagler, J.; and Tucker, J. A. 2023. Exposure to the Russian Internet Research Agency foreign influence campaign on Twitter in the 2016 US election and its relationship to attitudes and voting behavior. *Nature Communications*, 14(1): 1–11.

Fagni, T.; and Cresci, S. 2022. Fine-Grained Prediction of Political Leaning on Social Media with Unsupervised Deep Learning. *Journal of Artificial Intelligence Research*, 73: 633–672.

FORCE11. 2020. The FAIR Data principles. https://force11.org/info/the-fair-data-principles/.

Gaisbauer, F.; Pournaki, A.; Banisch, S.; and Olbrich, E. 2021. Ideological differences in engagement in public de-

bate on Twitter. *Plos one*, 16(3): e0249241.
Garimella, V. R. K.; and Weber, I. 2017. A long-term analysis of polarization on Twitter. In *Eleventh international AAAI conference on web and social media*.
Gebru, T.; Morgenstern, J.; Vecchione, B.; Vaughan, J. W.; Wallach, H.; Iii, H. D.; and Crawford, K. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12): 86–92.
Green, J.; Edgerton, J.; Naftel, D.; Shoub, K.; and Cranmer, S. J. 2020. Elusive consensus: Polarization in elite communication on the COVID-19 pandemic. *Science Advances*, 6(28): eabc2717.
Grinberg, N.; Joseph, K.; Friedland, L.; Swire-Thompson, B.; and Lazer, D. 2019. Fake news on Twitter during the 2016 U.S. presidential election. *Science*, 363(6425): 374–378.
Gu, Y.; Chen, T.; Sun, Y.; and Wang, B. 2016. Ideology detection for twitter users with heterogeneous types of links. *arXiv preprint arXiv:1612.08207*.
Havey, N. F. 2020. Partisan public health: how does political ideology influence support for COVID-19 related misinformation? *Journal of Computational Social Science*, 3(2): 319–342.
Heiss, R.; von Sikorski, C.; and Matthes, J. 2019. Populist Twitter Posts in News Stories: Statement Recognition and the Polarizing Effects on Candidate Evaluation and Anti-Immigrant Attitudes. *Journalism Practice*, 13(6): 742–758.
Hobbs, W.; Friedland, L.; Joseph, K.; Tsur, O.; Wojcik, S.; and Lazer, D. 2017. " Voters of the Year": 19 Voters Who Were Unintentional Election Poll Sensors on Twitter. In *Eleventh International AAAI Conference on Web and Social Media*.
Hobolt, S. B.; Leeper, T. J.; and Tilley, J. 2021. Divided by the Vote: Affective Polarization in the Wake of the Brexit Referendum. *British Journal of Political Science*, 51(4): 1476–1493.
Hu, Z.; Dong, Y.; Wang, K.; and Sun, Y. 2020. Heterogeneous graph transformer. In *Proceedings of the web conference 2020*, 2704–2710.
Huang, Q.; He, H.; Singh, A.; Lim, S.-N.; and Benson, A. 2020. Combining Label Propagation and Simple Models out-performs Graph Neural Networks. In *International Conference on Learning Representations*.
Iyengar, S.; Lelkes, Y.; Levendusky, M.; Malhotra, N.; and Westwood, S. J. 2019. The origins and consequences of affective polarization in the United States. *Annual Review of Political Science*, 22(1): 129–146.
Jiang, J.; Ren, X.; Ferrara, E.; et al. 2021. Social media polarization and echo chambers in the context of COVID-19: Case study. *JMIRx med*, 2(3): e29570.
Johnston, R. 2019. Affective Polarization in the Canadian Party System, 1988–2015. In *Canadian Political Science Association Meetings, June*, volume 4.
Jost, J. T.; Barberá, P.; Bonneau, R.; Langer, M.; Metzger, M.; Nagler, J.; Sterling, J.; and Tucker, J. A. 2018. How social media facilitates political protest: Information, motivation, and social networks. *Political psychology*, 39: 85–118.
Kamienski, C.; de Camargo Penteado, C. L.; Goya, D.; Rocha, R. V.; de Souza, L. M.; di Gênova, D. V. B.; Ramos, D. F. S.; de França, F. O.; Horita, F.; and dos Santos, C. d. S. 2022. Measuring Network Polarization and Political Sectarianism During the 2020 Pandemic. *IEEE Transactions on Computational Social Systems*.
Kim, Y.; and Kim, Y. 2019. Incivility on Facebook and political polarization: The mediating role of seeking further comments and negative emotion. *Computers in Human Behavior*, 99: 219–227.
Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
Kipf, T. N.; and Welling, M. 2017. Semi-Supervised Classification with Graph Convolutional Networks. arXiv:1609.02907.
Kubin, E.; and von Sikorski, C. 2021. The role of (social) media in political polarization: a systematic review. *Annals of the International Communication Association*, 45(3): 188–206.
Lawson, M. A.; and Kakkar, H. 2022. Of pandemics, politics, and personality: The role of conscientiousness and political ideology in the sharing of fake news. *Journal of Experimental Psychology: General*, 151(5): 1154.
Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. arXiv:1907.11692.
Liu, Y.; Zhang, X. F.; Wegsman, D.; Beauchamp, N.; and Wang, L. 2022. POLITICS: Pretraining with Same-story Article Comparison for Ideology Prediction and Stance Detection. *arXiv preprint arXiv:2205.00619*.
Loshchilov, I.; and Hutter, F. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
Luceri, L.; Deb, A.; Badawy, A.; and Ferrara, E. 2019. Red bots do it better: Comparative analysis of social bot partisan behavior. In *Companion proceedings of the 2019 World Wide Web conference*, 1007–1012.
Makazhanov, A.; and Rafiei, D. 2013. Predicting Political Preference of Twitter Users. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, ASONAM '13, 298–305. New York, NY, USA: Association for Computing Machinery. ISBN 9781450322409.
McCarty, N.; Poole, K. T.; and Rosenthal, H. 2016. *Polarized America: The dance of ideology and unequal riches*. mit Press.
Mou, X.; Wei, Z.; Chen, L.; Ning, S.; He, Y.; Jiang, C.; and Huang, X.-J. 2021. Align Voting Behavior with Public Statements for Legislator Representation Learning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1236–1246.
Németh, R. 2022. A scoping review on the use of natural language processing in research on political polarization: trends and research prospects. *Journal of Computational Social Science*, 1–25.
Osmundsen, M.; Bor, A.; Vahlstrup, P. B.; Bechmann, A.; and Petersen, M. B. 2021. Partisan Polarization Is the Primary Psychological Motivation behind Political Fake News Sharing on Twitter. *American Political Science Review*, 115(3): 999–1015.

Pastor-Galindo, J.; Zago, M.; Nespoli, P.; Bernal, S. L.; Celdrán, A. H.; Pérez, M. G.; Ruipérez-Valiente, J. A.; Pérez, G. M.; and Mármol, F. G. 2020. Spotting political social bots in Twitter: A use case of the 2019 Spanish general election. *IEEE Transactions on Network and Service Management*, 17(4): 2156–2170.

Pathak, A.; Madani, N.; and Joseph, K. 2021. A Method to Analyze Multiple Social Identities in Twitter Bios. *Proceedings of the ACM on Human-Computer Interaction*, 5(CSCW2): 1–35.

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

Pennacchiotti, M.; and Popescu, A.-M. 2011. A machine learning approach to twitter user classification. In *Proceedings of the international AAAI conference on web and social media*, volume 5, 281–288.

Pennycook, G.; Epstein, Z.; Mosleh, M.; Arechar, A. A.; Eckles, D.; and Rand, D. G. 2021. Shifting attention to accuracy can reduce misinformation online. *Nature*, 592(7855): 590–595.

Preoţiuc-Pietro, D.; Liu, Y.; Hopkins, D.; and Ungar, L. 2017. Beyond binary labels: political ideology prediction of twitter users. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 729–740.

Rheault, L.; and Musulan, A. 2021. Efficient detection of online communities and social bot activity during electoral campaigns. *Journal of Information Technology & Politics*, 18(3): 324–337.

Rodríguez-García, M. Á.; Herranz, S. M.; and Unanue, R. M. 2022. URJC-Team at PoliticEs 2022: Political Ideology Prediction using Linear Classifiers.

Ruggeri, K.; Većkalov, B.; Bojanić, L.; Andersen, T. L.; Ashcroft-Jones, S.; Ayacaxli, N.; Barea-Arroyo, P.; Berge, M. L.; Bjørndal, L. D.; Bursalıoğlu, A.; et al. 2021. The general fault in our fault lines. *Nature human behaviour*, 1–11.

Sinno, B.; Oviedo, B.; Atwell, K.; Alikhani, M.; and Li, J. J. 2022. Political Ideology and Polarization: A Multi-dimensional Approach. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 231–243.

Stefanov, P.; Darwish, K.; Atanasov, A.; and Nakov, P. 2020. Predicting the topical stance and political leaning of media using tweets. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 527–537.

Törnberg, P. 2023. ChatGPT-4 Outperforms Experts and Crowd Workers in Annotating Political Twitter Messages with Zero-Shot Learning. *arXiv preprint arXiv:2304.06588*.

Tucker, J. A.; Guess, A.; Barberá, P.; Vaccari, C.; Siegel, A.; Sanovich, S.; Stukal, D.; and Nyhan, B. 2018. Social Media, Political Polarization, and Political Disinformation: A Review of the Scientific Literature. *Political Polarization, and Political Disinformation: A Review of the Scientific Literature*.

Tucker, J. A.; Theocharis, Y.; Roberts, M. E.; and Barberá, P. 2017. From liberation to turmoil: Social media and democracy. *Journal of democracy*, 28(4): 46–59.

Törnberg, P. 2022. How digital media drive affective polarization through partisan sorting. *Proceedings of the National Academy of Sciences*, 119(42): e2207159119.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Liò, P.; and Bengio, Y. 2018. Graph Attention Networks. In *International Conference on Learning Representations*.

Voelkel, J.; Stagnaro, M.; Chu, J.; Pink, S.; and Mernyk, J. 2022. Megastudy Identifying Successful Interventions to Strengthen Americans' Democratic Attitudes.

Wang, M.; Zheng, D.; Ye, Z.; Gan, Q.; Li, M.; Song, X.; Zhou, J.; Ma, C.; Yu, L.; Gai, Y.; Xiao, T.; He, T.; Karypis, G.; Li, J.; and Zhang, Z. 2019a. Deep Graph Library: A Graph-Centric, Highly-Performant Package for Graph Neural Networks. *arXiv preprint arXiv:1909.01315*.

Wang, X.; Ji, H.; Shi, C.; Wang, B.; Ye, Y.; Cui, P.; and Yu, P. S. 2019b. Heterogeneous graph attention network. In *The world wide web conference*, 2022–2032.

Wang, Y.; Feng, Y.; Hong, Z.; Berger, R.; and Luo, J. 2017. How polarized have we become? a multimodal classification of trump followers and clinton followers. In *International conference on social informatics*, 440–456. Springer.

Warner, B. R.; and Villamil, A. 2017. A test of imagined contact as a means to improve cross-partisan feelings and reduce attribution of malevolence and acceptance of political violence. *Communication Monographs*, 84(4): 447–465.

Waugh, A. S.; Pei, L.; Fowler, J. H.; Mucha, P. J.; and Porter, M. A. 2011. Party Polarization in Congress: A Network Science Approach. arXiv:0907.3509.

Wojcieszak, M.; Casas, A.; Yu, X.; Nagler, J.; and Tucker, J. A. 2022. Most users do not follow political elites on Twitter; those who do show overwhelming preferences for ideological congruity. *Science Advances*, 8(39): eabn9418.

Wolf, T.; Debut, L.; Sanh, V.; Chaumond, J.; Delangue, C.; Moi, A.; Cistac, P.; Rault, T.; Louf, R.; Funtowicz, M.; and Brew, J. 2019. HuggingFace's Transformers: State-of-the-art Natural Language Processing. *ArXiv*, abs/1910.03771.

Xiao, Z.; Song, W.; Xu, H.; Ren, Z.; and Sun, Y. 2020. TIMME: Twitter Ideology-detection via Multi-task Multi-relational Embedding. *arXiv preprint arXiv:2006.01321*.

Xu, K.; Hu, W.; Leskovec, J.; and Jegelka, S. 2019. How Powerful are Graph Neural Networks? In *International Conference on Learning Representations*.

Yang, K.-C.; Hui, P.-M.; and Menczer, F. 2022. How Twitter data sampling biases US voter behavior characterizations. *PeerJ Computer Science*, 8: e1025.

Zhu, X.; and Ghahramani, Z. 2002. Learning from labeled and unlabeled data with label propagation.

Checklist

1. For most authors...

(a) Would answering this research question advance science without violating social contracts, such as violating privacy norms, perpetuating unfair profiling, exacerbating the socio-economic divide, or implying disrespect to societies or cultures? Yes
(b) Do your main claims in the abstract and introduction accurately reflect the paper's contributions and scope? Yes
(c) Do you clarify how the proposed methodological approach is appropriate for the claims made? Yes
(d) Do you clarify what are possible artifacts in the data used, given population-specific distributions? Yes
(e) Did you describe the limitations of your work? Yes
(f) Did you discuss any potential negative societal impacts of your work? Yes
(g) Did you discuss any potential misuse of your work? Yes
(h) Did you describe steps taken to prevent or mitigate potential negative outcomes of the research, such as data and model documentation, data anonymization, responsible release, access control, and the reproducibility of findings? Yes
(i) Have you read the ethics review guidelines and ensured that your paper conforms to them? Yes

2. Additionally, if your study involves hypotheses testing...
(a) Did you clearly state the assumptions underlying all theoretical results? NA
(b) Have you provided justifications for all theoretical results? NA
(c) Did you discuss competing hypotheses or theories that might challenge or complement your theoretical results? NA
(d) Have you considered alternative mechanisms or explanations that might account for the same outcomes observed in your study? NA
(e) Did you address potential biases or limitations in your theoretical framework? NA
(f) Have you related your theoretical results to the existing literature in social science? NA
(g) Did you discuss the implications of your theoretical results for policy, practice, or further research in the social science domain? NA

3. Additionally, if you are including theoretical proofs...
(a) Did you state the full set of assumptions of all theoretical results? NA
(b) Did you include complete proofs of all theoretical results? NA

4. Additionally, if you ran machine learning experiments...
(a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? No, the data is not released per Twitter terms of use and to preserve anonymity of the users.
(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? Yes
(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? Yes
(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? Yes
(e) Do you justify how the proposed evaluation is sufficient and appropriate to the claims made? Yes
(f) Do you discuss what is "the cost" of misclassification and fault (in)tolerance? Yes

5. Additionally, if you are using existing assets (e.g., code, data, models) or curating/releasing new assets, **without compromising anonymity**...
(a) If your work uses existing assets, did you cite the creators? Yes
(b) Did you mention the license of the assets? NA
(c) Did you include any new assets in the supplemental material or as a URL? Yes
(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? Yes
(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? Yes
(f) If you are curating or releasing new datasets, did you discuss how you intend to make your datasets FAIR (see FORCE11 (2020))? NA
(g) If you are curating or releasing new datasets, did you create a Datasheet for the Dataset (see Gebru et al. (2021))? NA

6. Additionally, if you used crowdsourcing or conducted research with human subjects, **without compromising anonymity**...
(a) Did you include the full text of instructions given to participants and screenshots? NA
(b) Did you describe any potential participant risks, with mentions of Institutional Review Board (IRB) approvals? NA
(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? NA
(d) Did you discuss how data is stored, shared, and deidentified? NA

Impact and Ethics

Political information has potential for misuse by malicious actors. In the following we discuss impacts and ethical considerations of this work and steps taken to mitigate potential harms.

Malicious actors, such as governments attempting to interfere in other countries' elections, are already known to spread misinformation and try to sway users to particular ideologies and voting patterns (Eady et al. 2023). In addition, although exact mechanisms and the role of the platform are still debated, social media has been consistently linked to increasing partisan polarization (Németh 2022). Consequently, there is no security through obscurity in this domain. Strong understanding and effective countermeasures are needed to prevent the spread of misinformation and divisiveness (Tucker et al. 2017).

As we show in our introduction and literature review, party prediction has a fundamental role in detecting partisan biases on Twitter. To understand how groups become polarized or influenced, it is necessary to accurately identify members of these groups. Accurate party prediction can help researchers and policymakers understand how and why certain groups become polarized and how information and ideas spread within and between these groups. Although this could potentially be misused to attempt to influence users, this risk is relatively low because our methods do not enable fine-grained targeting. In addition, fine-grained targeting has already been done using other unrelated tools, such as targeted advertisements through social media platforms. Therefore, the risks are low compared to the societal benefits of better party predictions offered by our method, which should in turn lead to a better and more robust understanding of how to create healthy online environments, mitigate extreme polarization, reduce the influence of bots, and limit the spread of misinformation.

In our study, we only use publicly available data that has been collected in compliance with Twitter's terms of service. In addition, we will never release any identifiable information in order to prevent any future misuse of the user data collected in this study. Through our collection process, all users collected publicly discuss politics in both their tweets and user profiles, and consequently would not assume their political affiliations are hidden. We note that this is an equivalent or stronger constraint on users included in our study than in many other studies in the literature (see for example Barberá 2015; Grinberg et al. 2019; Pathak, Madani, and Joseph 2021; Hobbs et al. 2017).

Comparing Coverage of Signals

Another critical question is how many users each approach can be applied to. We compare the coverage of different methods in Table 9, which shows the percentage of users that have the data needed to run the method. Please note that here the test sets vary between approaches, but are always strictly larger than in experiments of Table 4 where we selected only users that have the data needed to apply all approaches (to keep the test set constant there). Therefore, this is a more general and harder task. Further, we report the average users for which the needed data could be retrieved per 15 minute Twitter API cycle. Unlike the theoretical numbers in the preceding table, these numbers are calculated from the real data. Some additional notes on this and other implementation aspects are provided in the supplementary materials.

Unsurprisingly, performance almost always decreases compared with exclusively considering users with all interaction types simultaneously. Nonetheless, we see that our methods still provide strong performance. Here RoBERTa-base is tied for top performance, and has 100% coverage and 95% accuracy. On the graph side, using GCN with all interactions except Like gives 100% coverage with over 93% accuracy, while one can get even better performance and still over 90% coverage from several versions of label propagation, especially retweet. Furthermore, we see that retrieving tweets (from which retweets, mentions, quotes, and hashtags can all be derived also; 343.2 ± 152.6 users per 15m) is far faster than retrieving friends or followers (14.5 ± 0.7 users per 15m), and those two are again significantly faster than likes (3.4 ± 5.1 users per 15m). Thus, using label propagation retweet or a similar method may be advisable for large-scale work. Regardless, this information can help practitioners make an informed decision.

Baseline Methods

We compare our approaches with 4 state-of-the-art methods from the literature. These were selected based on performance, taking into account the difficulty level of the data they were tested on. Here we briefly summarize some of their key points and provide the details on how we implemented them:

Barberá (Barberá 2015; Barberá et al. 2015) This item response model assigns scores to users based on who they follow. We use the Tweetscores[9] implementation, which compares a user against "elite" (politicians and media) users with pre-trained scores.

We classify any user with score greater (resp. less) than 0 as Republican (resp. Democrat), which both matches intuition and gives the best performance empirically. In particular, we show here the results of different classification thresholds for Barberá's model. This shows a single run in the setting of Table 9 (all users available), with threshold increments of 0.05 from -2 to 1.

Preoţiuc-Pietro (Preoţiuc-Pietro et al. 2017) This is a bag-of-words style approach with a custom, specialized vocabu-

[9] https://github.com/pablobarbera/twitter_ideology

Type	Accuracy	Cov.	Speed
Barberá (2015)	94.6	92.7	14.5
Preoţiuc-Pietro et al. (2017)	84.7	100.0	343.2
POLITICS (2022) Finetuned	95.5	100.0	343.2
RoBERTa-base Finetuned	95.7 ± 0.2	100.0	343.2
RoBERTa-large Finetuned	95.4 ± 0.1	100.0	343.2
Label Prop. Retweet	95.7 ± 0.9	91.2	343.2
Label Prop. Mention	87.8 ± 0.7	92.1	343.2
Label Prop. Quote	94.7 ± 0.5	80.4	343.2
Label Prop. Hashtag	90.4 ± 1.3	74.9	343.2
Label Prop. Like	94.1 ± 0.7	93.1	3.4
Label Prop. Friend	94.8 ± 0.6	93.8	14.5
Label Prop. Follow	94.7 ± 0.6	82.2	14.4
GCN All-Like	93.6 ± 0.9	100.0	14.4
GCN All-Like-Follow	94.1 ± 0.9	99.9	14.5
GCN RT+QT+Friend	94.6 ± 0.7	99.6	14.5
GCN RT+Friend	94.2 ± 1.0	99.6	14.5
GCN QT+Friend	93.7 ± 1.0	98.8	14.5
GCN RT+QT	93.9 ± 0.7	91.3	343.2
GCN Act. (RT+QT+@+#)	92.7 ± 1.2	98.1	343.2
GCN Rel. (Friend+Follow)	92.2 ± 0.8	94.8	14.4

Table 9: Accuracy on users without all relations, along with coverage (Cov.; percentage of users the approach will work on), and speed (the average number of users one can retrieve per 15 minute Twitter API cycle with that data type, lower means more cost to retrieve the required data, i.e., higher data acquisition cost).

lary of 352 politics-related words. For each user we concatenate all their tweets into one document and calculate the counts of these words. Following the original paper, we then use this feature vector as input to a logistic regression that classifies each user. We implement the logistic regression through scikit-learn (Pedregosa et al. 2011) with default hyperparameters.

POLITICS (Liu et al. 2022) This model is based on RoBERTa-base, but adds political domain adaptation through training on news articles with ideology labels. We pass the final-layer embedding of the [CLS] token through a single fully-connected one-layer classification head. With this architecture, we fine-tune the model with the same setup that we used for RoBERTa-base. We also report performance of an untuned version where we obtain user embeddings from averaging the pretraining-only tweet chunk [CLS] embeddings, and classifying users in the same way that the we classify the user embeddings from GCN.

TIMME (Xiao et al. 2020) This approach is based on a variation of a GCN, but the architecture is adapted to learn from multiple data types simultaneously and end-to-end. This contrasts with our GCN approach which trains an independent GCN per data type, and combines them in a separate final stage. Therefore, this design hopes to learn more complex interactions between the different data types, at the cost of being much more computationally intensive and potentially more difficult to train. Unfortunately, even when using high-end AI-specialized hardware and the original authors' code,[10] we found the computational burden is severe. We elaborate in the following note.

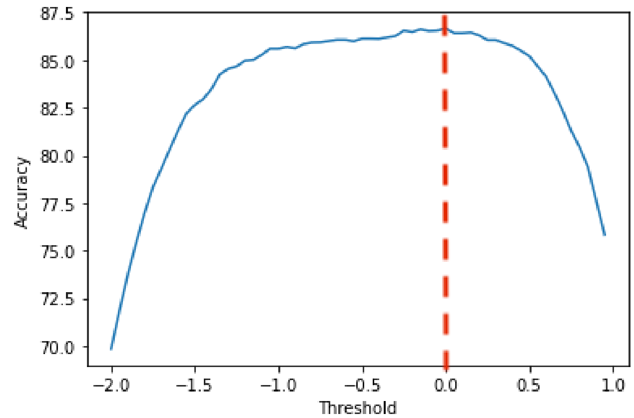

Figure 3: Tuning the threshold for classification with Barberá's approach. Accuracy is maximized at exactly 0.

A Note on TIMME Model

We attempted to implement TIMME using 100GB RAM and an RTX8000 GPU (which has 48GB VRAM), using the code released by the authors. However, we found both versions TIMME and TIMME-Hierarchical frequently produced out of memory errors with our scale of data. This is likely because, unlike all our proposed graph-based approaches, TIMME requires loading multiple graphs in memory simultaneously for end-to-end training. We also note that our data has over 4 times more interactions than the largest dataset used in the TIMME paper, and over 100 times more than the smallest. In addition, in some tests where we restricted the data and got it to run, it was over 100 times slower per epoch than our GCN approach (which is itself 100 times slower than our label propagation approach), and did not give good performance. Therefore, although this is arguably the most sophisticated approach and might give strong performance in some settings like the original paper, the current version is challenging to apply to larger datasets.

Distribution of Unretrievable US Public Users

We examine the distribution of the missing users in Table 11. We see that a strong majority are Republicans. We hypothesize that many of these users left around the January 6^{th} capitol attack, either voluntarily or during the subsequent wave of suspensions.

Distribution of Retweets and Quotes

In Table 12 we show the proportion of tweets that are retweets or quotes. We see that there is a similar proportion of retweets between the US and Canadian public datasets. This is intuitive since both datasets contain users from the general public and are sampled with relatively similar processes aside from the country difference. On the other hand,

[10] https://github.com/PatriciaXiao/TIMME

Methods	Accuracy	Media	Activity	Relation	Content	Type	Size	Code	Difficulty
Liu et al. (2022)	<50%				✓	Public	1,079	✓	Hard
Pastor-Galindo et al. (2020)	NA				✓	Public	20,364	✓	Easy
Sinno et al. (2022)	55%	✓			✓	News	175		Easy
Yang, Hui, and Menczer (2022)	NA	✓			✓	Public	NA	✓	Easy
Chen (2015)	NA			✓	✓	Both	NA		Hard
Bright (2016)	NA		✓		✓	Both	NA		Medium
Gaisbauer et al. (2021)	NA		✓		✓	Both	NA		Medium
Garimella and Weber (2017)	NA	✓	✓	✓	✓	Both	NA		Medium
Kamienski et al. (2022)	NA		✓	✓	✓	Public	NA		Easy

Table 10: Appendix: Survey of methods to predict ideology (with accuracy below 65% or no mention of the accuracy). Note we have also included a news dataset ("Both" refers to public and politicians both, not news).

	Republican	Democrat
Suspended	1,824	439
Deleted	1,857	543
Private	186	267

Table 11: Distribution of users whose data could not be retrieved retroactively. The majority are Republican, possibly users who left around January 2021, i.e., the wave of suspensions that included Donald Trump.

the US Politicians show a clear difference in behavior compared to the public – they retweet less and quote more.

	Tweets	% Retweet	% Quote
US Public	5,804,713	75.8	7.2
US Politicians	156,562	53.6	34.9
CA Public	11,361,581	74.3	-

Table 12: Proportions of tweets that are retweets or quotes. US and CA public are similar, while politicians retweet less and quote more.

Additional Information on Canadian Data

Collection We sampled 1% of real-time tweets containing at least one of the following keywords:

'trudeau', 'freeland', 'o'toole', 'bernier', 'blanchet', 'jagmeet singh', 'annamie', 'debate commission', 'reconciliation', 'elxn44', 'cdnvotes', 'canvotes', 'canelection', 'cdnelection', 'cdnpoli', 'canadianpolitics', 'canada', 'forwardforeveryone', 'readyforbetter', 'securethefuture', 'NDP2021', 'votendp', 'orangewave2021', 'teamjagmeet', 'UpRiSingh', 'singhupswing', 'singhsurge', 'VotePPC', 'PPC', 'peoplesparty', 'bernierorbust', 'mcga', 'saveCanada', 'takebackcanada', 'maxwillspeak', 'LetMaxSpeak', 'FirstDebate', 'frenchdebate', 'GovernmentJournalists', 'JustinJournos', 'everychildmatters', 'votesplitting', 'ruralcanada', 'debatdeschefs', 'électioncanadienne', 'polican', 'bloc', 'jevotebloc'

Labeling From the main dataset, we sampled users with the following keywords in their profile:

CPC: 'erin o'toole', 'andrew scheer', 'conservative', 'conservative party', 'cpc', 'cpc2021', 'cpc2019', 'conservative party of canada'

GPC: 'annamie paul', 'green party', 'gpc', 'gpc2019', 'gpc2021', 'green party of canada'

LPC: 'justin trudeau', 'liberal', 'liberal party', 'lpc', 'lpc2021', 'lpc2021', 'lpc2019', 'liberal party of canada'

NDP: 'jagmeeet singh', 'new democrat', 'new democrats', 'new democratic party', 'ndp', 'ndp2021', 'ndp2019'

PPC: 'maxime bernier', 'people's party', 'ppc', 'ppc2019', 'ppc2021', 'people's party of canada'

Two political science graduate students who have studied Canadian politics then manually verified or corrected the labels. This led to the counts of labeled users shown in Table 13.

Party	Count
CPC	2039
LPC	1808
NDP	704
PPC	451
GPC	299

Table 13: Counts of manually-labeled Canadian users.

Supplementary Implementation Information

For each individual interaction type, we train only on the top 50% of most active users for that relation according to raw counts. This filter is not applied to test data.

GCN We train for 1000 epochs with the Adam optimizer (Kingma and Ba 2014) with PyTorch default parameters (learning rate = 1e-3). When we use a random forest to combine different data types, we use the scikit-learn (Pedregosa et al. 2011) implementation with default hyperparameter settings. As in all experiments, we train it using users with profile classifier labels, and report test results on a (fully separate) set of users using manual labels.

GAT We use the implementation and hyperparameters from https://github.com/xnuohz/CorrectAndSmooth-dgl corresponding to (Huang et al. 2020). This includes hyperparameters of 2000 epochs of training, learning rate 0.002, dropout 0.75, and 3 layers. We use hidden dimension 100 matching our GNN models. We slightly adjust the last GAT layer to exactly match the others (i.e. GATConv, linear, batchnorm, activation, dropout), followed by a final linear layer, bias, and softmax. This produces practical final layer embeddings, similar those we obtain from the GCN, which we tested in Table 14.

HAN We use the implementation of https://github.com/dmlc/dgl/tree/master/examples/pytorch/han which is a reimplementation of the original HAN paper (Wang et al. 2019b). For the hyperparameters, we found that the ones given originally, applied to our data, resulted in collapse of the predictions to a single class. Matching our GAT hyperparameters, however, gave more effective results reported in the main experiments. Specifically, we matched learning rate, dropout, number of heads, hidden units, and number of epochs. We left the default 20 batch size, and removed weight decay for both a fairer comparison with our other models which do not use it, and because all nonzero values we tested (including the default) resulted in class collapse of the predictions.

For the metapaths, we use the basic ones corresponding to the edge types: retweet, mention, quote, and hashtag. We also tested using only retweets and mentions, and only retweets and quotes. However, these proved significantly more unstable than using all four together (resulting in more cases of class collapse), and with the exception of two runs of the ten using all four edge types resulted in equal or better performance. Of the two cases where it did not, one was when using all four resulted in class collapse, while in the other the margin was slight (in favor of retweet plus quote by less than half a percentage point).

Label Propagation There are two parameters: the number of iterations of the propagation process, and the rate α at which new label information replaces the old one. We first tested it on projected graphs, where we found performance decreases consistently and monotonically with more iterations irrespective of α value (specifically, tested iterations in [1,2,3,4,5,6,8,10,15,20,25], with fully tested alpha values [0.1,0.5,0.9] and partially tested [0.03,0.97]). When using a single iteration, this method corresponds to taking the majority vote of the neighbors' labels, and changing α has no effect.

We later found that performance is better on the direct graph, specifically with two iterations and $\alpha = 0.5$. With 1 iteration on this graph, the performance is poor, while with more iterations performance remains constant (accounting for margin of error) or decreases. The direct graph takes much longer to run, so due to time/computation constraints we were not able to test more values of α in this setting. Therefore, except where stated otherwise, we report results from the best performing version, i.e., two iterations on direct graph with $\alpha = 0.5$.

RoBERTa In order to provide additional context for the models, We concatenated each user's tweets in time order into chunks. Since the positional encoding scheme of these language models limits the length of input to a fixed number of tokens, we start a new chunk each time the previous chunk goes above the length limit of the model. Any tweets that don't fit into a single chunk are truncated and placed into a chunk of their own. As a result, no tweet are split between chunks. Note that we also use this same preprocessing for the POLITICS (Liu et al. 2022) baseline model described previously, which has a similar architecture.

These models are pretrained without a sequence classification task. Consequently, despite RoBERTa models having a [CLS] token like BERT and many similar language models, the representation of this token is not pretrained and thus unsuitable for direct use in downstream tasks. In some experiments we tested the model without the finetuning needed to properly learn this token. Therefore, to improve comparability, in all experiments with RoBERTa instead of [CLS] we use the mean of all individual word embeddings from the final output layer.

To further improve the prediction accuracy, we fine-tuned these models with a tweet chunk classification task. First, we label each chunk in the train set according to the profile classifier (weak) label of the corresponding user. We add a fully-connected dense layer (the "classification head") to each model. The input size of this dense layer is equal to the size of the hidden dimensions of each transformer model, while the output size of this layer is equal to the number of label classes (2 in our case.). We then finetuned each model end-to-end for 1 epoch using the Optax (Babuschkin et al. 2020) implementation of the adamw optimizer (Loshchilov and Hutter 2017) with default weight decay strength 1e-4. We set the learning rate to 1.592e-5 for RoBERTa-base, 2.673e-5 for POLITICS, and 4.269e-6 for RoBERTa-large, based on gridsearch hyperparameter tuning on a validation set of users labeled by our profile classifier. When evaluated on the profile classifier labels, this setup leads to $91.5\% \pm 0.2\%$ test accuracy on tweet chunks for RoBERTa-base, $91.6\% \pm 0.3\%$ for POLITICS and $91.9\% \pm 0.3\%$ for RoBERTa-large. Note that these accuracies for tweet chunk classification are only on weak labels, and are correlated with but not directly comparable to user classification reported in all other experiments.

To get a prediction for each user, instead of a prediction on an individual tweet chunk, we first predict a label for each of a user's chunks. Then we take the majority vote.

Additional Experiments

Effect of Supervision

We compare our GCN model, which does both unsupervised link prediction and supervised party prediction with training data labeled by the profile classifier, with training embeddings through a fully unsupervised GCN doing link prediction alone. We also evaluate the difference between using the supervision in the graph model itself versus sending the embeddings to a random forest (RF) for the final prediction.

Results are shown in Table 14. First, we see that although the margin is not huge, the semisupervised GCN version performs consistently better. This indicates the supervision helps to produce more informative embeddings from the GCN.

Second, we see that the RF, although it enables one to combine different interaction types together as in the experiments above, hurts performance slightly with GCN. With GAT, it is significantly detrimental, possibly due to architectural differences. However, even without the RF, both GCN and GAT are outperformed by label propagation in almost every case.[11]

[11]Consequently, in other experiments, unless otherwise noted,

	Retweet	Mention	Quote	Hashtag	Friend	Follow
GCN SS	96.7 ± 0.7	83.7 ± 1.8	90.3 ± 1.5	88.7 ± 1.8	96.8 ± 0.6	93.3 ± 1.1
GCN SS+RF	96.2 ± 1.2	85.6 ± 1.7	89.4 ± 1.1	88.1 ± 1.8	96.5 ± 0.7	92.5 ± 1.2
GCN US+RF	96.1 ± 0.6	83.1 ± 2.2	87.6 ± 1.2	87.1 ± 1.3	94.0 ± 1.1	91.9 ± 0.9
GAT SS	96.9 ± 0.4	88.1 ± 1.3	93.5 ± 1.0	91.4 ± 1.3	96.7 ± 0.4	95.4 ± 0.8
GAT SS+RF	91.9 ± 1.6	73.3 ± 3.5	81.9 ± 1.6	78.2 ± 3.4	88.3 ± 1.1	86.3 ± 1.2
Label Prop.	97.2 ± 0.5	91.0 ± 0.9	95.7 ± 0.8	92.6 ± 0.7	96.5 ± 0.6	96.1 ± 0.8

Table 14: Comparing semisupervised (SS) vs. unsupervised (US) GCN. Semisupervised performs better.

	Retweet	Mention	Quote	Hashtag	Friend	Follow
Pro. GCN-1L	96.2 ± 1.2	85.6 ± 1.7	89.4 ± 1.1	88.1 ± 1.8	96.5 ± 0.7	92.5 ± 1.2
Dir. GCN-1L	92.1 ± 0.8	77.1 ± 2.3	83.1 ± 1.7	79.1 ± 1.7	88.9 ± 1.8	75.1 ± 2.8
Dir. GCN-2L	87.4 ± 1.5	71.2 ± 2.3	76.3 ± 2.6	76.3 ± 1.3	82.4 ± 2.8	69.3 ± 2.7
Pro. GAT	96.6 ± 0.6	66.9 ± 1.3	92.2 ± 1.3	90.2 ± 1.6	94.5 ± 0.9	96.1 ± 0.9
Dir. GAT	96.9 ± 0.4	88.1 ± 1.3	93.5 ± 1.0	91.4 ± 1.3	96.7 ± 0.4	95.4 ± 0.8
Pro. LP-1I	96.6 ± 0.6	66.5 ± 1.9	83.4 ± 0.9	86.3 ± 1.1	70.1 ± 1.6	96.6 ± 0.8
Dir. LP-1I	75.4 ± 1.9	68.1 ± 1.8	68.7 ± 1.9	66.3 ± 1.9	88.0 ± 1.4	87.3 ± 1.5
Dir. LP-2I	97.2 ± 0.5	91.0 ± 0.9	95.7 ± 0.8	92.6 ± 0.7	96.5 ± 0.6	96.1 ± 0.8

Table 15: Comparing projected (Pro.) vs. direct (Dir.) graphs. Projection improves GCN performance. But it hurts label propagation (LP) performance, at least if the number of iterations is tuned for it.

Effect of Projection

In this experiment, we examine the impact on performance of using projected (indirect) graphs vs. the original (direct) ones. We show results in Table 15.

We see that GCN performance degrades when not projecting the graph. Adding another layer to the GCN, which in principle might help it use information from two-hop neighbors in a similar way to projection, turns out to be further detrimental. On the other hand, for GAT and label propagation, the direct graph generally performs better. The exception is the follow relation, which in all cases is better with the projected graph, which might reflect underlying user behavior differences between the interaction types. For label propagation in particular, the best version we found is two iterations on the direct graph. The best projected graph version is one iteration, but it is clearly worse overall, while one iteration on the direct graph is worse still.

Overall, along with the earlier results on runtime, the choice of direct vs. projected graph can have a significant impact. It is not as commonly tested as simple hyperparameters, and may be worth examining in more contexts.

Users Without All Interaction Types Experiment: Additional Notes

Average users retrievable is calculated by first taking the per user counts of the necessary data type in our dataset. From there and the numbers in Table 6, we calculate how many API requests are needed to retrieve that user's data, and finally how many users can be retrieved in every 15 minutes. Retrieval from different API endpoints can be run in parallel, so when an approach uses a combination of data types, we report on the one that takes the longest to retrieve.

The standard deviation reported for our approaches here is the result of re-running the model itself 10 times, as in other experiments, but here because we examine all possible test users the test set does not change within these 10 runs. Existing state-of-the-art models were run once. The standard deviation for users retrievable is not from multiple runs; rather, it is the standard deviation of our sample in its estimation of the average users retrievable.

When evaluating combinations of interaction types we use the GCN embedding where available; otherwise we treat the embedding as all zeros. So for example, if we are considering Retweet plus Quote (RT+QT in Table 9), then if both types of activity are available we use both. While if only one is available we concatenate that one with a vector of zeroes for the missing one, which tells the final classification model that quote is not available.

we report all GCN results using the semisupervised version with RF, so the results are comparable when combining data types. While for GAT, which we seldom use for combining data types, we report the version without RF.

Politicians Experiment: Additional Notes

In this experiment we use the unsupervised GCN version of our model. This lets us test how training the final classification on US Politicians alone will translate to performance on US Public, with the GCN part of our model (and thus the embeddings) held constant.

When testing on US Public using US Politicians, we use all of them, while when testing on US Politicians, we do a 75-25 random split. In all cases the setting corresponds to Table 4, i.e. users with all interaction types, again to keep the test set consistent between the different approaches.

Computational Resources and Libraries

Most experiments were done using RTX8000 GPUs. A number of text-based experiments were also run on v3-8 and v4-8 TPU VMs. Graph-based models were implemented using DGL (Wang et al. 2019a), and language models using HuggingFace (Wolf et al. 2019) in JAX (Bradbury et al. 2018). To run all models of the main comparison experiment (Table 4), the roughly estimated time using 10 RTX8000 is one to two weeks.

Additional Literature Summary

In Table 10 we provide information on papers with lower or not reported performance to predict ideology, below the threshold for inclusion in Table 1. In contrast with Table 1, this survey includes only papers for which the accuracy was less than 65% or unknown.