Social Network Analysis of Twitter Topic-Network Structures during the 2019 European Elections

Andrea Domenico Antonacci, SNR 2032465

January 15, 2020

Master Thesis in Marketing Management
Tilburg School of Economics and Management
Tilburg University

Supervisor:	dr. Hannes Datta	Associate Professor
Second reader:	Nazli Alagöz	Ph.D. Candidate

Contents

Abstract

This paper explores how Twitter users differently interact based on the topic of discussion in online conversations. For this purpose, we employ concepts of Social Network Analysis to examine how node-level network topology measures may affect the likelihood of interaction and the extent to which two nodes interact in distinct topic-network structures. We collect 21 million sampled tweets via the Twitter API, tracking more than 700 keywords during the 2019 European Elections. We perform our analysis on a final data set of 2,259,717 tweets on five politics-related topics. We find systematic differences in the direction of the effect of centrality measures between distinct topics of discussion, but not for degree and clustering metrics. Notably, the target user's eigencentrality positively predicts the interaction between users, regardless of the topic. However, among the pairs of interacting users, the direction of this effect varies between topics. We also show that the sender's eigencentrality negatively influences the probability of interaction, but not the extent to which users interact between them.

1 Introduction

The rise of synchronous computer-mediated communication (CMC) in the last decades allowed for instant sharing and access to worldwide information, which profoundly revolutionized social activity and human interaction on the Internet. Ideas, political messages, tweets, memes, and advertisements spread today on the Internet within enormous networks of nodes and edges. Using online social networks services (OSN), individuals can choose whom to engage with, constructing their own network of interactions. Albeit recommendation algorithms have taken hold in the recent years, users still actively select their information sources. This process varies according to the way the social network platform is designed – e.g., by following, subscribing, sharing, etc. – yet its rationale is almost universal.

Many recent studies on social network analysis (SNA) document these patterns of interaction. Central issues of particular interest range from the evolution of interaction behavior in social networks over time (e.g., Mulder and Leenders, 2019) to the role of message in viral marketing campaigns (e.g., Liu-Thompkins, 2012) and Twitterstorms phenomena (e.g., Timm et al., 2016). Researchers have extensively studied single network characteristics – i.e., tie strength, centrality, and density measures, to name a few – to classify users and explain information diffusion at the individual level in social networks (Granovetter, 1973; Borgatti, 2005). In fact, SNA has been applied to OSN in many fields, from studying the influence of word of mouth on new product launches (e.g., Deer et al., 2019) to sexual relationships and couple formation (e.g., Ortega and Hergovich, 2017). While these approaches have proven successful, a deeper understanding of information diffusion at the network level can only be achieved when taking multiple measurements jointly (Himelboim et al., 2017).

This appears of utmost relevance in the light of the prominent influence that OSN exert on the public opinion nowadays (Kwak et al., 2010; Lerman and Ghosh, 2010). For instance, it might be especially worth studying how OSN are displacing more traditional media outlets during political election campaigns (see, e.g., Hemsley, 2019; Buccoliero et al., 2020; Kruikemeier, 2014). In fact, many academics have carried out studies about the political discourse in the context of OSN, the majority of which mainly focused on two concerns: which users are the most "influential" (key-members identification) and how information flows throughout the network (as cited in Bode and Dalrymple,

2016). Previous work on information diffusion and key-members identification in social networks has been traditionally based on centrality measures (Freeman, 1978). However, we believe that further investigation is needed to better grasp how network characteristics, such as centrality, differ per topic in online conversations.

In this paper, we argue that the above-mentioned network-level measurements may systematically differ among topics of conversation on OSN. We are of the view that analyzing such topic-network structures might help to better understand information dissemination. We ground our work on political conversations because of their intrinsically polarizing, and easily identifiable topics. More specifically, we focus on the 2019 European Election event because of its presence in multiple countries. Twitter can be considered suitable for these research purposes. First, as one of the foremost social network platforms that is globally used and steadily growing, it established itself as one of the most popular online political arena (Tumasjan et al., 2011) and empowered politicians to share their messages broadly without the need of journalists (Blumler and Gurevitch, 2001). Moreover, by providing open access to its API – with a free tier available too – it offers researchers an unprecedented opportunity to easily collect data. The retweet feature allows its users to spread information beyond the reach of their original followers, and therefore enables us to clearly reconstruct the conversation trees. Lastly, Toubia and Stephen (2013) also showed that Twitter's design pushes its users to voluntarily contribute with content because of the intrinsic utility and the image-related utility, de-facto disclosing their personal information, thoughts, and experiences by tweeting, and thus providing a constant and sheer volume of data. In this study, we collect more than 20 million tweets on the 2019 European Parliament election, over the entire electoral period, from one week before (May 16, 2019) up to more than a week after the elections in all the Member States. This is a *sample* of the stream of tweets in that time period, resulted from the tracking of approximately 700 different keywords[1] via the Twitter real-time filter API[2]. We track eleven politics-related topics, chosen because of their relevance and omnipresence in most countries, with the following keywords[3]: `brexit`, `eu institutions`, `fake news`, `gender equality`, `lgbt`

[1] A complete list of tracked keywords can be found in the Appendix section.
[2] Twitter's APIs and their limitations will be thoroughly discussed in section 3.
[3] All the keywords have been accordingly translated into all the different languages spoken in the European Union.

`rights`, `populism`, `public debt`, `refugees`, `single market`, `terrorism`, `unemployment`. Subsequently, we select our data to retain only contributing users to a given topic – i.e., users whose tweets initiated a conversation, as well as their retweets, replies, and @-mentions. We construct topic-network structures by obtaining nodes and edges for each topic, and calculate a set of network measurements via Gephi 0.9.2[4]. We propose three generalized regression models to explain how information flow characteristics differ among topics, and apply them to four different topic-networks. We find evidence for systematic differences in the direction of the effect of centrality measures between distinct topics of discussion, but not for degree and clustering metrics.

Therefore, this paper will be structured as follows: in section 2 we deepen into the extant literature. Our data collection and modeling methodology is outlined in section 3. In section 4, we explain our models and analysis. In section 5, we discuss the results. Section 6 concludes this paper.

[4]Gephi is a visualization and exploration software for graphs and networks that is open-source and free: https://gephi.org/.

2 Literature Review

A vast body of research has focused on SNA applied to online social networks. Because of the nature of this paper, we put emphasis on three streams of literature that deal with the topological analysis of social networks, while we neglect the numerous studies devoted to other disciplines – e.g., linguistics, mostly focused on text analysis, user generated content, etc. Our classification of the current literature is summarized in Table 1 on page 7.

The first group of studies addresses information diffusion, namely how information propagates throughout the network. Many articles in this group focus on the issue of social influence, that is the change in behavior of individuals attributable to other actors in a network. The strength of such social influence may depend on many factors, such as tie strength, the distance between users in the network, etc. (Aggarwal, 2011). For instance, researchers in this group use Twitter data to investigate the influence of word of mouth (WOM) on consumer demand for new products (Deer et al., 2019), as well as social influence of important actors – called middle-level gatekeepers, who have between 1,800 and 26,000 followers on Twitter – in the spreading of viral events (Hemsley, 2019). Similarly, Aral et al. (2007) show how demographics, network factors, functional relationships, and the strength of ties influence the diffusion of news, compared to the diffusion of discussion topics. They find that, while demographic and network factors always heavily influence diffusion, tie strength only does so in the diffusion of discussion topics. Unfortunately, this study is based on email data and the question remains on whether the same conclusions apply to other OSN, such as social media platforms like Twitter. Lerman and Ghosh (2010) conduct a similar study on news propagation based on Digg and Twitter data sets, but without controlling for the effect of network factors such as centrality. Likewise, Liu-Thompkins (2012) concentrates on seeding strategies to spread viral messages and shows that choosing highly influential users with strong ties is a better strategy than simply opting for a wider reach. Lastly, Zhang et al. (2013) model the propagation probability based on topic relevance to the target message, although without controlling for the effect of network metrics. Unfortunately, all the approaches adopted by this group of studies are limited in showing the interaction of topics and network factors.

The second set of papers deals with community identification in OSN. Studies

on community detection highlight the structure of relationships in the network and identify the users with a particular position. For instance, Grandjean (2016) tries to answer to the question of "who's following who?" in a descriptive network analysis. Similarly, Java et al. (2007) document users intentions associated in a community and find that users with similar intentions tend to connect more with each other. Zhao et al. (2012) take a step closer to the analysis of topic-network structures and propose a topic-oriented community detection approach. However, a gap remains in our knowledge regarding how these topical clusters differ on a network metrics basis, and whether such differences can be explained and uniquely attributed to the distinct topics of discussion on online social media platforms.

The third stream of research, in which we position our own work, examines topic-network structures in the context of OSN. These studies center around the spreading of topics through the graph, taking into account different network factors and tracking the structure of information spread as if it was an "infection"'s growth, drawing inspiration from epidemiological studies. Ardon et al. (2013) perform a large scale measurement study, observing temporal, spatial, and geographical evolution of both popular and less popular topics. However, their approach remains topic-centered and does not investigate the interaction of users. One method to address this issue is to model the strength of topic-level direct and indirect influence between nodes in a network, and later apply it to predict user behavior (Liu et al., 2010; Tang et al., 2009; Lim et al., 2016). Some researchers adopt a different approach and, instead of relying on hashtags, they infer the topics of discussion from a text analysis, where an unsupervised machine learning technique is used to identify latent topic information (Hong and Davison, 2010; L'huillier et al., 2011). Unfortunately, these studies cannot be taken as evidence for the existence of a causal link between the differences in network measures and the interacting topics of discussion. There are several ways of handling this problem. Many scholars tried to characterize information diffusion in entire networks using a single network measurement only - e.g., density, or centrality. Himelboim et al. (2017) recognize the pitfall and overcome this difficulty by classifying Twitter conversations based on multiple network-level measurements and patterns of information flow. In fact, the authors claim that to gain insights on information flow for the whole network it is crucial to integrate single network-level measurements with one another.

Nevertheless, their findings do not imply that the observed differences can be uniquely attributed to the distinct topics in question, as no causal relationship is assessed.

The contribution of this paper to the literature is twofold. First, we aim at reconstructing topic-network structures to subsequently test the differences of multiple network-level measurements among topics with three generalized linear models. As a second contribution, we propose an easily reproducible methodology for data collection and modeling, that could serve as a blueprint for similar large-scale social network analyses, and facilitate the replication of a real-time data collection and visualization environment.

Table 1: Current literature overview of topological SNA applied to online social networks.

Paper	Effect of				Outcome measure(s)	Topic(s) control	Text data	Data size
	degree	tie strength	centrality	clustering				
Information diffusion								
Aral et al. (2007)		✓	✓		information awareness	✓		125,000 emails
Deer et al. (2019)					consumer demand for movies		✓	50M tweets
Hemsley (2019)					information spreading			11,000 tweets
Lerman and Ghosh (2010)	✓				information spreading	✓		137,582 Twitter users
Liu-Thompkins (2012)	✓	✓			# of views per video			101 YouTube videos
Timm et al. (2016)	✓		✓		–[a]	✓	✓	51,000 tweets
Zhang et al. (2013)	✓				propagation probability	✓		10,892 Twitter users
Community identification								
Grandjean (2016)	✓		✓		–[b]			2,500 users
Java et al. (2007)	✓				–[b]			76,177 users
Zhao et al. (2012)	✓	✓			purity of topics in community	✓		275,332 emails, 1,490 webblogs, 2,708 papers
Topic-network structures								
Ardon et al. (2013)	✓		✓	✓	topic popularity	✓		10M users, 5.96M topics
Himelboim et al. (2017)			✓		–[c]	✓		60 topics
Hong and Davison (2010)					–[d]	✓	✓	1,992,758 tweets, 514,130 users
L'huillier et al. (2011)			✓		–[d]	✓	✓	29,057 posts
Lim et al. (2016)					–[e]	✓		60,370 + 781,186 tweets
Liu et al. (2010)	✓	✓			influence strength	✓		40,000 users
Tang et al. (2009)	✓	✓			interacted, # of interactions	✓		2,329,760 papers, 640,134 authors, 18,518 films
THIS PAPER	✓		✓	✓	interacted, # of interactions	✓		21,337,037 tweets

[a] They use an agent-based actor model to run a social simulation. [b] They apply algorithms such as Clique Percolation Method (CPM) to find overlapping communities. [c] They use the Clauset–Newman–Moore clustering algorithm included in the NodeXL package. [d] They use the Latent Dirichlet Allocation technique to run a text analysis. [e] They use hierarchical Poisson-Dirichlet processes (PDP) for text modeling.

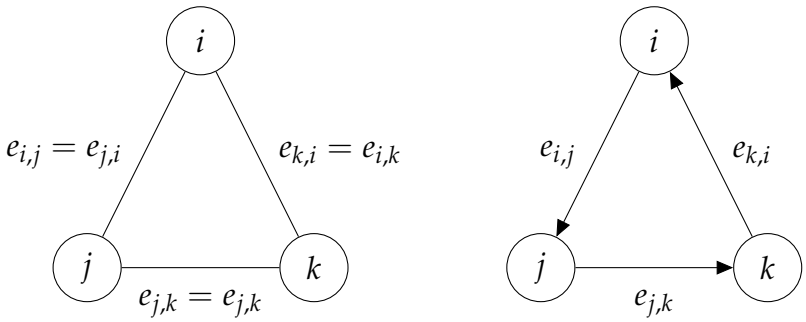
3 Data and Methodology

3.1 Research Design

We are interested in assessing how Twitter network structures differ among distinct topics of discussion. We explore the differences between such topic-network structures in terms of their network topology measures and how these affect the number of interactions between pairs of users. For this purpose, we first need to collect the data and retrieve tweets conversations based on selected topics. Next, we perform a regression analysis on these networks, where the predictors are the nodes' respective network measures, and the explained variable is the number of interactions between every pair of nodes in the network. Before explaining our methodology in more detail, we set out some basic definitions of fundamental concepts of graph theory and social network analysis.

A social network is a network (or graph) of *interactions* or *relationships* between actors (Aggarwal, 2011). The actors are called *nodes*, whereas their interactions are called *edges*, links, or ties. Social networks are not restricted to a purely online form. In fact, they have been studied by sociologists for generations before the advent of OSN (see, e.g., Milgram, 1967). However, in the context of OSN, users are often the nodes who form ties or connections (edges) among themselves to exchange content like text, images, videos, or in another form. In this paper, the nodes represent the interacting users and the edges their tweets, retweets, mentions, or replies.

Figure 1: Nodes and edges in undirected and directed graphs.



Undirected graph Directed graph

A network can be *directed*, like follow relationships on Twitter, or *undirected*, like

mutual friendships on Facebook (see Figure 1 on the previous page). In directed graphs, asymmetric binary relations may occur, hence the edge from node i to node j is different than the one from j to i. Moreover, a network may be *weighted*, where a weight is set for each edge or, whenever multiple edges can occur between two nodes (e.g., in a conversation of tweets), the weight can be the number of edges between these nodes. In our case, in a weighted graph $\mathcal{G}$, let $e_{i,j}$ be the edge between node i and node j, and let $w_{i,j}$ be the weight on edge $e_{i,j}$ computed as the number of interactions between i and j. The total weight w_i of the node i can be thus defined as the sum of weights of all its edges, that is $w_i = \sum_{k=1}^{d_i} w_{i,k}$, where d_i denotes its degree. The *degree* of a node is defined as the number of edges for that given node – i.e., how many connections a node has in the network. The weighted degree is similarly based on the number of edges for that node, but ponderated by the weight of such edges – i.e., the number of interactions between a pair of nodes. One final notable network concept is related to *density*, which stands for the proportion of edges present in a network compared to the number of all potentially possible edges - i.e., the completeness of a network. Density is therefore defined as $Network\ Density = \frac{No.\,of\ actual\ edges}{No.\,of\ potential\ edges} = \frac{No.\,of\ actual\ edges}{\frac{N\,(N-1)}{2}}$, where N is the total number of nodes in the network.

To perform our study, we require nodes and edges tables for each topic of discussion. The former are tables with data for each node, while the latter contain information on the edges between each pair of nodes. In our case, nodes tables include network topology measures for each node, while edges tables specify which are the source and target nodes, and the weight of their edge. We provide examples of the structure of these tables in Appendix A.3 to A.5 on pages 41–42.

The process to obtain the final data set for each topic consists of four main sequential stages. First, primary data is collected in real-time via the Twitter API endpoints, then stored on a MongoDB (NoSQL) database for real-time querying and visualization, as well as on an Amazon Web Services (AWS) S3 bucket for long-term storage. Secondly, data is cleaned, selected, and prepared with a five-step pipeline in Python. These scripts transform the raw data set into nodes and edges tables for each analyzed topic. Subsequently, the latter tables are imported to Gephi in order to visualize their graph structures and compute network measurements. Lastly, we import into an R program the updated tables with new measures exported from Gephi. This R script constructs

adjacency lists and runs three generalized linear models for each topic, where the covariates are node-level network metrics. First, we run a multiple regression model on the number of interactions for every pair of users. Secondly, a logistic model to assess the probability of interaction between two users. Finally, a conditional (filtered) multiple regression model on the number of interactions, only for those pairs of users that interacted with each other.

In this chapter, we further illustrate the rationale behind each step in the pipeline. To enable reproducibility of our research, we publish all our scripts on GitHub with open access[5].

3.2 Data Collection

3.2.1 Twitter Real-time Filter API

We conduct all the analyses in this paper on primary data collected via the Twitter real-time filter API[6] (standard plan) – or streaming API. We commence the collection of data on May 16, 2019, at 12:50:00 UTC, using the `tweepy`[7] library for Python to access the Twitter API.

The API returns public tweets that match one or more filters (e.g., a keyword, hashtag, username, or location) in a JSON response format[8]. The latency from the tweet publication to its delivery on the API is typically within seconds. The default access level (standard plan) to the Twitter API is free but subject to some limitations: it allows for tracking up to 400 keywords and returns at most 1% of all tweets being published on Twitter at a given moment (Morstatter et al., 2014). Because of the limited resources available for this project, we cannot set up an enterprise plan and utilize the PowerTrack API[9], which provides full-fidelity data and reliable connectivity instead. However, researchers have shown that the resulting data sets from the streaming API are a representative sample and truthfully reflect the activity patterns of Twitter users (Morstatter et al., 2013, 2014; Wang et al., 2015).

[5] The repository can be accessed from: https://github.com/andreantonacci.

[6] https://developer.twitter.com/en/docs/tweets/filter-realtime/overview.

[7] https://www.tweepy.org.

[8] A full list of JSON tweet objects and a tweet data dictionary can be found here: https://developer.twitter.com/en/docs/tweets/data-dictionary/overview/tweet-object.

[9] https://developer.twitter.com/en/docs/tweets/filter-realtime/overview/powertrack-api.

To overcome the limitation of 400 keywords and ensure a continuous flow of tweets, we run our script on two Amazon Elastic Compute Cloud (Amazon EC2) instances[10]. Each instance executes the same script 24 hours a day but tracks a different set of keywords and connects to the API with a different Twitter developer account – i.e., with a different API key and access token. In this way, we can track up to 800 keywords but run the risk of collecting the same tweet twice – i.e., in the case that a tweet contains two or more hashtags, each one tracked on a different EC2 instance, both of the machines may collect it. We address this issue a posteriori, as illustrated in section 3.2.2 on the following page.

A second limitation of this approach is that the data set may contain only a portion of a conversation, which happens if the initial tweet (called *seed* or conversation starter) is not part of the collection returned by the API. In such a case, we treat the earliest tweet available in the conversation tree (a tweet replying to another one that is not part of our data set) as the initial tweet.

The collection process stopped on June 2, 2019, with a total of 21,337,037 unique tweets. We tracked approximately 800 different keywords, which include:

- General hashtags on the elections (in all the different languages spoken in the EU): e.g., #EUElections, #EUvaalit2019, #EP2019;
- All the European Parliament groups' handles and official hashtags: e.g., @ALDE, #ALDE, @EPP, #EPP;
- All the major national parties' handles (i.e., those which have at least a seat in the national parliament): e.g., @groen, @PdA_Austria, @LegaSalvini;
- All the national party's leaders' handles: e.g., @theresa_may, @luigidimaio, @pablocasado_;
- Keywords on popular politics-related topics and challenges faced by the EU at that time (in all the different languages spoken in the EU): `brexit`, `eu institutions`, `fake news`, `gender equality`, `lgbt rights`, `populism`, `public debt`, `refugees`, `single market`, `terrorism`, `unemployment`.

[10]The instance type is `m5.xlarge`, powered by Intel Xeon Platinum 8175 processors, 4 vCPU, and 16GB of memory.

A complete list of tracked keywords and a logbook of events that occurred during the collection process can be found in Appendix B on page 51.

3.2.2 Electionstats: MongoDB and Real-time Visualization

Electionstats[11] is a side project of this research study. It consists of a visualization dashboard of insights retrieved from the collected tweets during the electoral period that updates in real-time. Because the objective of this side project is outside the scope of the present study, we do not deepen into its characteristics[12]. Nonetheless, it is necessary to outline how Electionstats affects our data collection and storage strategy.

Storing data to a secure cloud service seems the most efficient solution if there is not the need for live data querying. Instead, in this case, we first add an intermediate step: we import our data to a NoSQL database (MongoDB[13]), which allows us to run the (almost) real-time[14] queries needed for the live dashboard. To do so, we write the raw data returned from the API to a new JSON file every five minutes. Another Python script then scans the directory where the files are saved, parses the tweets, and imports them to MongoDB only if their file is older than five minutes and their tweet ID does not exist in the database already. Lastly, it also uploads the raw files to an Amazon Simple Storage Service (Amazon S3) bucket and removes the file locally, if the upload is successful.

3.2.3 Long-term Storage on Amazon S3

Once the official electoral results are published, and our collection is complete, there is no need for further updates of the Electionstats dashboard, and thus for a MongoDB database. We export all data from the database to three equally sized JSON files[15] and upload them to an encrypted Amazon S3 bucket for long-term storage. We conduct all analyses in this study on derived data from a merged file (master data file) that is the sum of these three.

[11] The dashboard can be accessed from: `https://electionstats.eu`.

[12] However, we make the scripts available in the project's repository.

[13] The cluster tier is M30, with 7.5GB of RAM, 101GB of storage, encrypted, with MongoDB 4.0 and hosted on the Google Cloud Platform.

[14] Or rather, frequent ones.

[15] For simplicity, we refer to them in tables and figures as, respectively, `eu2019`, `eu2019v2`, and `eu2019v3`.

3.3 Data Hygiene and Preparation in Python

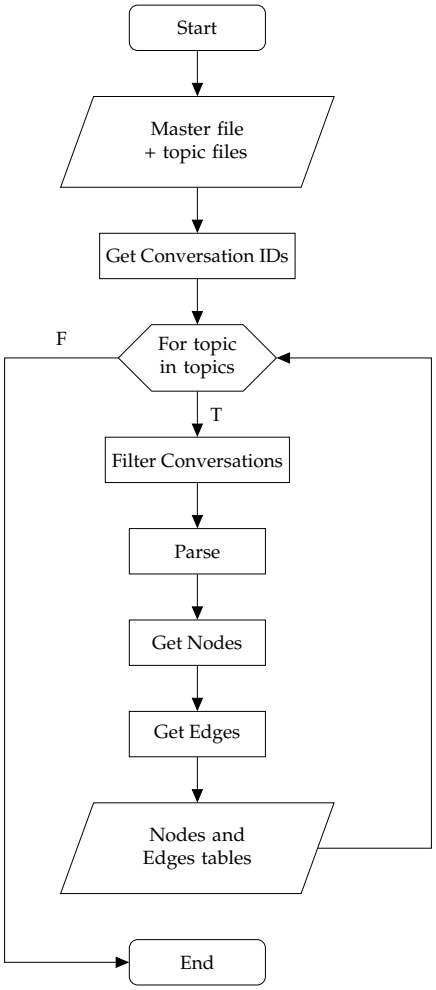
We perform our data selection, modeling, and analysis on a single server with Windows Server 2012 R2 Standard, two Intel Xeon Gold 6126 CPU at 2.60GHz, and 512GB of RAM available. Because the resources needed for a computer cluster architecture are too expensive for this project, we must first reduce our data to enable further wrangling and analysis.

The algorithm employed to sample tweets from the master data file retains one line out of ten lines. This sampling method ensures that the resulting data set is still a representative sample, because the tweets in the master data file are ordered by timestamp, and the retained ones encompass the entire electoral period. The final sampled data set is a tenth of the size of the original master data file (16.22GB), with 2,259,717 unique tweets.

Subsequently, we employ a five-step pipeline to prepare our data and obtain nodes and edges tables for each topic. The overall process is run in Python 3 and illustrated in the flowchart (see Figure 2 on the following page). The first algorithm creates a list of unique tweets – identified by their ID – that are part of a conversation (see Algorithm 1 on page 15 with the pseudocode). We define such tweets as those that either started a conversation (seeds) – i.e., tweets that have been retweeted or replied to by another user at least once – or extended one by retweeting, quoting, or replying previous users in the sequence. This task is remarkably easy because we do not need to reconstruct the entire conversation tree. We simply filter all the tweets that are either a retweet, a quote, or a reply. Then, we store both their tweet ID and the ID of the tweet they are interacting with. In this way, we are sure to retain the seed IDs too because they must emerge from their retweets, quotes, and replies.

All topics share this first step of the pipeline, and its script is invoked only once. The subsequent algorithms are executed for each topic instead. The computational complexity of our analysis in R (see section 4 on page 25) sets a limit upon the number of topics we can analyze. Hence, we select those we deem the most relevant and equally widespread in all the Member States. Because of the different relative importance in the public opinion of different States, we exclude the following topics from further analysis: `eu institutions`, `lgbt rights`, `public debt`. We also dismiss `gender equality` and `single market` because too few tweets appear in our sample – only 17 and 6 nodes,

Figure 2: Data wrangling pipeline in Python.



respectively – and `ep2019` because it does not identify a discussion topic. Instead, it refers to an event whose discussions do not revolve around a single topic. Finally, we exclude `fake news` due to technical difficulties. As a consequence, we perform our analysis on the following five discussion topics: `brexit`[16], `populism`, `refugees`, `terrorism`, and `unemployment`. Appendix A.2 on page 40 provides an example of a topic file, which is a list of the tracked keywords for that specific topic.

The second algorithm selects only contributing users to a specified conversation filtered by topic (see Algorithm 2 on page 16). In order to further progress through the pipeline, a tweet must thus satisfy two conditions. Its tweet ID must be in the list of IDs that take part in a conversation, and at least one of its hashtags must

[16]We further sample `brexit` nodes (but not their respective edges and target nodes) because of the large volume of tweets collected. The final data set contains 4,843 nodes and 9,226 edges.

Algorithm 1 Get Conversation IDs

```
1: Input: master file
2: Output: conversation ids file
3: Initialize toBeExported as empty array
4: read allTweets from master file
5: for tweet in allTweets do
6:     if tweet is not valid then                          ▷ skip empty lines
7:         skip
8:     end if
9:     if tweet is a retweet or is a quote or is a reply then
10:        toBeExported ← tweet.id
11:        if tweet is a retweet then
12:            toBeExported ← tweet.seedId
13:        end if
14:        if tweet is a quote then
15:            toBeExported ← tweet.seedId
16:        end if
17:        if tweet is a reply then
18:            toBeExported ← tweet.seedId
19:        end if
20:    end if
21: end for
22: save toBeExported to conversation ids file
```

be in the list of tracked keywords for a given topic note. More precisely, a perfect match between the hashtag and the tracked keyword is not required, because the tracked keyword can also be part of the hashtag. For instance, a tweet with the hashtag `#IranianRefugeesInTurkey` can also satisfy this condition because it contains the tracked string `refugees`. As the outcome of this script, we obtain a filtered JSON file for each topic, which still contains full information for the selected tweets.

In the third step, we parse each JSON file to maintain only relevant fields and further reduce the file size. We retain selected fields, as shown in Algorithm 3 on page 17, and write the data on a CSV file for each topic.

The fourth and fifth algorithms produce, respectively, the nodes and edges tables for a given topic. They make use of two Python libraries, `pandas` and `numpy`, to wrangle the data structure. Algorithm 4 on page 17 obtains a list of the unique IDs of all interacting users (nodes) for each topic, including user IDs, retweet user IDs, quoted user IDs, mentioned user IDs, and replied user IDs. Algorithm 5 on page 18 computes the number of interactions for every pair of users in a given data set and constructs an

Algorithm 2 Filter Conversations

```
Input: master file, conversation ids file, current topic file
Output: filtered tweets file
initialize toBeExported as empty array
read conversationIds from conversation ids file ▷ all tweets part of a conversation
read topics from current topic file
read allTweets from master file
for tweet in allTweets do
    if tweet is not valid then
        skip
    end if
    Initialize hashtagsList as empty array
    if len(tweet.hashtags) > 0 then ▷ look for hashtags only if present
        for hashtag in tweet.hashtags do
            hashtagsList ← hashtag
        end for
    end if
    if tweet.id in conversationIds then ▷ retain only matching tweets
        for hashtag in hashtagsList do
            for topic in topics do
                if topic in hashtag then
                    toBeExported ← tweet
                end if
            end for
        end for
    end if
end for
save toBeExported to filtered tweets file
```

adjacency list where this number is the weight for the edge between the source node and the target one.

Due to the previous steps in the pipeline, all users interacted with at least another user, and therefore no edge in the resulting file has a weight equal to zero. Since we require a complete edge table in the following analysis, we could have also included users that did not interact, with their weight set to zero. Nevertheless, we prefer the adopted approach because it produces smaller file sizes that are easier to handle. Therefore, we integrate these tables in R at a later time (see section 3.4.2 on page 20).

As a result of this process, we retrieved nodes and edges tables for each topic that we will utilize in further analyses, as explained in the following sections.

Algorithm 3 Parse JSON file and convert to CSV

Input: *filtered tweets file*
Output: *parsed tweets file*
read *filteredTweets* **from** *filtered tweets file*
for *tweet* **in** *filteredTweets* **do**
    if *tweet* **is not** valid **then**
        skip
    end if
    write line to *parsed tweets file* with fields:
    tweet.timestamp
        .id
        .userId
        .userScreenName
        .retweetId
        .retweetUserId
        .retweetUserScreenName
        .quoteId
        .quoteUserId
        .quoteUserScreenName
        .replyToTweetId
        .replyToUserId
        .replyToUserScreenName
        .userMentionsId
        .userMentionsScreenName
end for

Algorithm 4 Get Nodes

Import libraries: `pandas`, `numpy`
Input: *parsed tweets file*
Output: *nodes file*
load *df* **from** *parsed tweets file*
$dfSeeds \leftarrow df.userId, df.userScreenName$
$dfRetweets \leftarrow df.retweetUserId, df.retweetUserScreenName$
$dfQuotes \leftarrow df.quotedUserId, df.quotedUserScreenName$
$dfMentions \leftarrow df.userMentionsId, df.userMentionsScreenName$
for $dfSeeds, dfRetweets, dfQuotes, dfMentions$ **do**
    rename 1st column to *Id*
    rename 2nd column to *Label*
end for
$dfOutput \leftarrow$ concatenate $dfSeeds, dfRetweets, dfQuotes, dfMentions$
drop duplicates from $dfOutput$
save $dfOutput$ to *nodes file*

Algorithm 5 Get Edges

```
Import libraries: pandas, numpy
Input: parsed tweets file
Output: edges file
procedure CREATEORINCREMENT(source, target)                    ▷ function later used
    if target is null then
        return
    end if
    match ← dfEdges where source column = source and target column = target
    if match is empty then
        edge data ← [source, target, 1]                        ▷ new row with weight = 1
        append edge data to dfEdges
    else
        append weight + 1 to match                             ▷ increment weight to existing match
    end if
end procedure

load df from parsed tweets file
initialize dfEdges as empty df with columns [source, target, weight]
for row in df do
    CREATEORINCREMENT(row.userId, row.retweetUserId)
    CREATEORINCREMENT(row.userId, row.quotedUserId)
    CREATEORINCREMENT(row.userId, row.replyToUserId)
    for mention in row.mentions do
        CREATEORINCREMENT(row.userId, mention)
    end for
end for
save dfEdges to edges file
```

3.4 Variables Operationalization

3.4.1 Nodes Measures with Gephi

For every one of the five chosen topics, we import the relative nodes and edges tables as a directed and weighted graph in Gephi. Then, we draw the graphs with the original Yifan Hu's attraction-repulsion model[17](Hu, 2005). Once the algorithm stops itself, we compute node-level network measures that we will include later in our regression models as explanatory variables.

Degree centrality is a centrality measure, often used to assess the relative node's importance in a network. Centrality is one of the most popular and studied tools in

[17]The Yifan Hu's algorithm is used with Optimal Distance: 100; Relative Strength: 02; Initial Step size: 20; Step ratio: 0.95; Adaptive Cooling checked; Convergence Threshold: 1.0E-4.

SNA (e.g., Freeman, 1978). Over the years, many metrics have been proposed, and the degree centrality is merely one of the simplest. In fact, it is a volume-based measure simply defined as the node's degree: $c_i^{DEG} = deg(i)$. In other words, it only refers to how well a node is connected in the network, regardless of its role within it.

A more sophisticated approach to measuring centrality is based on the count of the length of walks. A *walk* is a form of connection between two nodes, that is the sequence of nodes and edges between these two. A *path* is a particular kind of walk in which each node and edge must be crossed at most once. *Closeness centrality* is one of the best examples of this group of measures. It is defined as the average of the shortest path[18] length from one node to every other node in the network: $c_i^{CLO} = \frac{1}{\sum_j d(j,i)}$, where $d(j,i)$ is the distance between the nodes. Hence, the closer a node is to all the other ones, the more central it is, and the lower its closeness centrality value is.

Harmonic centrality is an alternative to closeness centrality that is also useful in unconnected graphs and thus has similar use cases. It reverses the operations in the definition of closeness centrality, and it is therefore defined as the sum of the inverse distances of a node to all other nodes: $c_i^{HAR} = \sum_{i \neq j} \frac{1}{d(j,i)}$. We utilize the normalized version of this measure that is divided by $N-1$, where N is the total number of nodes in the network.

Eccentricity centrality is defined as the reciprocal of a node's eccentricity, which is the maximum distance between i and any other node in the network: $c_i^{ECC} = \frac{1}{e_i} = \frac{1}{max\{d(i,j):j \in \mathcal{G}\}}$, where $\mathcal{G}$ is a given graph, and the maximum distance between i and any other $j \in \mathcal{G}$ is the longest shortest path between them. Consequently, if the eccentricity of i is high, it means that all the other nodes and their neighbors are in proximity.

A more popular metrics based on shortest paths is *betweenness centrality*. It is defined as the percentage of shortest paths that pass through a given node: $c_i^{BET} = \sum_{x \neq y \neq i} \frac{\sigma_{xy}(i)}{\sigma_{xy}}$, where σ_{xy} is the number of shortest paths between the nodes x and y, and $\sigma_{xy}(i)$ is the number of those that pass through i. This widely adopted measure allows identifying highly influential nodes in a network, because more shortest paths – and thus more interactions, connections, information, etc. – pass through that node.

One last pivotal centrality measure is *eigenvector centrality* (or eigencentrality). It is a more sophisticated measure of node's importance that assigns a higher score to nodes if

[18] One that minimizes the number of edges through nodes.

they are connected to more influential ones. Put differently, it assumes that the number of interactions is not the main criterion of importance. Instead, interactions with other highly influential nodes contribute more than those with less influential ones. As a consequence, a higher eigencentrality value means that the node is connected to more nodes that have a high value themselves too. We can define the eigenvector centrality of node i as follows: $c_i^{EIG} = \frac{1}{\lambda} \sum_{j \in M(i)} c_j^{EIG} = \frac{1}{\lambda} \sum_{j \in \mathcal{G}} a_{i,j} c_j^{EIG}$, where $\mathcal{G}$ is a given graph, $a_{i,j}$ is the adjacency matrix, $M(i)$ is the set of neighbors[19] of i, and λ is a constant.

The *clustering coefficient* is a measure of the degree to which nodes tend to cluster together in a network (Watts and Strogatz, 1998). In other words, it indicates how the nodes are embedded in their neighborhoods. The clustering of a single node i is defined as the proportion of edges in the node's neighborhood divided by the number of all potentially possible edges between the neighbors: $C_i = \frac{e_i}{k_i\,(k_i - 1)}$, where k_i is the number of neighbors of i and assuming a directed graph. The average of local clustering coefficients (those of the single nodes) gives an overall indication of the clustering in the network – i.e., the clustering coefficient of the graph.

Finally, we also measure *modularity* in our networks, which is a measure of the division of a network into communities. A highly modular network has very dense communities – i.e., many edges between nodes within a community – but spare edges between nodes of different communities. The modularity values in our data set merely represent the label of the community that a given node is part of (modularity class). This information would be useful to filter further our network into sub-modules. However, we do not investigate the single communities within each topic-network, and therefore we exclude this variable from our regression models.

3.4.2 Data Wrangling in R

Once we export the updated nodes tables that contain node-level network measurements, we import them, together with their respective edges tables, into an R program. Before performing the regressions, we merge the information from both nodes and edges tables to construct complete adjacency lists for each topic. In other words, the final data set has for each row the ID, label, and network topology information for both the source (that we call User 1, or U1) and the target node (U2), as well as the

[19] A set of neighbors $M(i)$ of the node i is defined as a set of its immediately connected nodes.

number of interactions between them (the former edge's weight). By doing so, we can run a regression on all possible pairs of users for a given topic-network and not only with those that interacted with each other. However, the computational complexity of this task grows considerably as the number of observations in the regression models increases to $N^2 - N$, where N is the number of nodes in a topic-network structure. Lastly, we mean center all variables[20] add a new dummy variable, coded 1 when the number of interactions is greater than zero, and 0 otherwise.

3.5 Descriptive Statistics

From a sampled data set of 2,259,717 unique tweets, we obtain five sub-networks of users based on their topic of discussions: `brexit` (with 4,843 nodes and 9,226 edges), `populism` (with 704 nodes and 1,373 edges), `refugees` (with 2,952 nodes and 13,949 edges), `terrorism` (with 2,033 nodes and 4,521 edges), and `unemployment` (with 1,406 nodes and 3,332 edges). Table 2 on the next page provides some summary statistics on the data set for the `refugees` topic. Similar tables for all the other topic-network data sets can be found in Appendix A.6 to A.10 on pages 43–45.

Since we work on primary data computed ad hoc, we are confident to have a complete data set and no issues of missing values. Hence, we do not perform a Missing Value Analysis.

Figure 3 on the following page displays the bivariate (linear) correlations between all pairs of variables in our data set for the topic `brexit`. Similar matrices for the other data sets can be found in Appendix A.1 to A.4 on pages 46–47. All correlations are statistically significant at the 0.01 level. We notice strong correlations within the group of degree variables, as well as within centrality measures, and between these two groups. For instance, from table 3 on the next page we see that *degree* is strongly and positively correlated with *indegree* (.87), similarly to *closeness centrality* and *eccentricity* (.81), and that *closeness* is also positively correlated to *outdegree* (.60). Since many of these variables measure the same underlying concept but in different ways, we suspect some serious multicollinearity issues. Therefore, we only include in our models a selection of these metrics, as discussed in section 4.1 on page 25.

[20] Except for the user IDs and character variables.

Table 2: Summary statistics for topic: `refugees`.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
indegree	2,952	4.725	16.975	0	0	2	430
outdegree	2,952	4.725	14.191	0	0	4	265
degree	2,952	9.451	24.953	1	3	5	430
weighted_indegree	2,952	9.395	49.085	0	0	3	1,614
weighted_outdegree	2,952	9.395	46.012	0	0	5	1,120
weighted_degree	2,952	18.791	81.807	1	4	7	1,864
eccentricity	2,952	2.932	3.238	0	0	7	10
closness_centrality	2,952	.358	.378	0	0	.643	1
harmonic_closness_centrality	2,952	.371	.382	0	0	.750	1
betweeness_centrality	2,952	665.544	6,589.509	0	0	0	213,888.300
clustering	2,952	.094	.177	0	0	.160	1
eigencentrality	2,952	.026	.081	0	0	.003	1

Number of nodes = 2,952; Number of edges = 13,949.

Figure 3: Bivariate correlation matrix for topic: `brexit`.

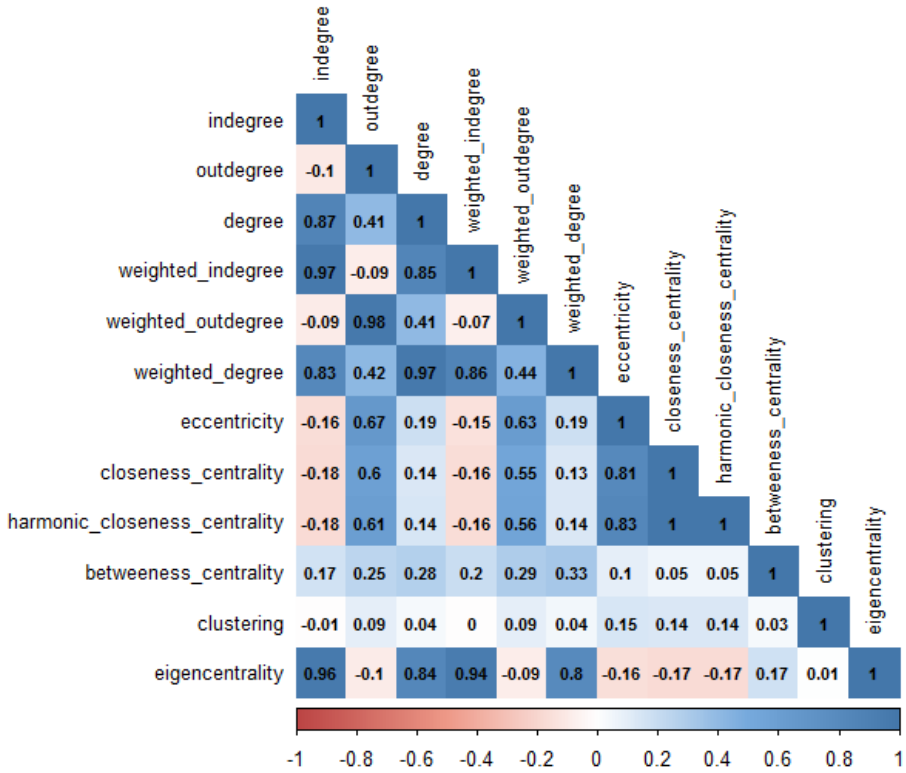
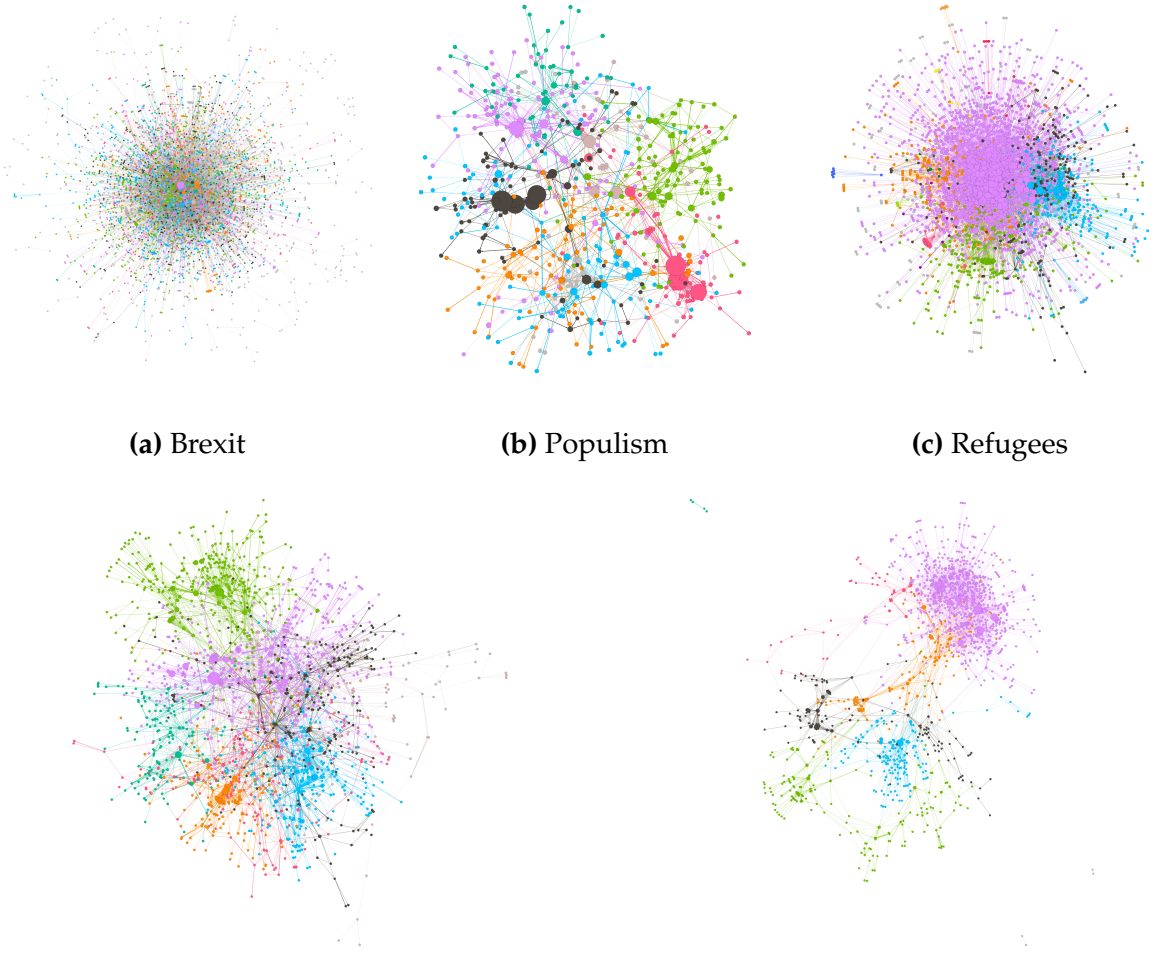


Figure 4: Network graphs per topic.



(a) Brexit **(b)** Populism **(c)** Refugees

(d) Terrorism **(e)** Unemployment

3.6 Graphs Visualization

While mathematical metrics are essential to test hypotheses and draw sound conclusions, only a visual representation allows grasping the overall structure of a network quickly. Therefore, we show the five selected networks[21] in Figure 4 to describe the collected data visually. These topic-network structures seem to differ on many network measures. Nevertheless, we can identify some shared features.

The geographical position of nodes is determined by a force-vector algorithm[22] that simulates a system of physical forces: nodes tend to repulse each other, while edges hold back bounding nodes. The algorithm changes the disposition of nodes until it reaches an equilibrium where a balance of forces is guaranteed by minimizing the number of

[21] Only interacting nodes are included.
[22] The Yifan Hu's algorithm, in our case, as mentioned earlier.

edges crossings. In other words, two nodes are closer if they are connected with each other or to the same set of nodes. Once the equilibrium is reached, the relative position of nodes reveals clusters and structural holes – the empty zone between clusters – in the network. The color of nodes in our networks represents the community to which they belong to, determined by the modularity. Lastly, the size of the nodes represents their (eigenvector) centrality in the network.

We can classify our networks into two main groups. `Brexit` and `refugees` appear as a one, highly dense cluster, with almost no subclusters or structural holes and many sparse nodes around it. Few central individuals, called hubs, attract many connections in a star structure, where the surrounding nodes are connected to the center but not among them.

On the other hand, `populism` and `terrorism` show many small and dense subclusters, while `unemployment` has three distant and massive clusters.

4 Models

4.1 Considerations on Multicollinearity

Results from preliminary test models suggest severe multicollinearity issues due to the strong correlation between some variables. Because several variables in our data set measure similar concepts, we identify the most meaningful ones to include in our models.

We choose *eigencentrality* among the set of available centrality measures because it takes into account the extent to which a node is connected to highly influential nodes. Furthermore, we do not include other centrality variables because they can be seen as a linear transformation of a different measure – e.g., in the case of *closeness centrality* with *harmonic closeness centrality* or with the proposed degree variable.

Among the group of degree measures, we select *outdegree* because the number of outbound connections of a node can predict the overall number of interactions between that node and any other one in the network. Moreover, *outdegree* correlates weakly with the proposed centrality measure. On the contrary, *indegree* has an average positive correlation to *eigencentrality* of .87 between the five topic-networks, and therefore, despite its potential contribution, we exclude it from our regressions.

We also included interaction terms in our preliminary models – between variables from different groups, and between the same variable for the source node and the target one – but all of them caused significant multicollinearity issues (VIF > 30) and did not contribute to increasing the explanatory power of the model. Thus, we decide not to include any interaction term in our final models.

4.2 Multiple Linear Regression

We first construct a multiple linear regression model (called model A, or complete OLS) that is run for all five topic-network structures, where the explained variable is *interactions* – i.e., the number of interactions between the source and target nodes. This relationship is directed, which means that the number of interactions between nodes A and B might be different from the one between B and A. We include as explanatory variables *outdegree*, *eigencentrality*, and the *clustering coefficient* for both of the nodes in

every pair of nodes (see Equation 1). This model runs on all possible pairs of users in a topic-network and not only on those users who interacted with each other.

$$Y = \beta_0 + \beta_1 OD_{u1} + \beta_2 EC_{u1} + \beta_3 CL_{u1} + \beta_4 OD_{u2} + \beta_5 EC_{u2} + \beta_6 CL_{u2} + \epsilon \quad (1)$$

where Y = number of interactions between the source and target nodes (directed)
OD_{u1} = outdegree for the source node
EC_{u1} = eigencentrality for the source node
CL_{u1} = clustering for the source node
OD_{u2} = outdegree for the target node
EC_{u2} = eigencentrality for the target node
CL_{u2} = clustering for the target node

4.3 Logistic and Conditional Linear Regression

We build a logistic regression model (model B, or logistic) to predict the likelihood that paired users will interact with each other. Therefore, the dependent variable is the dichotomous *interacted*. The independent variables are the same as in model A (see Equation 2).

$$P(interacted) = \frac{e^{b_0 + b_1 OD_{u1} + b_2 EC_{u1} + b_3 CL_{u1} + b_4 OD_{u2} + b_5 EC_{u2} + b_6 CL_{u2}}}{1 + e^{b_0 + b_1 OD_{u1} + b_2 EC_{u1} + b_3 CL_{u1} + b_4 OD_{u2} + b_5 EC_{u2} + b_6 CL_{u2}}} \quad (2)$$

Finally, we build another multiple linear regression model (model C, or conditional OLS), which runs on a filtered subset of our data for all the pairs of nodes where *interacted* is true (1). The model equation is the same as Equation 1.

5 Results

Table 3 on page 31 displays the results from the multiple linear regression models (A) for each topic. F test is significant at 1% in all topic-networks and, unless otherwise noted, all coefficients are statistically significant at the 0.01 level too. We detect no issues of multicollinearity – VIF is never higher than 1.24 – but R^2 is extremely low. However, model fit is not a priority in the context of this research because we do not aim for an accurate prediction, but rather to assess whether a (perhaps small) reliable relationship exists among the considered variables. Even though the models do not explain much of the variation of the data, they are still significant. Indeed, trying to predict the number of interactions between two users exclusively based on their network characteristics is an unhelpful oversimplification of reality. Outcomes could be dictated by many latent factors at play that may affect other facets of the model.

The EC_{u2} predictor is the most potent one in explaining the dependent variable in models A. It seems that users are more likely to have interacted with highly influential (central) users, holding all other regressors constant. However, the magnitude of coefficients is generally very modest among these models. We suspect that this is because the effects of the vast majority of nodes that did not interact (N = 23,440,660 in the case of `brexit`) considerably mitigate those of the relatively rare nodes that did it (N = 9,146). In pursuit of more extreme coefficients, we take into account whether two nodes interacted and therefore build models B and C.

Table 4 on page 32 shows the outcomes of models B and C for each topic-network. All five logistic regressions (models B) fit the data well since the Nagelkerke's R^2 is generally close to 20%. We find no significant multicollinearity issues, and VIF scores never higher than 1.30, except for OD_{u1} (VIF = 6.74) and EC_{u1} (VIF = 6.52) in the `refugees` model but we deem these values acceptable. All coefficients are statistically significant at 1%, except for those of CL_{u2}, and the magnitude of effects is generally stronger than in the previous models.

In the `brexit` case, EC_{u1} is the regressor with the strongest partial effect on the explained variable. However, it is a rather powerful predictor in the other four models too. Thus, it seems that more central users are less likely to have initiated a conversation or replied to one compared to less central users, ceteris paribus. On the other hand, EC_{u2} coefficients are positive and significant at 1%, suggesting that users are more

likely to have interacted with central users rather than with less influential ones.

The positive and significant coefficients of OD_{u1} follow the intuitive concept that a higher number of outbound messages from the source user increases the probability of interaction between that user and any other one in the network. On the contrary, OD_{u2} estimates are generally negative and significant at 1%. Overall, our data reveal that the target node's coefficients are of opposite sign than those of the source node in all models B.

Lastly, the coefficients of CL_{u1} are positive and significant at the 0.01 level as well, indicating that, all else being equal, more clustered nodes are more likely to have interacted with other nodes. Instead, CL_{u2} is not significant, except for `refugees` and `unemployment`, where it shows a positive relationship with the outcome measure.

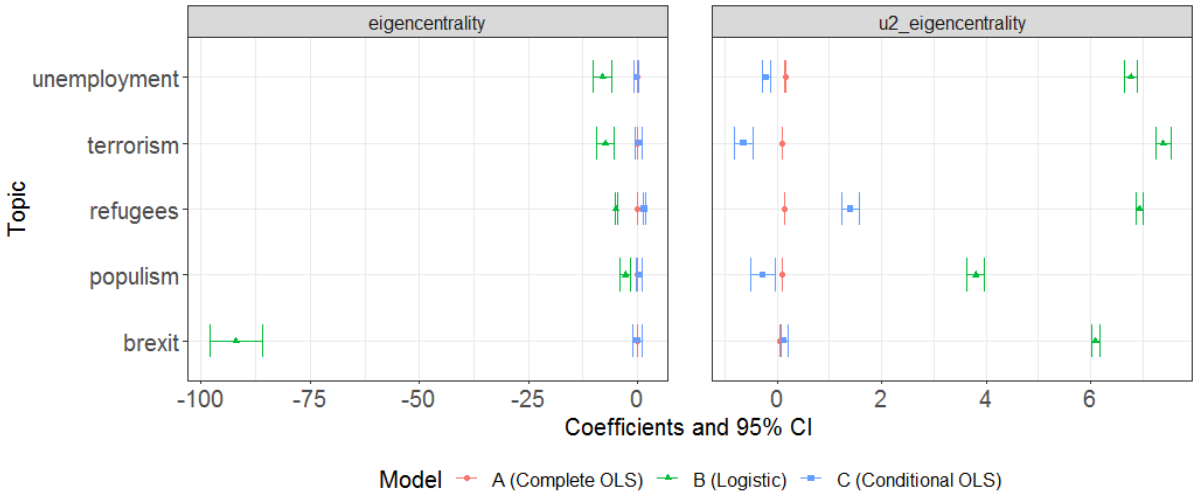
Results from models A and B indicate that node-level network characteristics may not differ between distinct topics of discussion. However, we cannot conclude this without first interpreting results from models C.

The last multiple linear regression models (C) are run on a filtered subset of our data, for all the pairs of nodes where the number of interactions is above zero. Hence, we can juxtapose the results of models C to those of models B, gaining an insight into the mechanism underlying the number of interactions between users, given that they have interacted.

The F statistic is significant at 1% for all topic-networks in models C except for `populism`, for which it is significant at 10%. R^2 is still low compared to models A, but it improves substantially for the topic `refugees` (R^2 = .201). The coefficients in these models fluctuate considerably between distinct topic-networks, and therefore it is more challenging to find a shared pattern of relationships. To compare the magnitude of effects between different topic-network structures and models for the same variable, we need to test the significance of the coefficients' differences pairwise. Because this task is painfully impractical due to the number of variables, models, and groups involved, we graph the models' coefficients and their standard errors (with a 95% confidence interval) to compare estimates visually.

Figure 5 on the next page shows these graphs for EC_{u1} and EC_{u2}. For instance, we observe that most of the EC_{u2} coefficients' standard errors and their confidence intervals do not overlap (this is particularly evident in models B and C). When the error bars

Figure 5: Coefficients and SE per topic for EC_{u1} and EC_{u2}.



EC_{u1}: We observe that the magnitude of effects varies between topic-networks in models B.
EC_{u2}: We observe that the sign of the relationship varies between topic-networks in models C.

do not overlap, the difference between models' estimates may be significant[23]. On the contrary, if error bars do overlap, we can conclude that the difference is not significant, and therefore that the paired coefficients do not statistically differ from each other. Thus, we infer that the EC_{u2}'s coefficients may statistically differ, and we speculate that this might be due to the intrinsic characteristics of the respective topic-network structures. In other words, structures with unique network characteristics – in terms of degree, centrality, and clustering – exist for different topics of discussion. Similar graphs to Figure 5 for all variables can be found in the Appendix A.5 to A.7 on pages 48–50.

In summary, we find that the *eigencentrality* of the target node (EC_{u2}) positively predicts the interaction and the number of interactions between nodes, regardless of the discussion topic. However, among those pairs of nodes that interacted, the direction of the effect of this variable on the number of interactions varies between topics. By contrast, the *eigencentrality* of the source node (EC_{u1}) negatively influences the probability of interaction, but not the extent to which pairs of users interact with each other. Therefore, we conclude that the effect of *eigencentrality* varies on whether it refers to the source or target node.

As regards the source's *outdegree* (OD_{u1}), our data indicate that this positively influences both the probability of interactions and the number of interactions, regardless of

[23] A statistical test should be performed to draw a conclusion.

the topic in discussion. However, given that two nodes have interacted, the direction of the effect of OD_{u1} fluctuates between topics, although in a weaker way than the one of EC_{u2}. On the other hand, the target's *outdegree* (OD_{u2}) negatively affects the probability of interaction between two nodes, regardless of the topic (except for `refugees`). However, among those pairs of users who interacted, the effect is overturned (positive and significant) for all topic-network structures.

Finally, we find that, regardless of the topic, both a highly clustered neighborhood of the source and target nodes (CL_{u1} and CL_{u2}) positively influence the probability of interaction. Nevertheless, given that two nodes have interacted, highly dense clusters negatively affect the number of interactions between nodes (this is especially evident for `refugees`).

Table 3: Regression models A (Complete OLS) per topic.

	Dependent variable:				
	no. of interactions	no. of interactions	no. of interactions	no. of interactions	no. of interactions
	(1)	(2)	(3)	(4)	(5)
	topic: `brexit`	topic: `populism`	topic: `refugees`	topic: `terrorism`	topic: `unemployment`
OD_{u1}	.0003***	.003***	.001***	.001***	.001***
	(0.00000)	(.0001)	(0.00000)	(.00002)	(.00002)
EC_{u1}	.0002	−.002	.018***	−.001*	−.001
	(.0002)	(.002)	(.001)	(.001)	(.001)
CL_{u1}	−.001***	−.001	−.002***	−.001*	−.001**
	(.0002)	(.002)	(.0003)	(.0004)	(.001)
OD_{u2}	−0.00000	−.001***	.0001***	−.0003***	−.0002***
	(0.00000)	(.0001)	(0.00000)	(.00002)	(.00002)
EC_{u2}	.058***	.100***	.151***	.103***	.155***
	(.0002)	(.002)	(.001)	(.001)	(.001)
CL_{u2}	−.002***	−.013***	−.017***	−.003***	−.005***
	(.0002)	(.002)	(.0003)	(.0004)	(.001)
constant	.001***	.006***	.003***	.002***	.002***
	(.00001)	(.0002)	(.00004)	(.00003)	(.00004)
Observations	23,449,806	494,912	8,711,352	4,131,056	1,975,430
R^2	.006	.010	.022	.006	.020
Adjusted R^2	.006	.010	.022	.006	.020
Residual Std. Error	.032 (df = 23449799)	.130 (df = 494905)	.129 (df = 8711345)	.068 (df = 4131049)	.062 (df = 1975423)
F Statistic	22,474.450*** (df = 6; 23449799)	820.195*** (df = 6; 494905)	32,597.910*** (df = 6; 8711345)	4,091.516*** (df = 6; 4131049)	6,872.124*** (df = 6; 1975423)
Note:					*p<0.1; **p<0.05; ***p<0.01

Table 4: Regression models B (Logistic) and C (Conditional OLS) per topic.

	Dependent variable:									
	interacted	no. of interactions	interacted	no. of interactions	interacted	no. of interactions	interacted	no. of interactions	interacted	no. of interactions
		(conditional)		*(conditional)*		*(conditional)*		*(conditional)*		*(conditional)*
	logistic	*OLS*	*logistic*	*OLS*	*logistic*	*OLS*	*logistic*	*OLS*	*logistic*	*OLS*
	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)
	topic: `brexit`		topic: `populism`		topic: `refugees`		topic: `terrorism`		topic: `unemployment`	
OD_{u1}	.079***	.004***	.302***	−.023	.040***	.012***	.304***	−.068***	.219***	.017***
	(.001)	(.001)	(.008)	(.016)	(.0003)	(.001)	(.005)	(.008)	(.004)	(.004)
EC_{u1}	−92.026***	−.001	−2.738***	.415	−4.875***	1.587***	−7.341***	.277	−7.961***	−.192
	(3.070)	(.500)	(.624)	(.377)	(.110)	(.174)	(1.014)	(.424)	(1.111)	(.294)
CL_{u1}	4.018***	−.205	.801***	.291	.609***	−.881***	.663***	−.340**	.693***	−.196
	(.231)	(.234)	(.258)	(.347)	(.058)	(.250)	(.157)	(.172)	(.215)	(.149)
OD_{u2}	−.434***	.015***	−.613***	.069**	−.001***	.010***	−.703***	.066***	−.629***	−.006
	(.011)	(.003)	(.027)	(.030)	(.0002)	(.0005)	(.013)	(.012)	(.015)	(.006)
EC_{u2}	6.101***	.136***	3.800***	−.278**	6.940***	1.411***	7.391***	−.638***	6.775***	−.204***
	(.037)	(.035)	(.085)	(.119)	(.032)	(.084)	(.072)	(.092)	(.062)	(.043)
CL_{u2}	−.069	−.693	−.541	−.470	.166***	−2.957***	−.347	.218	.485*	−.248
	(.545)	(.605)	(.363)	(.490)	(.046)	(.139)	(.237)	(.333)	(.273)	(.246)
constant	−9.227***	1.383***	−7.022***	2.465***	−7.306***	.885***	−8.113***	2.109***	−7.644***	1.398***
	(.030)	(.014)	(.056)	(.071)	(.013)	(.032)	(.033)	(.033)	(.037)	(.020)
Observations	23,449,806	9,146	494,912	1,346	8,711,352	13,846	4,131,056	4,452	1,975,430	3,296
R^2		.008		.009		.201		.028		.018
Adjusted R^2		.007		.004		.201		.027		.016
Pseudo R^2	.165		.181		.268		.175		.227	
Log Likelihood	−67,633.010		−7,636.007		−75,693.210		−28,825.970		−18,883.490	
Akaike Inf. Crit.	135,280.000		15,286.010		151,400.400		57,665.950		37,780.990	
Residual Std. Error		.784 (df = 9139)		1.080 (df = 1339)		2.330 (df = 13839)		.997 (df = 4445)		.611 (df = 3289)
F Statistic		12.133*** (df = 6; 9139)		1.917* (df = 6; 1339)		581.356*** (df = 6; 13839)		21.444*** (df = 6; 4445)		9.946*** (df = 6; 3289)

Note: *p<0.1; **p<0.05; ***p<0.01

6 Discussion

In this paper, we try to assess how users of OSN interact differently based on their topic of discussion. To answer this question, we examine how the node-level network characteristics of degree, centrality, and clustering affect the probability of interaction and the extent to which two nodes interact in different topic-network structures. We estimate regression models on a unique data set of 2,259,717 sampled tweets on five politics-related topics: `brexit`, `populism`, `refugees`, `terrorism`, and `unemployment`.

We find some evidence for systematic differences in the direction of the effect of centrality measures between distinct topics of discussion, but not for degree and clustering metrics. In particular, we find that the receiver user's eigencentrality positively predicts the interaction between users, regardless of the topic. Nevertheless, among the pairs of interacting users, the direction of this effect varies between topics. By contrast, the sender's eigencentrality negatively influences the probability of interaction, but not the extent to which users interact between them. We hypothesize that these variations might be uniquely attributed to the different intrinsic topic characteristics. However, the results of this study cannot be taken as evidence for supporting this conjecture because of the preliminary character of this investigation.

6.1 Managerial Implications

Although exploratory, this study provides some insights into how users behave and interact on OSN. Since some users in the network ignite multiple conversations, each revolving around a distinct topic, it seems essential to identify key nodes in the network. Furthermore, gaining knowledge of the specific network characteristics allows for the design of effective communication strategies that can minimize the latency in the information flow. Therefore, we propose a method for collecting Twitter data and analyzing topic-network structures that could be of interest not only to political marketers and, ultimately, to design the best strategy for efficient and well-targeted marketing campaigns on OSN.

6.2 Limitations of This Study and Further Research

We carry out a static analysis of tweet conversations. Studying the evolution of interaction behavior during the electoral period, perhaps in real-time, could be the object of future studies (see Mulder and Leenders, 2019).

The deficit of more and diverse network topology measures, as well as the limited number of topics taken into consideration, undermine the predictive power of this study. In particular, further research is needed to incorporate network modularity and take into account the resulting communities. Moreover, we capture topic information merely from the tracked hashtags without analyzing the content of conversation tweets. Future studies could overcome these limitations with parallel or distributed computing that allows for text analysis on such large data sets and without the need for a further sampling of data. Finally, future research could be extended to OSN other than Twitter.

Acknowledgements

This project is partly supported by research funds from Tilburg School of Economics and Management. I thank my supervisor, dr. Hannes Datta, for his guidance and having encouraged this project from the very beginning, instilling in me the passion for data analysis.

This thesis would not have been possible without the aid and support of my friends and valuable proofreaders Edoardo, Manuele, Marco, and Valentina. I owe them much.

Last but by no means least, I wish to thank my parents, my great emotional (and financial...) supporters who every day encourage me in all of my passions and inspire me to chase my dreams.

Bibliography

Aggarwal, C. C., editor (2011). *Social network data analytics*. Springer, New York, NY. OCLC: 729959038.

Aral, S., Brynjolfsson, E., and Van Alstyne, M. W. (2007). Productivity Effects of Information Diffusion in Networks. *SSRN Electronic Journal*.

Ardon, S., Bagchi, A., Mahanti, A., Ruhela, A., Seth, A., Tripathy, R. M., and Triukose, S. (2013). Spatio-temporal and events based analysis of topic popularity in twitter. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management - CIKM '13*, pages 219–228, San Francisco, California, USA. ACM Press.

Blumler, J. G. and Gurevitch, M. (2001). The New Media And Our Political Communication Discontents: Democratizing Cyberspace. *Information, Communication & Society*, 4(1):1–13.

Bode, L. and Dalrymple, K. E. (2016). Politics in 140 Characters or Less: Campaign Communication, Network Interaction, and Political Participation on Twitter. *Journal of Political Marketing*, 15(4):311–332.

Borgatti, S. P. (2005). Centrality and network flow. *Social Networks*, 27(1):55–71.

Buccoliero, L., Bellio, E., Crestini, G., and Arkoudas, A. (2020). Twitter and politics: Evidence from the US presidential elections 2016. *Journal of Marketing Communications*, 26(1):88–114.

Deer, L., Chintagunta, P. K., and Crawford, G. S. (2019). Online Word of Mouth and the Performance of New Products. pages 1–83.

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social Networks*, 1(3):215–239.

Grandjean, M. (2016). A social network analysis of Twitter: Mapping the digital humanities community. *Cogent Arts & Humanities*, 3(1).

Granovetter, M. S. (1973). The Strength of Weak Ties. *American Journal of Sociology*, 78(6):1360–1380.

Hemsley, J. (2019). Followers Retweet! The Influence of Middle-Level Gatekeepers on the Spread of Political Information on Twitter. *Policy & Internet*, 11(3):280–304.

Himelboim, I., Smith, M. A., Rainie, L., Shneiderman, B., and Espina, C. (2017). Classifying Twitter Topic-Networks Using Social Network Analysis. *Social Media + Society*, 3(1):1–13.

Hong, L. and Davison, B. D. (2010). Empirical study of topic modeling in Twitter. In *Proceedings of the First Workshop on Social Media Analytics - SOMA '10*, pages 80–88, Washington D.C., District of Columbia. ACM Press.

Hu, Y. (2005). Efficient and High Quality Force-Directed Graph Drawing. *The Mathematica Journal*, 10:37–71.

Java, A., Song, X., Finin, T., and Tseng, B. (2007). Why we twitter: understanding microblogging usage and communities. In *Proceedings of the 9th WebKDD and 1st SNA-KDD 2007 workshop on Web mining and social network analysis - WebKDD/SNA-KDD '07*, pages 56–65, San Jose, California. ACM Press.

Kruikemeier, S. (2014). How political candidates use Twitter and the impact on votes. *Computers in Human Behavior*, 34:131–139.

Kwak, H., Lee, C., Park, H., and Moon, S. (2010). What is Twitter, a Social Network or a News Media? In *Proceedings of the 19th international conference on World wide web - WWW '10*, pages 591–600, Raleigh, North Carolina, USA. ACM Press.

Lerman, K. and Ghosh, R. (2010). Information Contagion: an Empirical Study of the Spread of News on Digg and Twitter Social Networks. In *Proceedings of 4th International Conference on Weblogs and Social Media (ICWSM-10)*. arXiv: 1003.2664.

L'huillier, G., Alvarez, H., Ríos, S. A., and Aguilera, F. (2011). Topic-Based Social Network Analysis for Virtual Communities of Interests in the Dark Web. *SIGKDD Explorations*, 12(2):66–73.

Lim, K. W., Chen, C., and Buntine, W. (2016). Twitter-Network Topic Model: A Full Bayesian Treatment for Social Network and Text Modeling. *arXiv preprint arXiv:1609.06791*, pages 1–6. arXiv: 1609.06791.

Liu, L., Tang, J., Han, J., Jiang, M., and Yang, S. (2010). Mining topic-level influence in heterogeneous networks. In *Proceedings of the 19th ACM international conference on Information and knowledge management - CIKM '10*, pages 199–208, Toronto, ON, Canada. ACM Press.

Liu-Thompkins, Y. (2012). Seeding Viral Content: The Role of Message and Network Factors. *Journal of Advertising Research*, 52(4):465–478.

Milgram, S. (1967). The Small-World Problem. *Psychology Today*, 1(1):61–67.

Morstatter, F., Pfeffer, J., and Liu, H. (2014). When is it biased?: assessing the representativeness of twitter's streaming API. In *Proceedings of the 23rd International Conference on World Wide Web - WWW '14 Companion*, pages 555–556, Seoul, Korea. ACM Press.

Morstatter, F., Pfeffer, J., Liu, H., and Carley, K. M. (2013). Is the Sample Good Enough? Comparing Data from Twitter's Streaming API with Twitter's Firehose. In *Proceedings of the Seventh International AAAI Conference on Weblogs and Social Media*, pages 400–408.

Mulder, J. and Leenders, R. T. (2019). Modeling the evolution of interaction behavior in social networks: A dynamic relational event approach for real-time analysis. *Chaos, Solitons & Fractals*, 119:73–85.

Ortega, J. and Hergovich, P. (2017). The Strength of Absent Ties: Social Integration via Online Dating. *arXiv:1709.10478 [physics, q-fin]*. arXiv: 1709.10478.

Tang, J., Sun, J., Wang, C., and Yang, Z. (2009). Social influence analysis in large-scale networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '09*, pages 807–815, Paris, France. ACM Press.

Timm, I. J., Berndt, J. O., Lorig, F., Barth, C., and Bucher, H.-J. (2016). Dynamic Analysis of Communication Processes using Twitter Data. In *HUSO 2016: The Second International Conference on Human and Social Analytics*.

Toubia, O. and Stephen, A. T. (2013). Intrinsic vs. Image-Related Utility in Social Media: Why Do People Contribute Content to Twitter? *Marketing Science*, 32(3):368–392.

Tumasjan, A., Sprenger, T. O., Sandner, P. G., and Welpe, I. M. (2011). Election Forecasts with Twitter - How 140 Characters Reflect the Political Landscape. *SSRN Electronic Journal*.

Wang, Y., Callan, J., and Zheng, B. (2015). Should We Use the Sample? Analyzing Datasets Sampled from Twitter's Stream API. *ACM Transactions on the Web*, 9(3):1–23.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442.

Zhang, C., Sun, J., and Wang, K. (2013). Information propagation in microblog networks. In *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining - ASONAM '13*, pages 190–196, Niagara, Ontario, Canada. ACM Press.

Zhao, Z., Feng, S., Wang, Q., Huang, J. Z., Williams, G. J., and Fan, J. (2012). Topic oriented community detection through social objects and link analysis in social networks. *Knowledge-Based Systems*, 26:164–173.

A Appendix A

Table A.1: Data collection logbook.

Timestamp	Description
16/5/2019 14:50:00	Collection starts
16/5/2019 19:06:00	Collection starts
17/5/2019 0:39:54	Collection stops: update current keyword list
17/5/2019 0:46:33	Collection starts
17/5/2019 15:56:52	Collection starts
17/5/2019 16:10:52	Collection starts
18/5/2019 0:00:07	No. of tweets collected: 865,855 (5.24GB)
19/5/2019 0:40:00	Secondary collection stops: add tracked topics
19/5/2019 0:42:00	Secondary collection resumes
19/5/2019 16:50:26	No. of tweets collected: 2,062,320 (12.73GB)
19/5/2019 22:10:48	Add `brexit` as tracked topic
20/5/2019 1:13:40	Drop `euthanasia` from hot topics chart; add `EP2019`
21/5/2019 11:35:52	Primary collection stops: update KDNP's handle
21/5/2019 11:37:02	Primary collection resumes
22/5/2019 23:30:13	eu2019 collection stops
22/5/2019 23:39:41	eu2019v2 starts
27/5/2019 00:10:27	eu2019v2 collection stops
27/5/2019 0:21:49	Insert to eu2019v3 starts
02/6/2019 0:00:00	All collections terminate

Note: Timezone is CET.

Table A.2: Topic file for `refugees`.

Keywords		
refugees	избеглице	menekültek
flóttamenn	utečenci	beguncev
bēgļi	flyktingar	ffoaduriaid
pabėgėliams	immigrati	бежанци
flyktninger	uprchlíků	flygtninge
uchodźcy	vluchtelingen	põgenikele
refugiados	pakolaisten	immigrés
refugiați	Flüchtlinge	πρόσφυγες

Table A.3: Nodes table for topic: `refugees`.

Id	Label
1091795183560210000	AMIRDA1975
1103761100733110000	maryamnorouzi11
1092802809513290000	Parisamohajer1
1095035742177380000	Zgh75204326
707483831331397000	MarjanG1234
...	
1095096157401800000	ODREC_ONG
140792228	bentebecker
293418869	economicsfest
872025895657144000	LordProvostGCC
161983599	UNHCRGreece

Table A.4: Edges table for topic: `refugees`.

Source	Target	Weight
1092802809513290000	1091795183560210000	4
1092802809513290000	1087275641727250000	19
1095035742177380000	1103761100733110000	4
1103761100733110000	1103761100733110000	3
707483831331397000	1103761100733110000	2
...		
1046126245426140000	18393773	1
707483831331397000	3060125596	5
307012746	1116005242737500000	2
307012746	43115163	1
307012746	3060125596	1

Table A.5: Nodes table with network measures for topic: `refugees`.

Id	Label	Indegree	Outdegree	Degree	Weighted indegree	Weighted outdegree	Weighted degree	Eccentricity	Closness centrality	Harmonic closness centrality	Betweeness centrality	Modularity	Clustering	Eigencentrality
1091795183560210000	AMIRDA1975	23	43	66	106	92	198	6	.404408	.451794	69269621413	2	.181186	.348815
1103761100733110000	maryamnorouzi11	15	9	24	47	18	65	6	.381894	.414124	756181729	2	.45	.231786
1092802809513290000	Parisamohajer1	87	118	205	289	360	649	6	.464263	.533878	45579722672	2	.065114	.555289
1095035742177380000	Zgh75204326	54	181	235	242	766	1008	6	.496281	.585195	53321060569	2	.054794	.548923
707483831331397000	MarjanG1234	213	209	422	724	894	1618	6	.509015	.605472	213888327949	2	.027361	.750295
...														
1095096157401800000	ODREC_ONG	1	0	1	1	0	1	0	0	0	0	2	0	.000562
140792228	bentebecker	1	0	1	1	0	1	0	0	0	0	2	0	.000562
293418869	economicsfest	1	0	1	1	0	1	0	0	0	0	2	0	.001731
872025895657144000	LordProvostGCC	1	0	1	1	0	1	0	0	0	0	2	0	.000562
161983599	UNHCRGreece	1	0	1	1	0	1	0	0	0	0	2	0	.000562

Table A.6: Summary statistics for topic: `brexit`.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
indegree	4,843	1.905	7.165	0	0	1	211
outdegree	4,843	1.905	3.938	0	0	3	76
degree	4,843	3.810	7.806	1	1	4	211
weighted_indegree	4,843	2.711	10.798	0	0	2	274
weighted_outdegree	4,843	2.711	6.094	0	0	4	121
weighted_degree	4,843	5.422	12.012	1	1	5	274
eccentricity	4,843	.441	.642	0	0	1	4
closnesscentrality	4,843	.353	.466	0	0	1	1
harmonic_closness_centrality	4,843	.357	.469	0	0	1	1
betweenesscentrality	4,843	.751	16.555	0	0	0	790
clustering	4,843	.004	.031	0	0	0	0
eigencentrality	4,843	.009	.036	0	0	.01	1

Number of nodes = 4,843; Number of edges = 9,226.

Table A.7: Summary statistics for topic: `populism`.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
indegree	704	1.950	4.746	0	0	2	53
outdegree	704	1.950	1.987	0	0	3	16
degree	704	3.901	4.545	1	2	4	53
weighted_indegree	704	4.470	10.809	0	0	4	145
weighted_outdegree	704	4.470	5.318	0	0	8	72
weighted_degree	704	8.940	11.170	1	4	10	145
eccentricity	704	.751	.829	0	0	1	5
closness_centrality	704	.501	.467	0	0	1	1
harmonic_closness_centrality	704	.515	.474	0	0	1	1
betweeness_centrality	704	.555	3.573	0	0	0	42
clustering	704	.042	.101	0	0	0	0
eigencentrality	704	.032	.106	0	0	.01	1

Number of nodes = 704; Number of edges = 1,373.

Table A.8: Summary statistics for topic: `refugees`.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
indegree	2,952	4.725	16.975	0	0	2	430
outdegree	2,952	4.725	14.191	0	0	4	265
degree	2,952	9.451	24.953	1	3	5	430
weighted_indegree	2,952	9.395	49.085	0	0	3	1,614
weighted_outdegree	2,952	9.395	46.012	0	0	5	1,120
weighted_degree	2,952	18.791	81.807	1	4	7	1,864
eccentricity	2,952	2.932	3.238	0	0	7	10
closness_centrality	2,952	.358	.378	0.000	0.000	.643	1.000
harmonic_closness_centrality	2,952	.371	.382	0.000	0.000	.750	1.000
betweeness_centrality	2,952	665.544	6,589.509	0.000	0.000	0.000	213,888.300
clustering	2,952	.094	.177	0.000	0.000	.160	1.000
eigencentrality	2,952	.026	.081	0.000	0.000	.003	1.000

Number of nodes = 2,952; Number of edges = 13,949.

Table A.9: Summary statistics for topic: `terrorism`.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
indegree	2,033	2.224	6.831	0	0	2	93
outdegree	2,033	2.224	1.925	0	0	3	15
degree	2,033	4.448	6.588	1	3	4	97
weighted_indegree	2,033	4.029	12.562	0	0	4	216
weighted_outdegree	2,033	4.029	3.850	0	0	8	35
weighted_degree	2,033	8.058	12.334	1	4	8	216
eccentricity	2,033	1.038	1.040	0	0	2	6
closness_centrality	2,033	.541	.434	0	0	1	1
harmonic_closness_centrality	2,033	.566	.440	0	0	1	1
betweeness_centrality	2,033	1.973	20.934	0	0	0	470
clustering	2,033	.034	.090	0	0	0	0
eigencentrality	2,033	.013	.047	0	0	.01	1

Number of nodes = 2,033; Number of edges = 4,521.

Table A.10: Summary statistics for topic: `unemployment`.

Statistic	N	Mean	St. Dev.	Min	Pctl(25)	Pctl(75)	Max
indegree	1,406	2.370	10.014	0	0	1	166
outdegree	1,406	2.370	2.124	0	0	3	19
degree	1,406	4.740	9.865	1	3	3	169
weighted_indegree	1,406	3.322	14.036	0	0	2	257
weighted_outdegree	1,406	3.322	3.339	0	0	4	50
weighted_degree	1,406	6.644	13.884	1	4	5	257
eccentricity	1,406	1.165	1.075	0	0	2	4
closness_centrality	1,406	.571	.421	0	0	1	1
harmonic_closness_centrality	1,406	.599	.426	0	0	1	1
betweeness_centrality	1,406	2.142	24.824	0	0	0	692
clustering	1,406	.027	.079	0	0	0	0
eigencentrality	1,406	.013	.055	0	0	.004	1

Number of nodes = 1,406; Number of edges = 3,332.

Figure A.1: Bivariate correlation matrix for topic: `populism`.

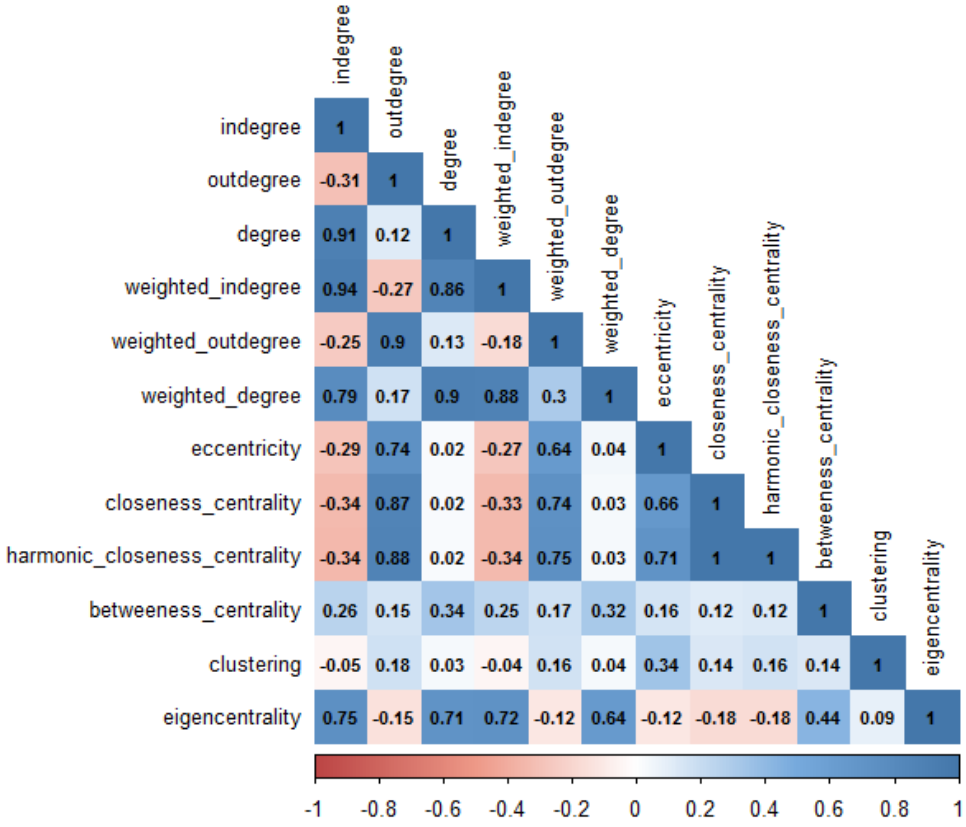


Figure A.2: Bivariate correlation matrix for topic: `refugees`.

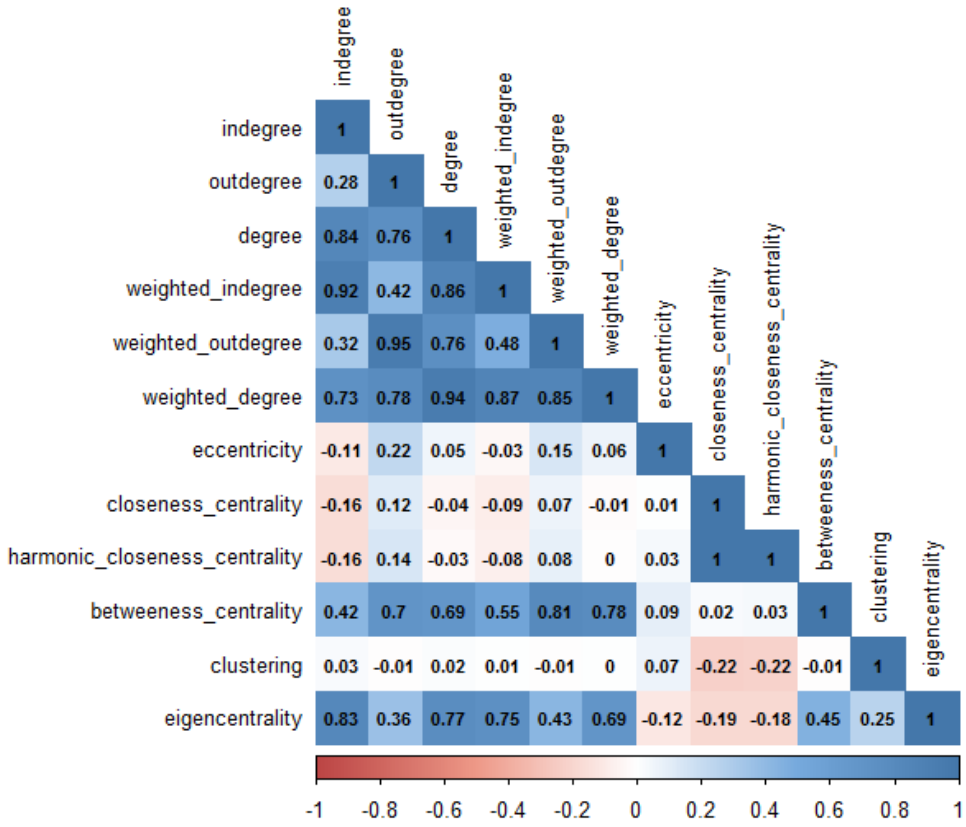


Figure A.3: Bivariate correlation matrix for topic: `terrorism`.

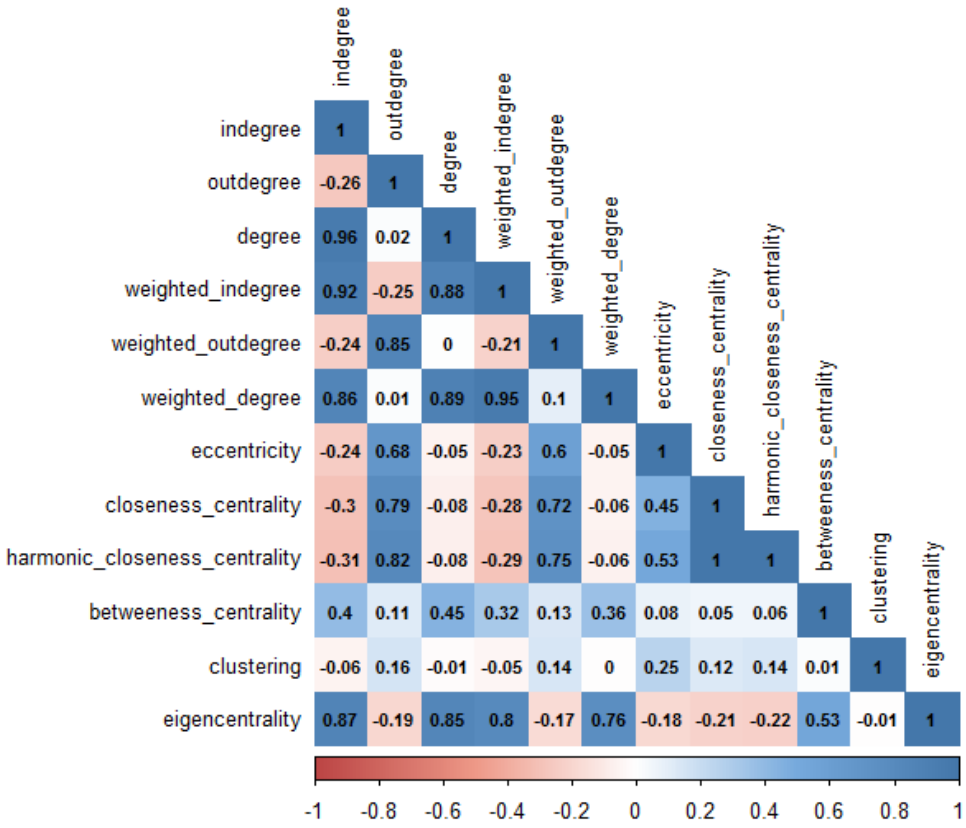


Figure A.4: Bivariate correlation matrix for topic: `unemployment`.

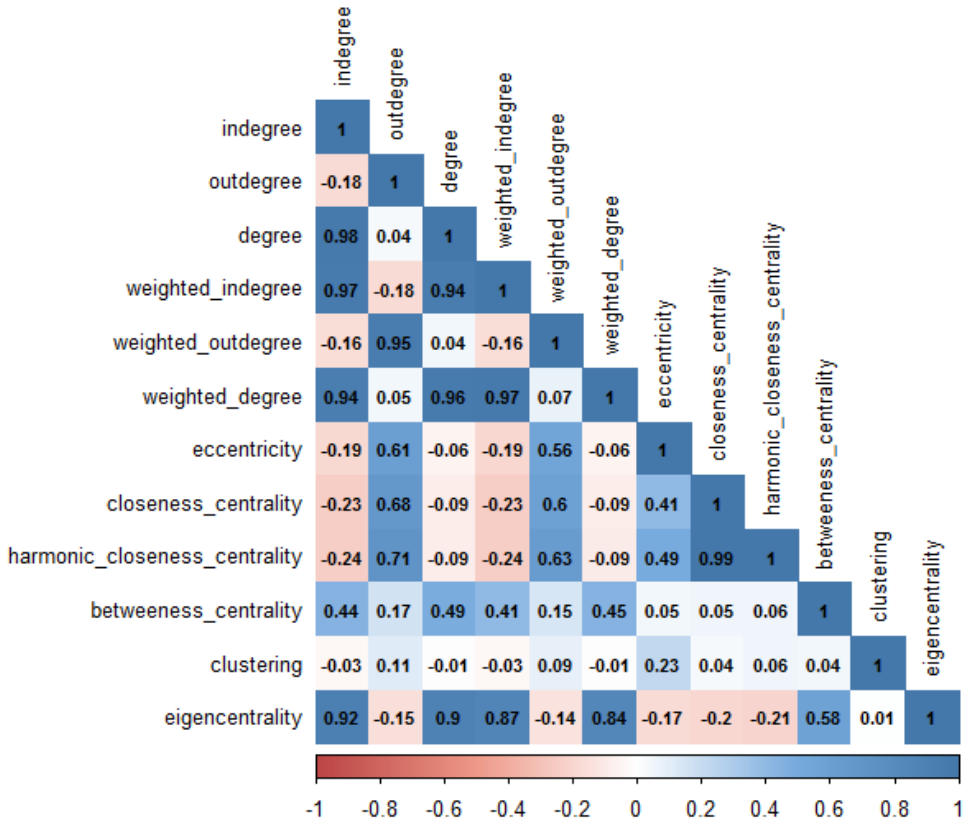


Figure A.5: Coefficients and SE per topic of Model A (Complete OLS).

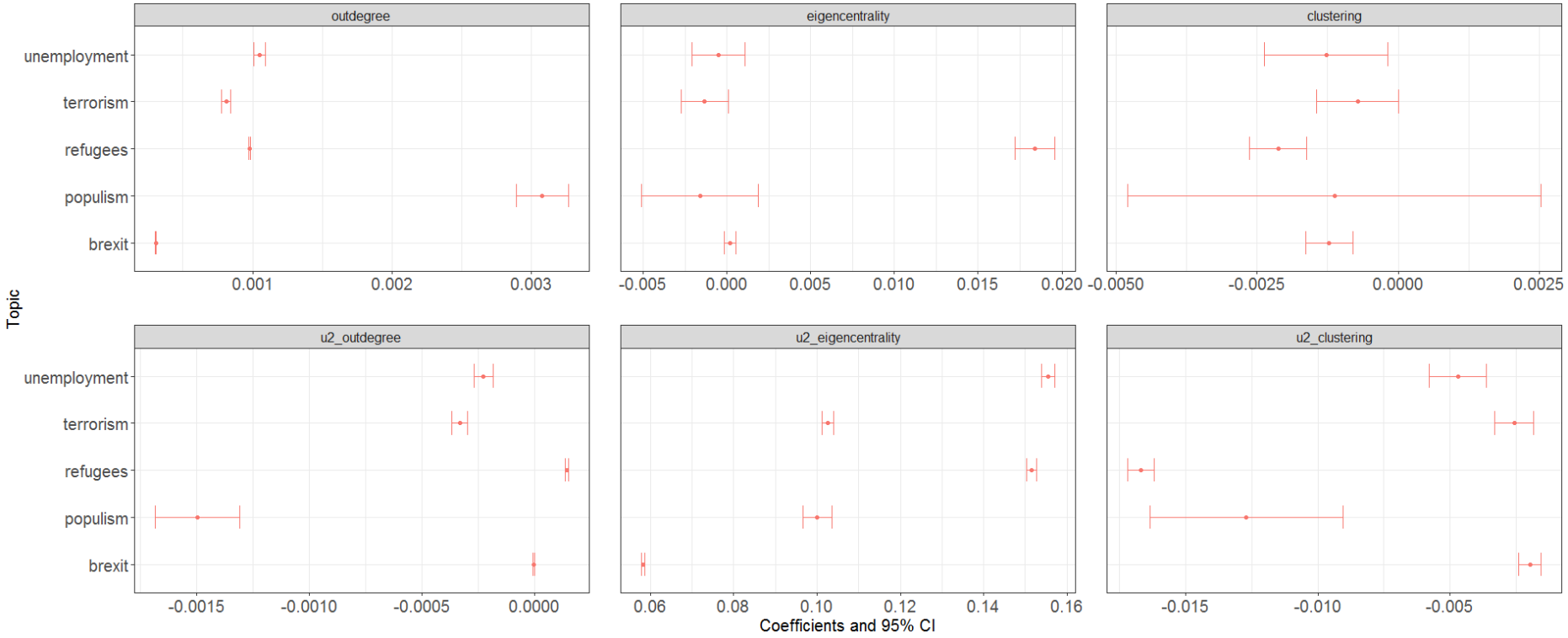
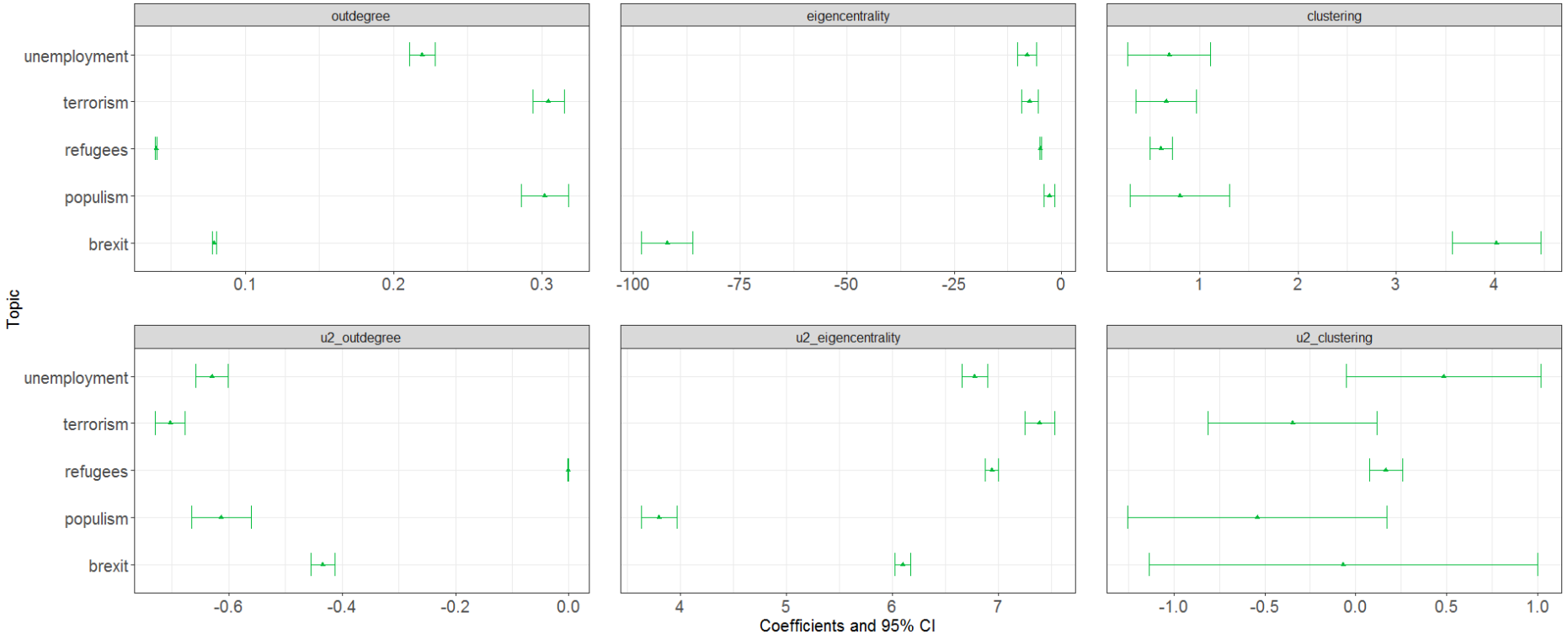


Figure A.6: Coefficients and SE per topic of Model B (Logistic).



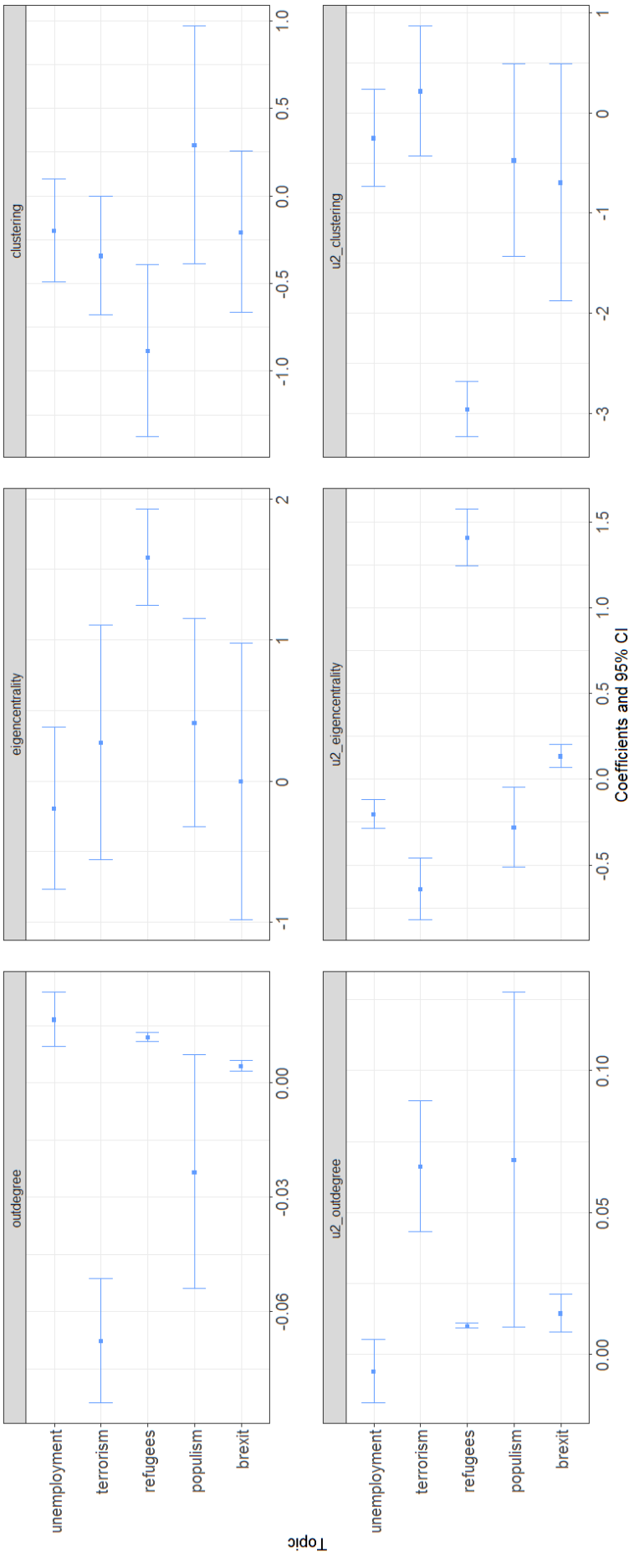


Figure A.7: Coefficients and SE per topic of Model C (Conditional OLS).

B Appendix B

@volkspartei
@SPOE_at
@neos_eu
#Jetzt
@Gruene_Austria
#ListeFirtz
@KPOE_EL
@vorwaertstirol
#LBL
@EUNEIN2
@GILTofficial
@PdA_Austria
@piratenparteiat
@de_nva
@cdenv
@openvld
@sp_a
@groen
@vlbelang
@partisocialiste
@mr_officiel
@lecdh
@ecolo
@defi_eu
@ppofficiel
@pvdabelgie
@CSPOstbelgien
@PPGERB
@BSP_Bulgaria
@PatriotiBG
@atakaBG
@HDZ001
@SDPHrvatske
@NLMost
@HSSHrvatska
@HaeNeS_
@StrankaGLAS
@zivizidhr
@IDSDDI
@BM365_stranka
@NHRvatsku
@HdsHrvatska
@demokrati_hr
@NS_reformisti
@HSLShr
@PromijenimoHR
@HDSSBhr
@DISY
@AKEL1926
@DIKO1976
@KiSos2020
@SymmaxiaPoliton
@allileggii2015
@cygreens
@ELAMcy
@matosmysl
@ODScz
@PiratskaStrana
@SPD_oficialni
@czKSCM
@CSSD
@kducsl
@TOP09cz
@STANcz
@svobodni
@zeleni_cz
@Spolitik
@DanskDf1995
@venstredk
@SFpolitik
@KonservativeDK
@Radikale
@Enhedslisten
@LiberalAlliance
@ep14dk
@alternativet_
@reformikad
@keskerakondlane
@aurorapropatria
@sotsdem
@Demarit
@persut
@kokoomus
@keskusta
@vihreat
@vasemmisto
@sfprkp
@KDpuolue
@enmarchefr
@lesRepublicains
@MoDem
@FranceInsoumise
@PCF
@RNational_off
@MvtRadical
@LesCentristes_
@FED_info
@LGM_67
@mdPMidiPyrenees
@DLF_Officiel
@AllianceC
@MRC_France
@CAP21_Tweets
@EELV38
@LePG
@GenerationsMvt
@Ensemble_FdG
@ecologistes_92
@agir_officiel
@ComitesJeanne
@ForceDu13
@GenCitoyens
@Liguedusud
@_LesPatriotes
@Gauche_RS
@Radicaux2Gauche
@Pe_A_Corsica
@PicardieDebout
@cdu
@spdde
@afd
@fdp
@dielinke
@die_gruenen
@csu
@BlaueWende
@LKR_Partei_Bund
@fwlandtag
@oedppresse
@diepartei
@piratenpartei
@npdde
@syriza_gr
@neademokratia
@kinimallagis
@ToPotami
@anexartitoi
@enosi_kentroon
@FideszEP
@JobbikMM
@kdnp
@mszptweet
@lehet_mas
@MLPLiberalisok
@FineGael
@fiannafailparty
@sinnfeinireland
@labour
@solidarityie
@indepenedents4_C
@greenparty_ie
@SocDems
@AontuIE
@HumanDignityIre
@RENUAIreland
@workersparty
@Vote4Homes
@RepublicanSF
@Mov5Stelle
@pdnetwork
@LegaSalvini
@forza_italia
@FratellidItaIia
@liberi_uguali
@Piu_Europa
@CasaPoundItalia
@SaskanaOnline
@KamPiederValsts
@konservativie
@AttistibaiPar
@VL_TBLNNK
@ZZS_
@Jauna_Vienotiba
#LVZS
@tslkd
@lsdp_vilnius
@darbopartija
@adr_lu
@CSV_news
@dp_lu
@deigreng
@dei_lenk
@lsap_lu
@Piratepartei
@PL_Malta
@PNmalta

@Demokratiku
@vvd
@cdavandaag
@d66
@groenlinks
@spnl
@pvda
@christenunie
@partijvddieren
@50pluspartij
@sgpnieuws
@denknl
@fvdemocratie
@pisorgpl
@Platforma_org
@KUKIZ15
@nowePSL
@Nowoczesna
@Porozumienie__
@SolidarnaPL
@WolniSolidarni
@wybierz_TERAZ
@Unia_Euro_Dem
@UPR_org
@PoZdroj
@republikanieorg
@Prawica_Rz
@RuchNarodowy
@Bialo__Czerwoni
@sldpoland
@NowaPrawica
@OsVerdes
@Blocomadeira
@psocialista
@Partido_PAN
@LIVREpt
@pdr_coimbra
@MovimentoJPP
@LiberalPT
@ppdpsd
@_CDSPP
@Partido_Alianca
@psdct
@PnlRomania
@usr_romania
@RMDSZ_UDMR
@alde_romania
@smersd
@stranasas
@strankaSDS
@strankaSD
@StrankaSMC
@strankalevica
@NovaSlovenija
@PS_DeSUS
@StrankaSAB
@SnsStranka
@populares
@PSOE
@ABUnidasPodemos
@CiudadanosCs
@Esquerra_ERC
@Pdemocratacat
@eajpnv
@PartidoPACMA
@ehbildu
@coalicion
@socialdemokrat
@moderaterna
@sdriks
@miljopartiet
@Centerpartiet
@vansterpartiet
@liberalerna
@kdriks
@Feministerna
@Conservatives
@UKLabour
@theSNP
@LibDems
@ForChange_Now
@duponline
@Plaid_Cymru
@TheGreenParty
@scotgp
@brexitparty_uk
@UKIP
@SDLPlive
@uuponline
@allianceparty
@GreenPartyNI
@pb4p
@TheSDPUK
@sebastiankurz
@rendiwagner
@HCStracheFP
@BMeinl
@Maria__Stern
@WKogler
#Dinkhauser
#Kölly
@GabrielHribar
@MarschallRobert
#RolandDüringer
@bart_dewever
@wbeke
@ruttengwendolyn
@johncrombez
@meyremalmaci
@tomvangrieken
@eliodirupo
@ochastel
@prevotmaxime
@zakiakhattabi
@oliviermaingain
@modrikamen
@peter_mertens
@OliverPaasch
@KattrinJadin
@BoykoBorissov
#KorneliyaNinova
@KKarakachanov
@valeri_simeonov
@volensiderov
@MareshkiVeselin
@AndrejPlenkovic
@davor_bernardic
@BozoPetrov
@KBeljak
@anka_mrak
@Ivan_Pernar
@_BorisMiletic
@BrunaEsih
@SilvanoHSU
@drMrsic
@DarinkoKosor
@ladislav_ilcic
@AverofCY
@GenSecAKEL
@NicholasPapadop
@sizopoulos
@yiorgoslillikas
@THEOCHAROUSE
@gperdikes
@AndrejBabis
@P_Fiala
@PiratIvanBartos
@tomio_cz
@vojtafilip
@jhamacek
@MarekVyborny
@Pospisil_Jiri
@JanFar_sky
@JTP07
@stepanekpraha
@Kristianthdahl
@larsloekke
@PiaOlsen
@SorenPape
@oestergaard
@PSkipperEL
@anderssamuelsen
@uffeelbaek
@kajakallas
@ratasjuri
@Mart_Helme
@HelirSeeder
@JevgeniO
@AnttiRinnepj
@Halla_aho
@PetteriOrpo
@juhasipila
@Haavisto
@liandersson
@anna_maja
@SariEssayah
@StanGuerini
@laurentwauquiez
@faureolivier
@bayrou
@JLMelenchon
@Fabien_Rssl
@MLP_officiel
@LaurentHenart
@Herve_Morin
@HerveMarseille
@BockelJeanMarie
@RobertHueOff
@dupontaignan
@philippefolliot
@jluc_laurent
@corinnelepage
@DavidCormand
@ericcoquerel
@benoithamon
@FdeRugy
@jeanlassalle
@franckriester

@lepenjm
@jnguerini
@JeanMarieCAVADA
@JacquesBompard
@f_philippot
@mnlienemann
@VRoziere
@Gilles_Simeoni
@Francois_Ruffin
@akk
@AndreaNahlesSPD
@Joerg_Meuthen
@c_lindner
@katjakipping
@ABaerbock
@Markus_Soeder
@FraukePetry
@Bernd_Koelmel
@HubertAiwanger
@ChristophRaabs
@MartinSonneborn
@sebulino
@FrankFranz
@atsipras
@kmitsotakis
@IliasKasidiaris
@FofiGennimata
@St_Theodorakis
@PanosKammenos
@semjen
@GyurcsanyMES
@marta_demeter
@Bajnai_Gordon
@LeoVaradkar
@MichealMartinTD
@MaryLouMcDonald
@BrendanHowlin
@EamonRyan
@CathMurphyTD
@Toibin1
@SeamusHealyTD
@RonanMullen
@johnleahyRENUA
@GraTire
@luigidimaio
@nzingaretti
@matteosalvinimi
@berlusconi
@GiorgiaMeloni
@PietroGrasso
@emmabonino
@distefanoTW
@nilsusakovs
@artuss
@Bordans
@pavluts
@RaivisDzintars
@ArmandsKrauze
@aseradens
@RamunasLVZS
@UspaskichV
@Linas_Balsys
@puteikis
@CorinneCahen
@KmiotekC
@FranzFayot
StarskyFlor
@JosephMuscat_JM
@adriandeliapn
@GodfreyFarrugia
@markrutte
@geertwilderspvv
@sybrandbuma
@AMSpierings
@jesseklaver
@MarijnissenL
@LodewijkA
@gertjansegers
@mariannethieme
@HenkKrol
@keesvdstaaij
@tunahankuzu
@thierrybaudet
@KaczynskiJaro
@SchetynadlaPO
@pkukiz
@KosiniakKamysz
@KLubnauer
@Jaroslaw_Gowin
@ZiobroPL
@KornelMorawiec1
@RyszardPetru
@EBinczycka
@BJozwiakUPR
@JkmMikke
@AnnaSiarkowska
@PrawicaKawecki
@wlodekczarzasty
@catarina_mart
@MadalenoPTP
@ruitavares
@RuiRioPSD
@CristasAssuncao
@PSantanaLopes
@_LiviuDragnea
@Ludovic_Orban
@kelemenhunor
@eugen_tomac
@MonikaFlaBenova
@SulikRichard
@igor_matovic
@MarianKotleba
@JJansaSDS
@sarecmarjan
@ZidanDejan
@MiroCerar
@LukaMesec
@MatejTonin
@ErjavecKarl
@ABratusek
@ZmagoPlemeniti
@pablocasado_
@CristinaNarbona
@Pablo_Iglesias_
@Albert_Rivera
@junqueras
@davidbonvehi
@andoniortuzar
@ArnaldoOtegi
@anioramas
@SwedishPM
@ulfgooglar
@jimmieakesson
@bolund
@annieloof
@jsjostedt
@bjorklundjan
@BuschEbba
@gudschy
@theresa_may
@jeremycorbyn
@NicolaSturgeon
@joswinson
@heidiallen75
@DUPleader
@Adamprice
@jon_bartley
@patrickharvie
@Nigel_Farage
@GerardBattenMEP
@columeastwood
@RobinSwannUUP
@naomi_long
@ClareBaileyGPNI
@JimAllister
@WilliamClouston
#eurovaalit2019
#EUvaalit2019
#ElezioniEuropee
#ElezioniEuropee2019
#EP2019
#EU2019
#EuropeanElections
#EuropeanElection2019
#Europawahl2019
#Europeias2019
#Europa2019
#Europawahl
#EUverkiezingen
#EuropeseVerkiezingen
#EUelections2019
#EP19
#ALDE
#EPP
#S&D
#ECR
#GUENGL
#Greens
#EFA
#EFDD
#ENF
@ALDEgroup
@EPP
@TheProgressives
@ecrgroup
@GUENGL
@GreensEP
@EFDgroup
@ENF_EP
@EFAparty
@EPPGroup
brexit
refugees
flóttamenn
bēgļi
pabėgėliams
flyktninger
uchodźcy

refugiados
refugiați
избеглице
utečenci
beguncev
flyktingar
ffoaduriaid
immigrati
бежанци
uprchlíků
flygtninge
vluchtelingen
põgenikele
pakolaisten
immigrés
Flüchtlinge
πρόσφυγες
menekültek
terrorism
hryðjuverk
terorismu
terorizmas
terrorisme
terroryzm
terrorismo
terorism
тероризам
terorizmus
terorizma
terfysgaeth
тероризъм
terorismus
terrorismiga
terrorismi
Terrorismus
τρομοκρατία
terrorizmus
public debt
skuldir hins opinbera
valsts parāds
valstybės skola
offentlig gjeld
dług publiczny
dívida pública
datorie publica
Јавни дуг
verejný dlh
javni dolg
deuda publica
statsskuld
dyled gyhoeddus
debito pubblico
публичен дълг
veřejný dluh
offentlig gæld
Publieke schuld
riigivõlg
julkinen velka
dette publique
Staatsverschuldung
δημόσιο χρέος
államadósság
single market
innri markaðurinn
vienoto tirgu
bendroji rinka
indre marked
jednolity rynek
mercado único
piață unică
единствено тржиште
jednotného trhu
enotnega trga
Enskild marknad
marchnad sengl
mercato comune europeo
Единен пазар
Jednotný trh
Interne markt
Ühtne turg
Sisämarkkinat
marché unique
Binnenmarkt
ενιαία αγορά
egységes piacon
gender equality
jafnrétti kynjanna
dzimumu līdztiesība
lyčių lygybė
Likestilling
równość płci
igualdade de gênero
egalitatea sexelor
родна равноправност
rovnosť pohlaví
enakost med spoloma
igualdad de género
jämställdhet mellan könen
Cydraddoldeb Rhyw
uguaglianza di genere
равенството между половете
rovnosti žen a mužů
ligestilling
geslachtsgelijkheid
sooline võrdõiguslikkus
sukupuolten tasa-arvo
égalité des sexes
Geschlechtergleichheit
ισότητα των φύλων
nemek közötti egyenlőség
lgbt rights
lgbt réttindi
lgbt tiesības
Lgbt teisės
lgbt rettigheter
prawa LGBT
direitos lgbt
Lgbt drepturile
лгбт ригхтс
lgbt práva
lgbt pravice
derechos lgbt
lgbt rättigheter
hawliau LHDT
diritti lgbt
lgbt права
práva lgbt
lgbt rettigheder
lgbt rechten
LGBT õiguste
lgbt-oikeudet
lgbt droits
lgbt Rechte
λγβτ δικαιώματα
lgbt jogok
unemployment
atvinnuleysi
bezdarbs
nedarbas
arbeidsledighet
bezrobocie
desemprego
şomaj
незапослености
nezamestnanosť
brezposelnosti
desempleo
arbetslöshet
diweithdra
disoccupazione
безработица
nezaměstnanost
arbejdsløshed
werkloosheid
tööpuudus
työttömyys
chômage
Arbeitslosigkeit
ανεργία
munkanélküliség
populism
populisms
populizmas
populisme
populizm
populismo
популизам
populizmus
populizem
популизъм
populismus
populismi
λαϊκισμού
fake news
eu institutions
eu stofnanir
Eiropas Savienības iestādes
Europos Sajungos institucijos
europeiske union institusjoner
instytucje unii europejskiej
instituições da união europeia
instituțiile Uniunii Europene
институције европске уније
inštitúcie EÚ
institucij eu
intituciones de la ue
eu institutioner
sefydliadau r UE
istituzioni europee
институции Европейския съюз
instituce Evropské unie
instellingen van de Europese Unie
Euroopa Liidu institutsioonid
Euroopan unionin toimielimet
institutions européennes
eu institutionen
EU-intézmények
θεσμικά όργανα της ΕΕ