A Lexicon-based Approach to Users' Clustering: An Application to Misinformation on Youtube

Ca' Foscari University of Venice
Department of Environmental Sciences, Informatics and Statistics

Master Thesis in Computer Science
Academic Year 2019-2020

Student	Santoro Arnaldo
Mat.	822274
Supervisor	Zollo Fabiana
Co-Supervisor	Cinelli Matteo

Acknowledgments

Grazie a tutti coloro che mi hanno supportato.

Abstract

The internet and social media allowed individuals to overcome barriers of time and space, enabling the connection of people and events. However their characteristics may contribute to the massive spread of digital misinformation, which is one of the major threats of our society.

In the wake of the COVID-19 global pandemic, disintermediation in the news cycle jointly with the spreading misinformation poses great dangers to society, and there's an urge to studying the phenomenon and providing appropriate countermeasures. Users of social media tend to group themselves into echo-chambers, communities made up of polarized individuals adhering to the group's common narrative and the "echo-chamber effect" seems to be even more prominent in social media that implement a feed algorithm. Furthermore, when presented with dissenting information these users react by consuming more (mis)information.

In this work we analyze characteristics YouTube comments of COVID-19 - related content; then we explore a lexicon-based approach to user clustering using their comments on YouTube. This method allows to identify possible sources of misinformation without information regarding users' preferences.

The insights of this work may prove useful in monitoring sources of misinformation on the platform, aid research on the topic, and lay a path for potential future interventions.

Keywords Misinformation, YouTube, Echo Chamber, Clustering, Lexical Clustering, User Clustering

Contents

Chapter 1

Introduction

Every day social media users are exposed to a huge amount of information coming from the platforms, which are designed for entertainment content. The news that social media users are exposed to comes from all kinds of sources and might be unreliable.

Using user preferences, some platforms implement feed algorithms to offer the users content they are more likely to engage with; this mechanism has the effect of creating or strengthen echo-chambers [35] [12]. Echo-chambers are clusters of like-minded users which share a common narrative. Because of this online misinformation finds an optimal environment to spread in these platforms [18]: users in these echo-chambers are presented with (mis)information that confirms their preexisting beliefs, polarizing opposing point of views [37]. Simultaneously we have seen the rise of both anti-vaccination movements and diseases which were thought to be defeated.

Misleading or even false information can be used by malevolent users to their political, economical, or ideological advantage [19]. It has been listed as one of the main threats to society alongside religious fanaticism and terrorism [22].

To implement any policy it is important to detect users which are subject to fake news: for example the practice of debunking was proven to cause a backfire effect on misinformation consumers, reinforcing their preexisting beliefs [43]. However social media platforms often retain the most informative data like user preferences, i.e. likes , making it difficult to identify the leaning of the users; this hinders any study on the subject, and forces researchers to seek other ways to investigate. The results in [17] and [7] indicate that one such way may be to employ the lexical properties of posts, comments, and tweets to identify users' affiliation with an echo-chamber.

This work's purpose is to employ an approach based on lexical properties of comments made by users on the YouTube platform on the topic of COVID-19 and to assess its efficacy, with the purpose of providing a tool to help address the barrier posed by missing proprietary data.

This Thesis is structured as follows:

Chapter 2 explains the problem of misinformation and the approaches in literature;

Chapter 3 explores the data using NLP techniques; in Chapter 4 we explain in detail data preparation, performs lexical clusterings using various techniques, evaluate them using statistics based on comments, inspect their content using word-clouds, and compare them to network clusterings using Rand Index; conclusions and future works are discussed in Chapter 5. Methods and techniques are presented in the Appendix.

Chapter 2

Misinformation

2.1 Introduction

When referring to misinformation, often the terms Misinformation, Disinformation, Propaganda, Rumors, Conspiracy Theories, Fake News are used interchangeably. To avoid additional unnecessary confusion we will refer to the definitions presented in [19]:

- Misinformation: “A claim that contradicts or distorts common understandings of verifiable facts”.
- Disinformation: “The subset of misinformation that is deliberately propagated”. It differs from the former mainly by its intent to deceive.
- Propaganda: “Information that can be true but is used to ’disparage opposing viewpoints’”.
- Rumors: “Claims whose power arises from social transmission itself” whose truth value is not put into discussion inasmuch as the definition is concerned.
- Conspiracy Theories: these claims “have specific characteristics, such as the belief that a hidden group of powerful individuals exerts control over some aspect of society”.
- Fake News: content which takes the form of “deliberately misleading articles designed to mimic the look of actual articles from established news organizations”.

Misinformation is not a new concept, and as the issue gained popularity in recent years many articles and books from which to draw examples were published.

Some notable examples are:

- Otto von Bismark's manipulation of the news to find a pretext for the Franco-Prussian war is a well known example of news manipulation [23].

- The "cannards", french variety of seventeenth century fake news, which were used to disparage Marie Antoniette before the french revolution [8].

- The conspiracy theory about the existence in Italy of a "stay-behind" organization named Operazion Gladio. Its existence was confirmed by notable Italian politicians once it dissolved [1] and exemplifies the lack of truth value in the very definition of "Conspiracy Theory", due to the verifiability of the news.

- The fake news about the death of Napoleon in 1814 to manipulate London Stock Exchange [1].

2.1.1 Misinformation and the Internet

It is thus clear that misinformation is not a new concept, and that the problem is very difficult to address because of its very nature of misinformation: when we are offered new information that we are not able to directly verify we need to either know the truth value of said information beforehand or trust the information source.

This however is a problem that spans all fields of human knowledge. It takes time, resources, and field knowledge to be solved properly, and it has to be dealt with each time anyone is offered new information.

Before the rise of the internet, information was written and verified by journalists, be it more or less skilled, be it more or less partisan, and was spread through traditional mass media. Things changed twenty years ago, when communication technology was spreading and new ideas took form and revolutionized the way society handled information. Instant messaging application made communication between people effortless, and social networks were able to connect people with same experiences or interests.

These technologies inspired great optimism about their potential, enabling collective actions like connecting donors for medical transplants, and facilitated new forms of expression [19].

The role of social media in the words of Facebook's founder and CEO is trying to do is to "just make it really efficient for people to communicate, get information and share information" [25] As a result news were subject to disintermediation, leaving to the general public the difficult task to verify a huge amount of information of all kind of fields. In this context news on geopolitics and science are subject to the same scrutiny of a cat picture.

[1] this voice is present among Wikipedia's examples for fake news, but the page albeit presenting general references lacks inline citations, thereby making it a potential Fake News about Fake News

2.2 Anatomy of Misinformation

One of the characteristics of the phenomenon lies in the way human cognition operates in the media environment. In the internet a huge amount of information competes for user's attention; attention in turn is limited, so the human mind uses heuristics and shortcuts to choose which information to examine. Namely, this cognitive process is subject to confirmation bias, i.e. a claim is easily more accepted if it adheres to a user's belief system. Moreover, misinformed subjects are sometimes more confident in their beliefs than the correctly informed [19]. This leads users to fragment themselves into echo chambers. Echo chambers can be defined as communities of like-minded users sharing a common narrative, which rarely interact outside their group. It has been shown that selective exposure to single Facebook pages increases with users' activity, while selective exposure to topics decreases [12], thus further limiting exposure to other sources of information. This system of interactions lead to highly polarized communities, which interact outside their echo chamber mainly to attack users adhering opposing narratives [11].

Information spreads at high speed within these groups. The difference in behaviour of misinformation against reliable information is object of debate: some [38] argue that in specific settings misinformation spreads faster than information, however this phenomenon does not always occur and could be platform-dependent [13].

By examining the personalities users in conflicting communities, it was shown that permanence inside the echo chamber seems to skew individual personalities toward that prevalent type, and that personality landscape seems mostly invariant from the communities' nature [2]. The implication is that prolonged use of social media shapes the personality of users, independently from what they use it for.

Summarizing, information selection is also influenced by social norms, therefore users do not operate as rational agents but choose information source based on who is sharing it and whether or not it adheres to its preexisting beliefs. Information and misinformation may spread alike or differently, depending on the platform, and users consuming misinformation do not show particularly different behaviour with respect to users consuming other kinds of information.

The effective role of the algorithmic bias induced by social media platforms in shaping echo chambers and their effect on selective exposure is still object of discussion and research; recently it has been proven that there's a link between sites that implement an automatic feed algorithm like Facebook and Twitter and the polarization effect [14]; this effect is instead absent in platform offering a tweakable feed algorithm.

The implication is that the business model followed by these companies directly hurts its users by exploiting human bias in information selection and bonding relations, and it generates an environment which can anesthetize society, brings social unrest, and may lead individuals to do physical harm to themselves and to others.

Furthermore, and importantly for this work's purposes, the companies that host the platforms collect information on its users for profiling purposes, and often choose not to share it, thereby hindering research and prevention.

2.3 Impact of misinformation

The impact of misinformation may differ across the world, depending on the type of government. In countries like China and Russia the government actively engages in misinformation campaigns towards its population more than towards other countries, whereas western countries direct their efforts mainly towards other countries [19]. The effects of misinformation may be particularly strong in democracies, where disintermediation of information, algorithms, polarization and echo chambers may harm dialogue between members of different groups. All these phenomena have been observed during online debates in the 2016 campaign for USA presidential elections, where the majority of fake news were pro-Trump, and during the Brexit referendum during the same year [10]. In a similar fashion during 2020 USA elections, one party's rhetoric based on unfounded rumors lead to a public manifestation, which in turn lead to the death of three persons [24] . These phenomena however do not affect only political topics. However the impact of misinformation is not limited to politics: it has been shown that the echo-chambers effect applies to communities around health topics, as it happens to all critical topics [36] [17]. In recent years diseases which were thought to be eliminated reappeared [42], [5], and vaccine hesitancy is one of the causes [30].

In the past years in some countries, among which we find Italy, the reactions to vaccine-hesitant movements have been answered with mandatory vaccination policies. These drastic decisions have demonstrably dangerous effects on polarized communities [3]: experimental evidence shows that mandatory vaccination policies increase resentment in dissenting communities, and have demonstrably detrimental effects on vaccination uptake. Moreover presenting argument against misinformation has been shown to provoke a back-fire effect on polarized individuals, reinforcing their pre-existing beliefs [6].

Studies in the communities against vaccination lead to the conclusion that anti-vaccination attitudes are deeply rooted into a person's identity [36], making any action regarding these individuals extremely delicate.

More recently, the COVID-19 outbreak found a very delicate media environment: the term "infodemic" has been coined to emphasise the results of an overabundance of information that worsens the effects of the disintermediation of news, in which misinformation thrives [29], and since vaccine hesitancy and online misinformation have been linked [41], studying users engaging in misinformation has become a critical field of study.

Chapter 3

Misinformation on Covid-19 on YouTube

YouTube is a social media platform designed for entertainment that hosts misinformation sources alongside reliable sources. It is known that individuals engaging in misinformation on other platforms mostly share URLs to YouTube videos [17]. It is known that YouTube receives a large volume of interactions on content related to COVID-19 [13]. Notwithstanding its importance it remains largely understudied [19]. The platform implements a feed algorithm, which suggests the presence of echo chambers [14].

In this chapter we analyze the nature of the data at our disposal, the limits we face, and the means we have to evaluate future results. We will explain how we identify misinformation sources on YouTube and YouTube users' interaction with them in detail.

3.1 Data

3.1.1 YouTube

Using YouTube API we downloaded data on YouTube videos related to COVID-19, including information on channels comments and commenters.

Statistics on YouTube data are summarized in Table 3.1

The detected language of the comments is 75.9% Italian, 8.9% NA, and 15.2%

Comments	470244
Videos	30436
Channels	7157
“likelySpam” Comments	6345
Detected Languages	108

Table 3.1: YouTube data statistics

Site	# of entries
Butac	559
Bufale	161
Newsguard	733

Table 3.2: Statistics on the misinformation blacklists.

other languages. A brief manual inspection revealed that some of the other languages and NA contained Italian or partially Italian comments, but others were indeed in English, Polish, German, and other languages. As this work interests only Italian language and fast thorough manual inspection is impossible, comments which were not recognized as Italian were excluded. After manual examination the 6345 "likelySpam" labeled comments were also excluded, as they mostly contained single worded comments or actual spam. This left approximately 75.8% of the original comments.

3.1.2 Blacklists

We downloaded three lists of misinformation spreading sites and channels to label YouTube misinformation sources. The datasets were downloaded from the free blacklist pages available in [9], [39], and [40].
The blacklists are updated regularly, and contain independently checked websites, Facebook pages, YouTube channels, and Blogs classified as misinformation originating from many countries.

Since the aim of these three lists is to identify misinformation spreaders we do not possess information on other types of channels, and we are both unable to place the channels on a spectrum, and to divide by categories such as "entertainment channel" and "scientific news channel".

Table 3.2 summarizes the content of the blacklists; Butac divides misinformation sources about many topics summarizable as:Q-Anon sites, Pseudo science, Pseudo journalism and satire, Meteoterrorism, Viral News, Conspiracy Theory, Clickbait, Facebook, YouTube; note that Newsguard contains information on non-Italian misinformation sources; some names in the three lists do overlap, and in some cases overlapping occurs even in the same list, as the same misinformation source is present in more than one media.

3.2 Identification of Misinformation Channels

The blacklists from fact-checking sites were matched against all channels in the Channels dataset to find the best match for each channel.

The technique of exact string matching does not suffice to spot misinformation channels from a list, given the common practice of using slight modifications in the channel's title; this happens for stylistic reasons, to avoid recognition, or because

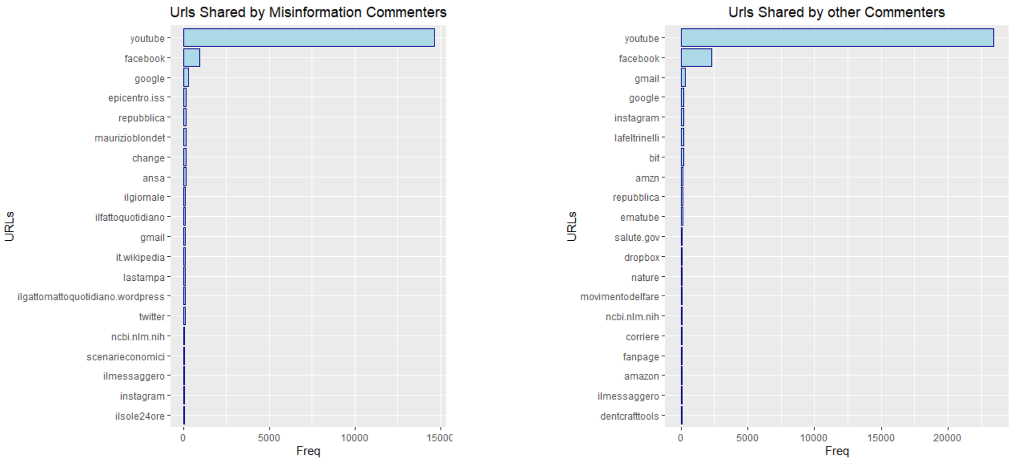


Figure 3.1: Domains shared by commenters in the whole data, divided by *leaning*.

there is more than one channel by the same misinformation producer, either in the same platform or in different ones.

In this last case the difference is usually a suffix like a numeral, or a word denoting the nature of the media, while for stylistic reasons the name may be altered using non-unicode characters. Whichever the case, a missing white-space for an exact matching algorithm is sufficient to miss a match.

Some mismatch causes may be dampened by applying a set of standard string transformations, such as lowercase transformations, white-space removal, and others; their appropriateness depends on the case.

The channels' names were matched with each element in the blacklists using exact matching; the transformations applied to both the blacklist and the channels' names were:

- Remove non-unicode characters
- Symbol removal
- White-space removal
- Lowercase transformation

- Remove non-unicode characters
- Symbol removal
- White-space removal
- Lowercase transformation

Then using a set of 8 string distance measures (see Appendix) the fuzzy matching was applied between the channels' names which did not match and the terms in the

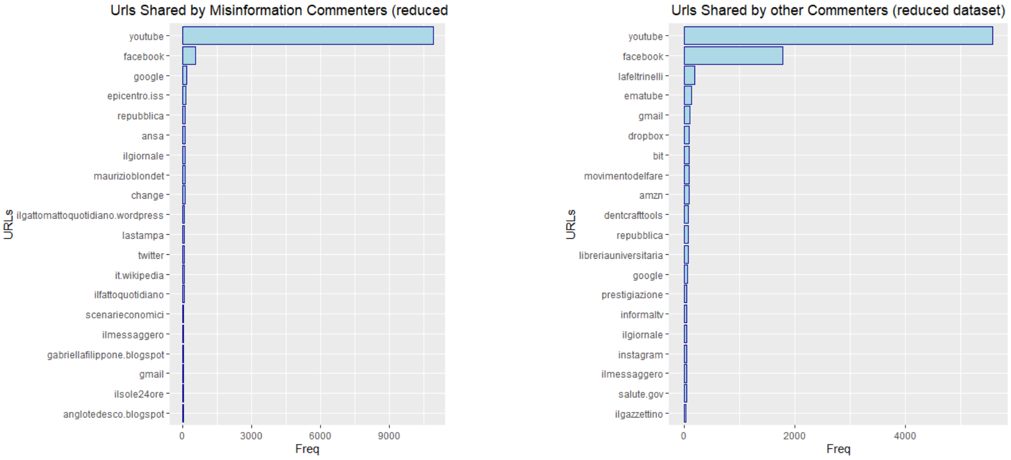


Figure 3.2: Domains shared by commenters in the reduced dataset, divided by *leaning*.

blacklists. The multiple distance measures are necessary because each one captures a distinct difference types between strings. For each distance $d^{(i)}, i = 1, \dots, 8$ and for each channel title s we compute the distance with each string t in the blacklist, obtaining a tensor of rank 3, containing members in the blacklists, in the channels, and the distance measure. We are interested in the best match, therefore we take the minimum value(s) $min_t d^{(i)}(s,t)$ as the best match(es) $d_s^{(i)}$. After computing d_s for each s it is necessary to manually inspect the list of best matches for each $d^{(i)}$. It is possible to choose an arbitrary threshold to speed the process and reduce the mole of matches to inspect, but the choice has to take into account the codomain of $d^{(i)}$ and its characteristics. Then the resulting distances were ordered by increasing distance and manually checked for possible matches. Some results are shown in Figures 3.8 and 3.9.

After the inspection there, 16 channels were labelled as misinformation. There is no guarantee that these are all misinformation channels in the dataset, but it is guaranteed that the channels are indeed misinformation spreaders. Therefore we can divide videos into “misinformation” and “other” categories.

Applying a left join between Channels and Videos using the channels’ id, 351 videos were found to correspond in the Videos dataset. Using left join on these videos’ id with the Comments dataset we found 99618 comments from 29199 different users.

3.3 Preliminary Analysis

We begin our analysis by exploring the users’ engagement with misinformation, the terms they use, and analyze the URLs shared.

We introduce a measure of misinformation engagement, based on the proportion

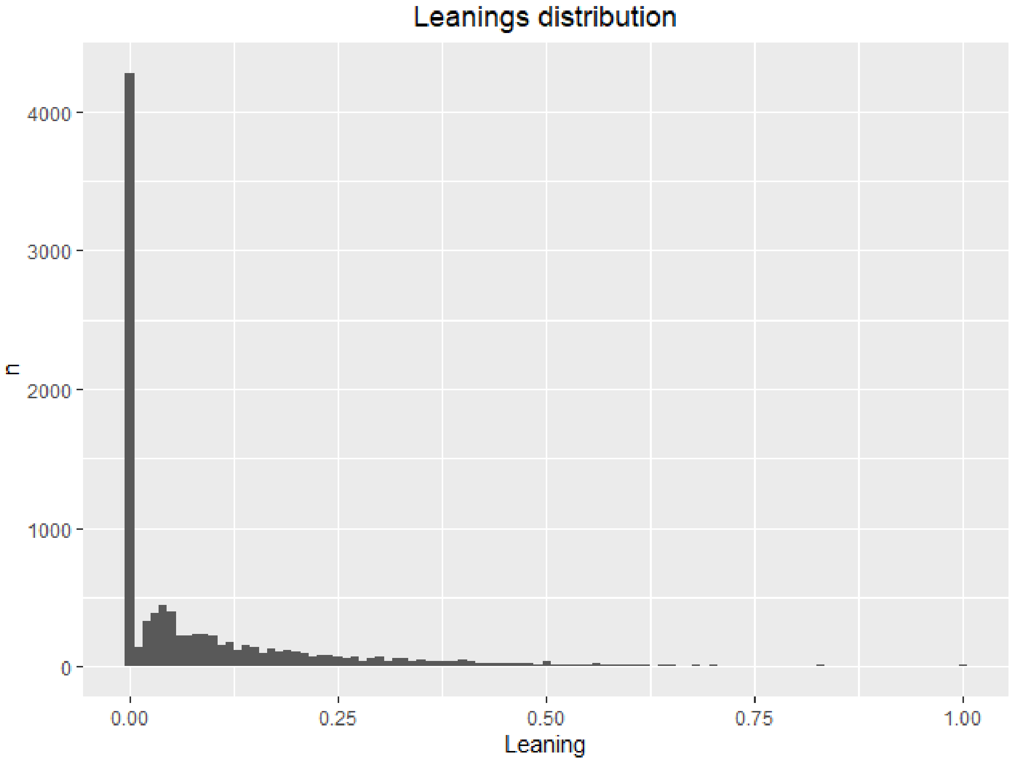


Figure 3.3: Distribution of the *leaning* statistics among users.

of shared comments in misinformation videos, and we call it *leaning*:

$$leaning_i = \frac{\text{n. comments in misinformation videos of user } i}{\text{total comments of user } i}$$

Since this statistic is based on comments, it does not give information on a user's personal beliefs about misinformation rhetoric.

We can see from the distribution of the users' leanings in Figure 3.3 that it is a bimodal distribution with modes at *leaning* values of 0% and $\approx$ 10%. The data shows an increase in number of comments when people consuming misinformation encounters those consuming mostly other kinds of content. A possible interpretation is that there are some users that comment misinformation videos to perform some form of debunking.

From the image we can observe that:

- The vast majority of the users have never commented misinformation videos identified with the blacklists;
- Most users that do comment misinformation videos comment mostly "other" videos;
- Only a small percentage of users comments mostly misinformation videos.

We can compare the lexicon used by users engaging in misinformation on YouTube by grouping comments into two non-overlapping groups of users:

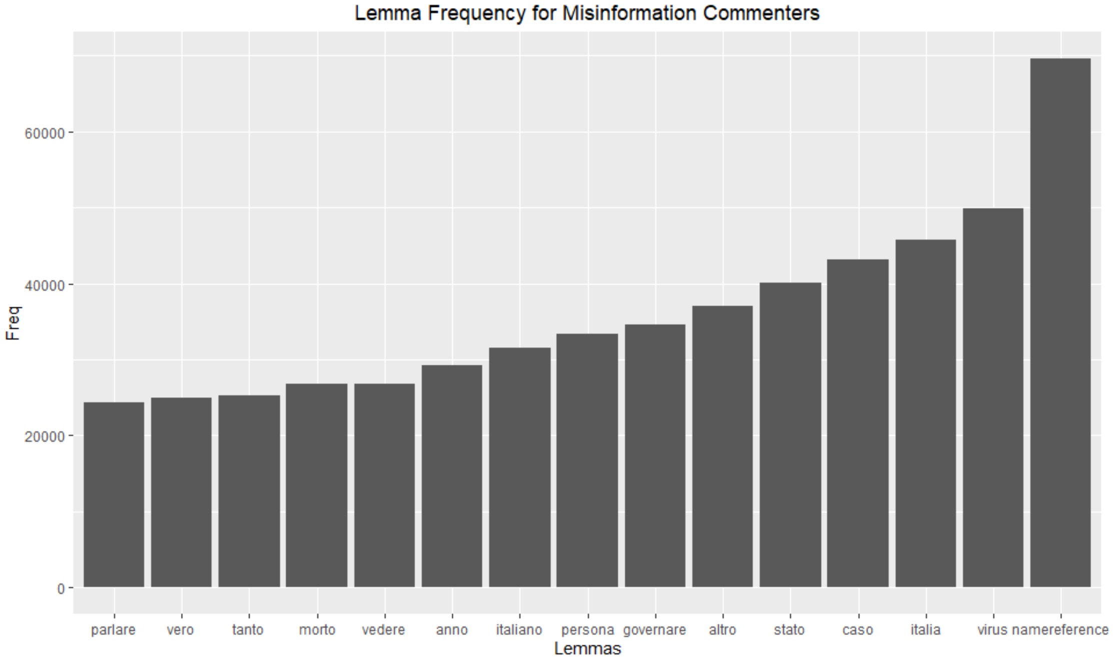


Figure 3.4: Lemma frequency for misinformation commenters: the most frequent term is ”namereference”, the word substituting any reference to usernames

- Users with *leaning* = 0 which have not engaged with misinformation
- Users with *leaning* > 0 who have engaged with misinformation

Figures 3.4 and 3.5 show frequencies of lemmas after stopwords removal, and substituting the reference to usernames with the keyword ”namereference”. We can see in Figure 3.4 that misinformation commenters have a much greater tendency to refer to other users, and given the usage of lemmas referring to state and government we can assume that they talk about politics more often. We also see in Figure 3.5 that the most common lemma for the other commenters is the word for ”chance”, which surpasses ”virus”.

The Figure 3.6 compares term frequencies in the two groups. The diagonal cuts the plot into terms more frequent in users engaging with misinformation and terms which are more frequent in other videos’ comments. The axes use logarithmic percentage scales. Since most points are lying on the diagonal we can conclude that most terms are present in equivalent proportion; however a minority of terms stand out as they are much more likely to appear in one group or in the other; this is in line with precedent studies [7]. By looking at the terms in the “Misinformation” half we can see proper names, names of misinformation spreaders like “byoblu” and “greg”, geopolitical blogs such as “anglotedesco”, and politicians names like “Clinton”.

On the “other” side we see among various common swear words like “cazzo” (dick) and “culi” (arses), that “Sgarbi”, an Italian right-wing politician who took controversial positions about face masks during the period these comments were downloaded, is more prevalent. Its presence could signify presence of users engaging in misinformation among the “other” videos, or the usage of this name to

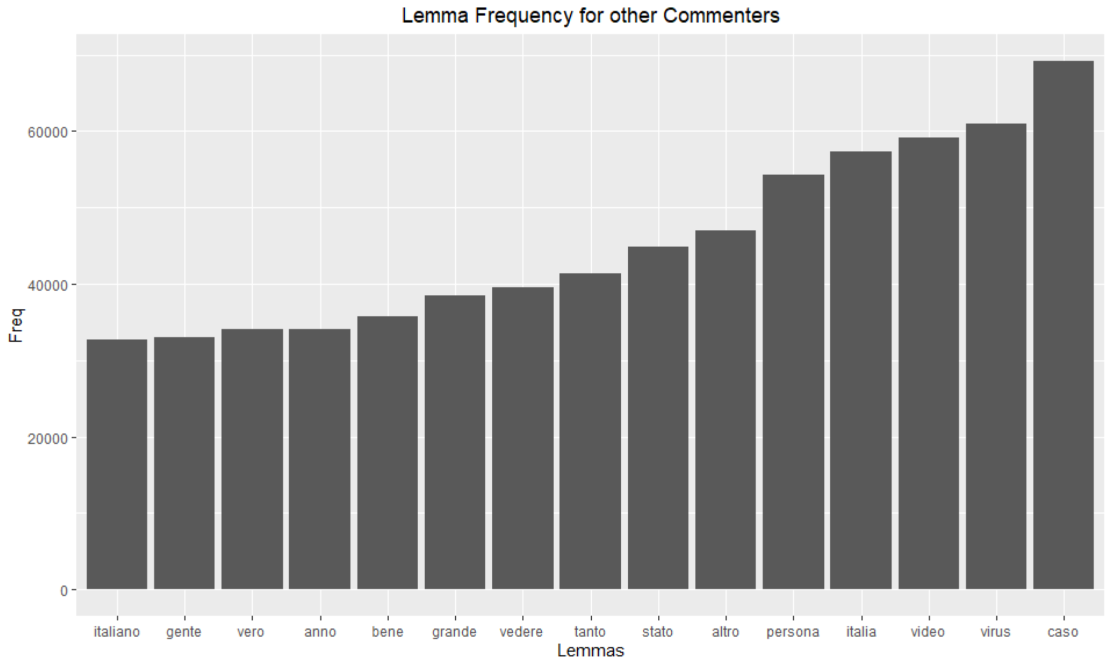


Figure 3.5: Lemma Frequency for other users: note that most frequent lemma is ”chance”

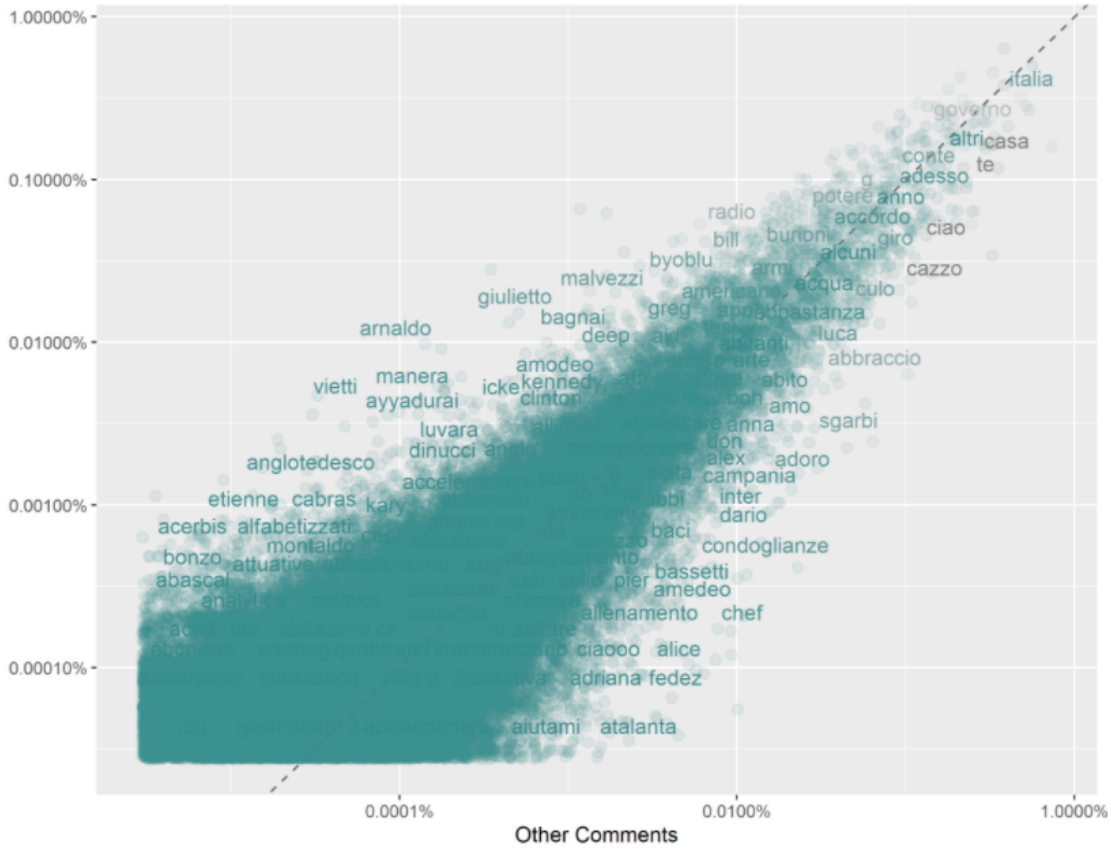


Figure 3.6: Term Frequency - Term Frequency plot comparing word frequency in the two groups using log-percentages.

	N	% Total	Mean Comments	Shared Url #	Shared Url %
All Users	323705	100%	4.4	1527421	100%
leaning= 0	294506	91%	3.2	971992	64%
leaning> 0	29199	**9%**	**16.3**	555429	**36%**
leaning> 0.5	10091	3.1%	4.4	50456	3.3%
leaning> 0.9	7458	2.3%	1.7	14706	1%

Table 3.3: *leaning*: user's percentage of comments under fact-checked misinformation videos

attack his positions, a phenomenon documented in [17].

This suggests that misinformation around COVID-19 is not isolated, but it is instead connected to other misinformation narratives. The larger prevalence of usernames has two possible explanations: the "weak" interpretation is that usernames are more frequent because the users examined are one tenth of the total, and therefore simple discussions between users make the term emerge; the "strong" interpretation is that the users are strongly connected because they either act as a community or have numerous debates, and therefore are more likely to reference each other.

In the comments there were 67395 shared links, of which 7299 were from comments in misinformation videos. Many YouTube links are a result of a platform's feature, which transforms numbers of the form *minutes* : *seconds* into a link to the same video, starting ad said minute and second, therefore we remove YouTube links. There is a substantial difference in the types and volume of URLs shared by users engaged in misinformation that can be seen in Figures 3.1 and 3.2.

From Table 3.3 we can see aggregated values for users separated by *leaning* values. We can see that the users with $leaning > 0$ have a high mean number of comments and high number of URLs shared. Indeed we see that 9% of users are sharing 36% of the links, which means that a user commenting misinformation videos is four times more likely to share an URL. We also observe that the mean number of comments drops significantly as users' leaning surpasses 50%; a similar behaviour is observed for shared URLs, that still remains higher than that of users with $leaning = 0$.

Another interesting information comes from the domains of these URLs. The URLs pointing outside YouTube underwent through the same string matching treatment as the channel names, and the results pointed that 43 sources of misinformation were shared using these URLs. Some of the more common misinformation URLs regarded extreme right blogs, while other links pointed towards, activist sites like "change.org", and official data sources like "epicentro.iss.it". This last case is documented in [17], where the authors found that official data sources were used in vaccine-hesitant debates on measles to reinforce their positions[17]; we cannot exclude other uses of the information in the URLs such as misinformation debunking.

It is notable how the comments of the videos, of which the topic is the coron-

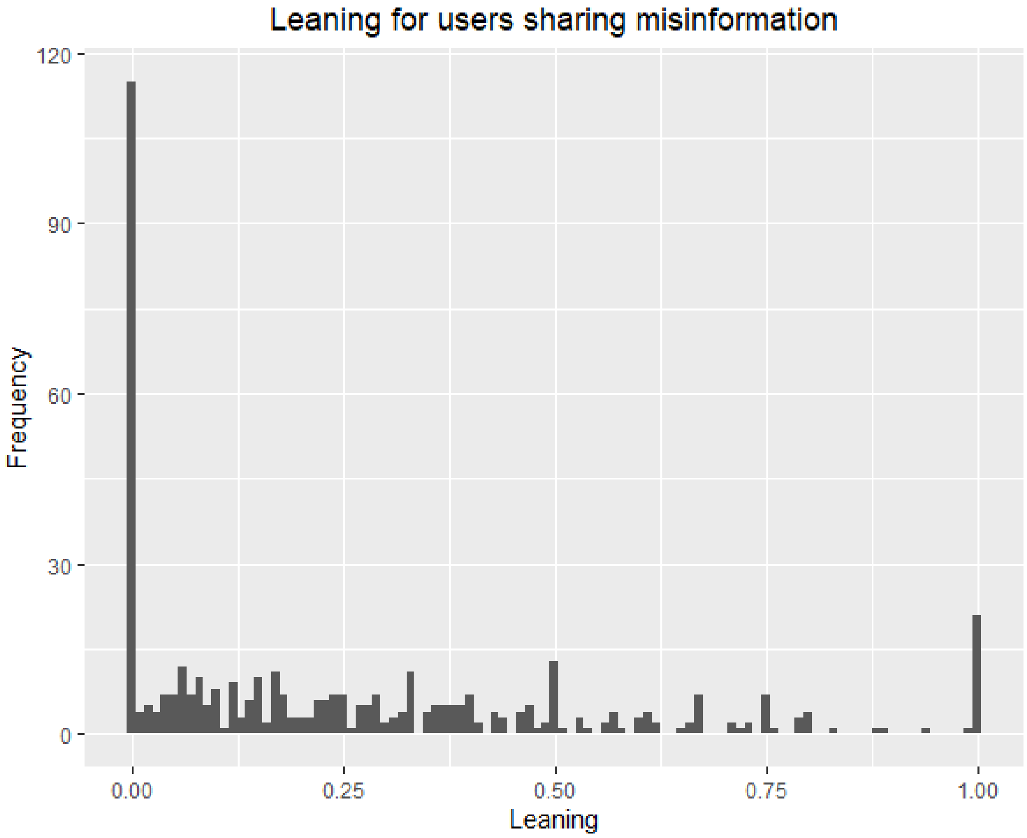


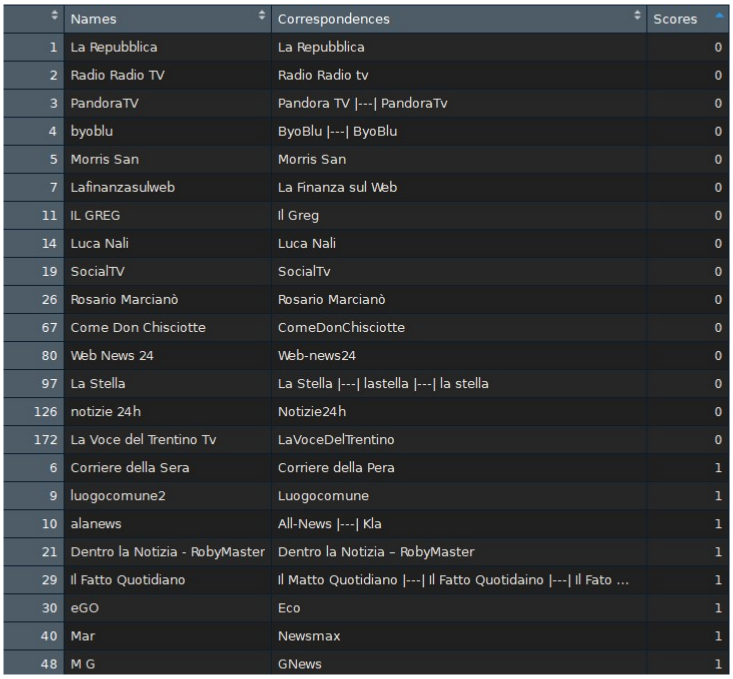
Figure 3.7: Users sharing misinformation have mostly a *leaning* value of 0.

avirus outbreak, brought a huge number of links towards an extreme right anti-Semite anti-European xenophobe blog: this may be an indicator that the debate around COVID-19 is politicized.

Furthermore, the large number of links to Facebook pages shows that the platforms' communities are connected. The links pointing to fact-checked misinformation sources is indeed a clear indicator of the user's preference; however the quantity of users sharing links is far too little to be of use on its own. All these information points to an interpretation of the *leaning* statistic: it reflects how much users are keen on commenting and take part of a discussion, and it is not an indicator of user preference. The truth of this statement is clear by observing Figure 3.7, which shows that users sharing misinformation do not necessarily comment on misinformation videos. This may be caused by the incompleteness of the blacklists used; another interpretation is that these commenters share their beliefs outside of the misinformation channels.

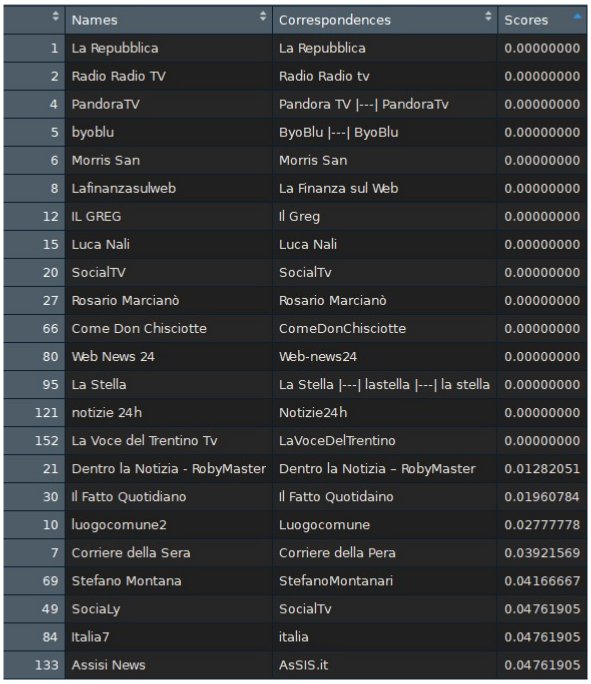
In conclusion it seems that users consuming misinformation are part of a vocal minority, which may adhere to right-wing ideologies and in general to a larger narrative than the one present in the videos. These users also tend to leave a much higher number of comments than the mean and shares a lot of links.

We are left with a measure of engagement, *leaning*, which does not indicate actual preference, and a label for a minority of users which shared misinformation sources in their comments.



	Names	Correspondences	Scores
1	La Repubblica	La Repubblica	0
2	Radio Radio TV	Radio Radio tv	0
3	PandoraTV	Pandora TV |---| PandoraTv	0
4	byoblu	ByoBlu |---| ByoBlu	0
5	Morris San	Morris San	0
7	Lafinanzasulweb	La Finanza sul Web	0
11	IL GREG	Il Greg	0
14	Luca Nali	Luca Nali	0
19	SocialTV	SocialTv	0
26	Rosario Marcianò	Rosario Marcianò	0
67	Come Don Chisciotte	ComeDonChisciotte	0
80	Web News 24	Web-news24	0
97	La Stella	La Stella |---| lastella |---| la stella	0
126	notizie 24h	Notizie24h	0
172	La Voce del Trentino Tv	LaVoceDelTrentino	0
6	Corriere della Sera	Corriere della Pera	1
9	luogocomune2	Luogocomune	1
10	alanews	All-News |---| Kla	1
21	Dentro la Notizia - RobyMaster	Dentro la Notizia – RobyMaster	1
29	Il Fatto Quotidiano	Il Matto Quotidiano |---| Il Fatto Quotidaino |---| Il Fato ...	1
30	eGO	Eco	1
40	Mar	Newsmax	1
48	M G	GNews	1

Figure 3.8: Results of exact and fuzzy string matching using Damerau–Levenshtein distance



	Names	Correspondences	Scores
1	La Repubblica	La Repubblica	0.00000000
2	Radio Radio TV	Radio Radio tv	0.00000000
4	PandoraTV	Pandora TV |---| PandoraTv	0.00000000
5	byoblu	ByoBlu |---| ByoBlu	0.00000000
6	Morris San	Morris San	0.00000000
8	Lafinanzasulweb	La Finanza sul Web	0.00000000
12	IL GREG	Il Greg	0.00000000
15	Luca Nali	Luca Nali	0.00000000
20	SocialTV	SocialTv	0.00000000
27	Rosario Marcianò	Rosario Marcianò	0.00000000
66	Come Don Chisciotte	ComeDonChisciotte	0.00000000
80	Web News 24	Web-news24	0.00000000
95	La Stella	La Stella |---| lastella |---| la stella	0.00000000
121	notizie 24h	Notizie24h	0.00000000
152	La Voce del Trentino Tv	LaVoceDelTrentino	0.00000000
21	Dentro la Notizia - RobyMaster	Dentro la Notizia – RobyMaster	0.01282051
30	Il Fatto Quotidiano	Il Fatto Quotidaino	0.01960784
10	luogocomune2	Luogocomune	0.02777778
7	Corriere della Sera	Corriere della Pera	0.03921569
69	Stefano Montana	StefanoMontanari	0.04166667
49	SociaLy	SocialTv	0.04761905
84	Italia7	italia	0.04761905
133	Assisi News	AsSIS.it	0.04761905

Figure 3.9: Results of exact and fuzzy string matching using Jaro-Winkler distance

Chapter 4

Lexical Clustering for Users' Classification

One of the approaches to classify misinformation sources is to resort to third-party fact-checking sites: this method allows to shift the burden of inquiring information reliability to specialized journalists, which provide blacklists of misinformation sources or evaluations on political bias. However this method's is limited by the quality of the fact-checking, and on whether the list is up-to-date.

With these lists then the simplest approach to classify users is to check a proxy for their preference; this proxy depends on the platform, and enables us to give different interpretation on its meaning depending on its nature. For example in [14] the authors analyze different platforms' networks: Facebook, Twitter, Reddit, and Gab. They build the interaction network using "follows" from Twitter, "likes" and comments for Facebook and Gab, and comment on submissions or on another user's comment for Reddit. Likes are a clear indication of user preference, while comments alone to not necessarily imply an endorsement of the commented content's position.

Another study [17] focuses on the lexical properties of commenters in social media. In their study they are able to build the interaction network of vaccine-promoters and vaccine-hesitant users, and to use it to label users; then they train logistic regression predictor using BOW features of the commenters to predict their community membership; their results suggest that there is a separation between BOW features used by these two communities. Furthermore they find that vaccination-hesitant communities contain also other types of misinformation, like conspiracy theories, anti-European sentiments, and far-right messages.

In another recent study the authors were able to find a high correlation between lexical properties of Facebook commenters belonging to the same community [7].

These results lead us to investigate an approach that exploits the characteristics of YouTube commenters' BOW features to obtain information on their preference, information which is not directly available. Then we pose the following questions:

1. Is it possible to cluster users commenting misinformation content through lexical clustering?

2. Are we able to tell if clustering users using lexical content gives us different information from a clustering based on commented videos? (i.e. does lexical clustering provide similar clustering to network clustering?)

3. Is it possible to cluster users which adhere to misinformation content?

A clustering based on lexicon instead is not directly dependent from watched videos, therefore we can see how well it performs in regards to *leaning*. However the conclusions are uncertain, as there is no one clustering method and it is possible that the background noise given by the terms which can be seen in the diagonal of Figure 3.6 could prevent any meaningful clustering.

Following [17], each user is treated as a document in a BOW model, and the terms in the document-term matrix are weighted using $Tf-idf$ weighting, which suppresses the weight of extremely common and extremely uncommon terms. $Tf-idf$ weighting guarantees that the terms are relatively meaningful to identify a document (user), and it is based solely on characteristics already present in the data.

In [17] lemmatization is applied to the terms; this procedure has the advantage of reducing the dimensionality of document-term matrices, and sometimes it improves results; however improvements are dependent on data and on the application, since lemmatization can eliminate important bits of information from the documents. We choose a conservative approach and did not apply lemmatization. To reduce the dimensionality of the DT matrix and distance matrix computations, users and terms were filtered by:

- Normalizing emphasized terms and interjections using regular expression (e.g.: aaahhhaaahh → hahaha);
- Replacing URLs with their domain name (e.g.: https://www.domain.name.given/example → domain name)
- Removing stopwords;
- Removing punctuation;
- Removing non-unicode characters;
- Substituting Mentions with a keyword $namereference$;
- Removing the terms appearing only in one document;
- Selecting users with at least 20 comments, to be sure to possess a large enough number of terms and fit the DTM and distance matrices into RAM.

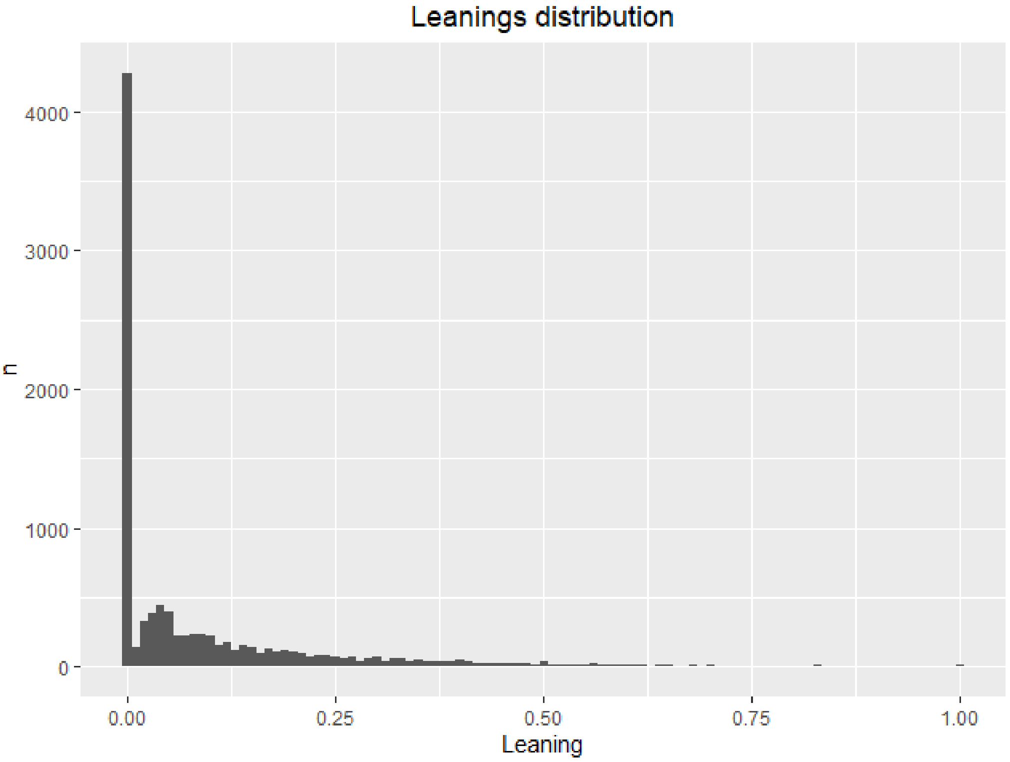


Figure 4.1: Leaning distribution of the reduced dataset

Then we apply $Tf - idf$ weighting.

The resulting DT sparse matrix is a 10353 users × 151771 terms. The users' *leaning* distribution can be seen in Figure 4.1: the reduction of the dataset has the effect of incrementing the proportion of users engaged a in misinformation to $\approx 59\%$. However the overwhelming majority has a low *leaning* value: of the 6092 users with *leaning* $> 0\%$ non-zero engagement only 33 have *leaning* $> 90\%$.

4.1 Clustering Analysis

The clustering algorithms employed are SKM and PAM for $K = 2, \ldots, 25$ using silhouette as heuristic, and HAC using the elbow method. Average Silhouette Scores are shown in Figure 4.2, 4.3, and 4.4.

For each clustering result we inspected the distribution of engaged users in the clusters, and produced a word-cloud using term frequency to inspect the contents of the comments.

When using HAC every available measure of cluster distance was used and analyzed, however for brevity we show here only the one calculated with the Ward cluster distance as it is the most meaningful.

Other methods which were discarded are various versions of Spectral Clustering

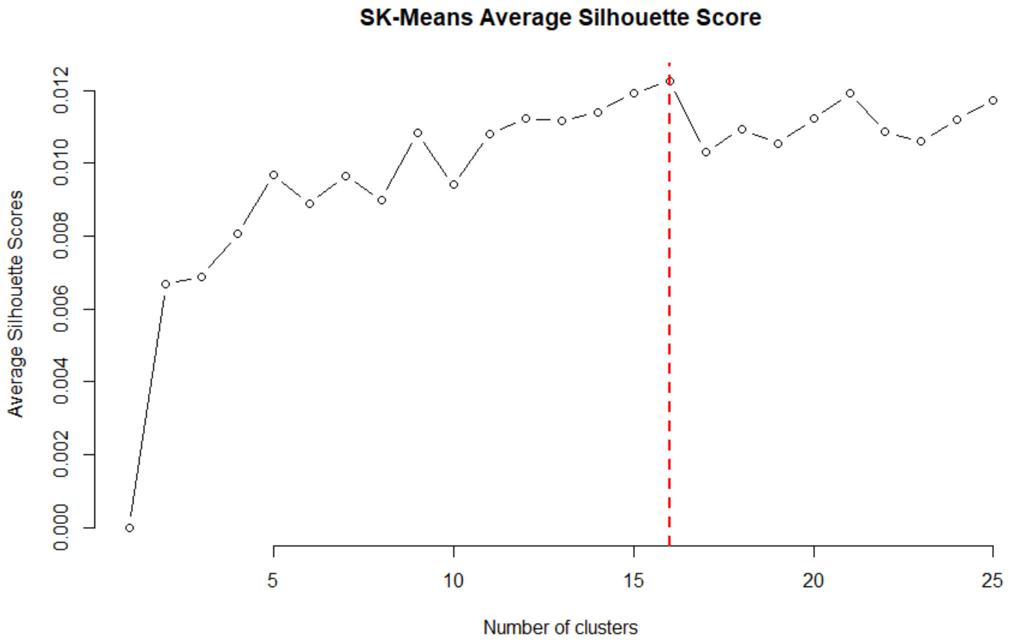


Figure 4.2: Spherical K-Means average silhouette score; we choose K corresponding to the highest score, shown in red.

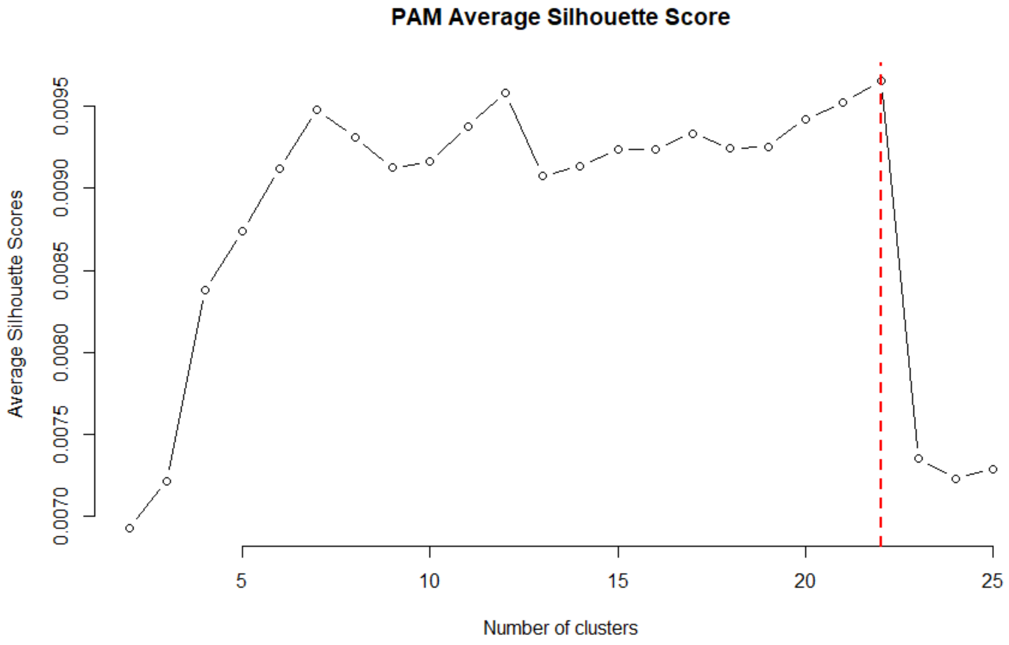


Figure 4.3: PAM average silhouette score; we choose K corresponding to the highest score, shown in red.

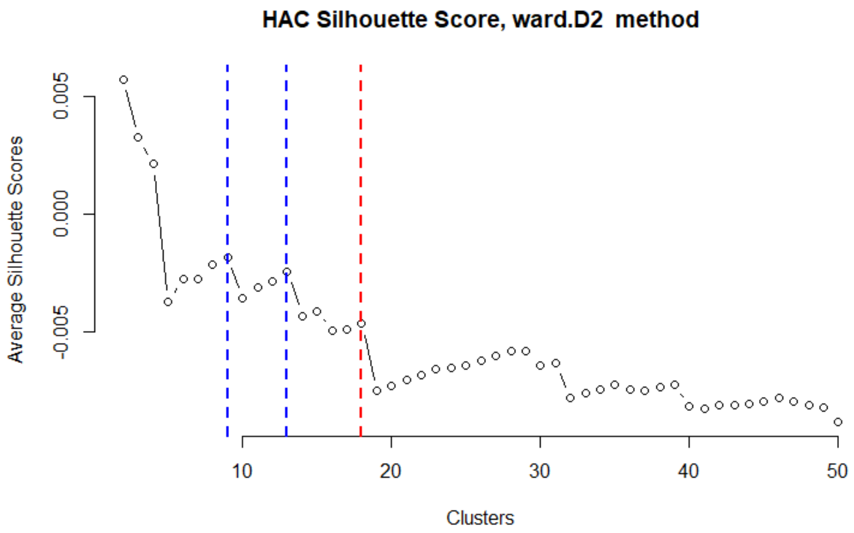


Figure 4.4: Hierarchical Agglomerative Clustering average silhouette score

using the Eigengaps heuristic, and Dominant Set Clustering using discrete replicator dynamics. These methods were discarded for their extremely unbalanced results, which are probably due to the effects of the curse of dimensionality.

4.2 Community Detection

Once the lexical clustering is completed we can cluster users based on their lexical usage, however it may be possible that users clustered in this way are simply users commenting on the same videos, speaking of the same topics, rather than users using a specific lexicon proper of narrative.

It is possible to assess these doubts by clustering users based on comments to the same video or channel, and comparing the two clusterings, with a measure of clustering similarity like the Rand index.

Community detection methods based on graphs of videos watched by the users is not very informative with respect to the *leaning* statistic, as it is obtained through comments on videos that users (presumably) watched, leading to a circular dependence between objective function and evaluation method. Therefore we cannot compare lexical and network clustering by their performance on *leaning*.

The User-Video bipartite graph was created for this purpose. The User-Video bipartite graph B_v is defined as $B_v = \{V_{u,v}, E_{u,v}\}$, where $V_{u,v} = U \cup V$, $U = \{\text{set of users}\}$ and $V = \{\text{set of videos}\}$. In B_v there is an edge (u, v) if user u commented video v; analogously in B_c there is an edge (u, c) if user u commented a video of channel c. The bipartite graph was then projected, maintaining only the users' modality. This resulted in the User-Video Projection Graph $G_v = \{V, E_v, W\}$, where V is the set of vertices representing the users, E_v is sets of edges, and W is a weighting function $W : E_v \to \mathbf{N}$ representing edge multiplicity.

In graph G_v there is an edge (u, v) if users u and v have commented the same video, and the weight $w(u, v)$ represents the number of videos co-commented. Therefore in G_v users are connected if they co-commented in at least one video, and the weight of their link is equal to the number of videos they co-commented.

For the analysis we prepared two analogous graphs for User-Channel interactions, B_c adn G_c; however they were discarded, as G_c possessed a far greater number of edges, making it densely connected: the RAM could not handle the weight matrix, and we were able to handle only an unweighted version of the graph; the great number of unweighted connections made it less informative than its Video counterpart.

Graph	Vertexes	Edges
B_c	13792	138254
B_v	30848	361772
G_c	10353	43253350
G_v	10353	14874404

Table 4.1: Statistics for the bipartite and projected networks. Note that G_c is close to the maxium number of edges in a graph

Then G_v was used to cluster users applying five community detection algorithms:

- Walktrap
- FastGreedy
- LabelPropagation
- Leading Eigenvector
- Multilevel (Louvain)

4.3 Evaluation

Here we try to answer to the first question: "Is it possible to cluster users commenting misinformation content through lexical clustering?"

The quality of the clustering procedures have been analyzed using the average *leaning* of the users in the clusters, and compared to each other using the Rand index. The visualization of Average Leaning for each cluster clarifies if clutter impedes lexical clustering or if this method is a viable option: if the clustering contains relatively small clusters with an average leaning value higher than the other clusters it is a good clustering; however if the average leaning is not very high or the clusters are of even size, then the cluster is not helpful.

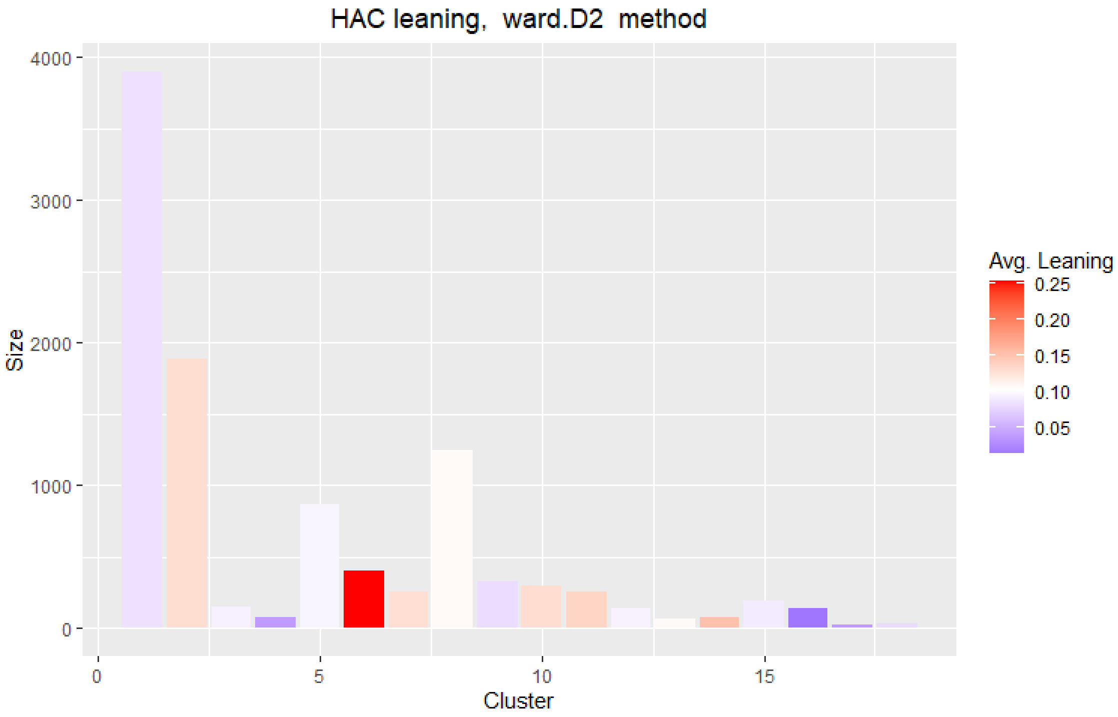


Figure 4.5: HAC clustering using Ward method: sizes and average leaning

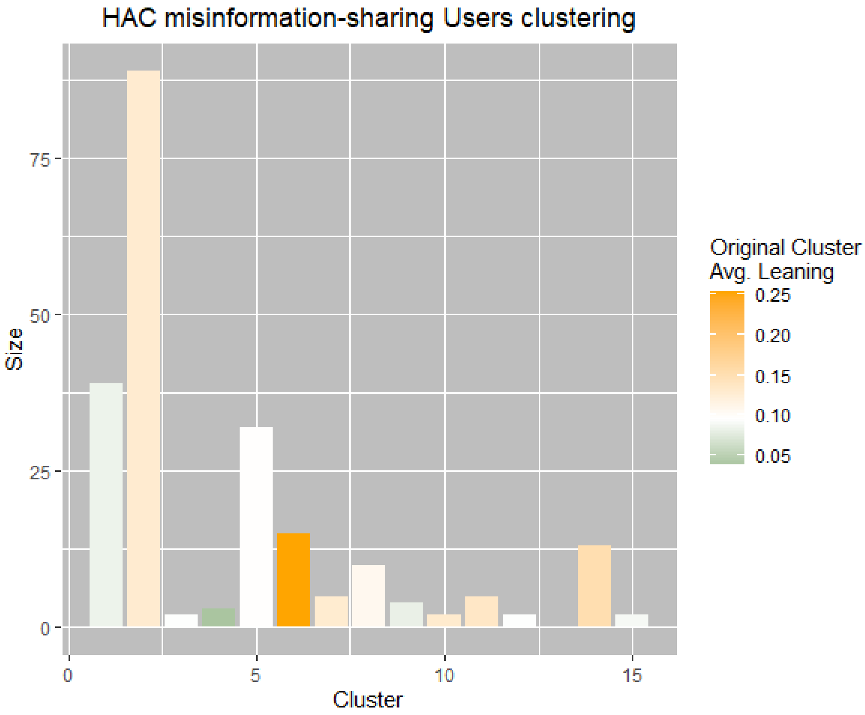


Figure 4.6: HAC clustering of users sharing URLs of misinformation sources; high and colored columns are sign of good clustering

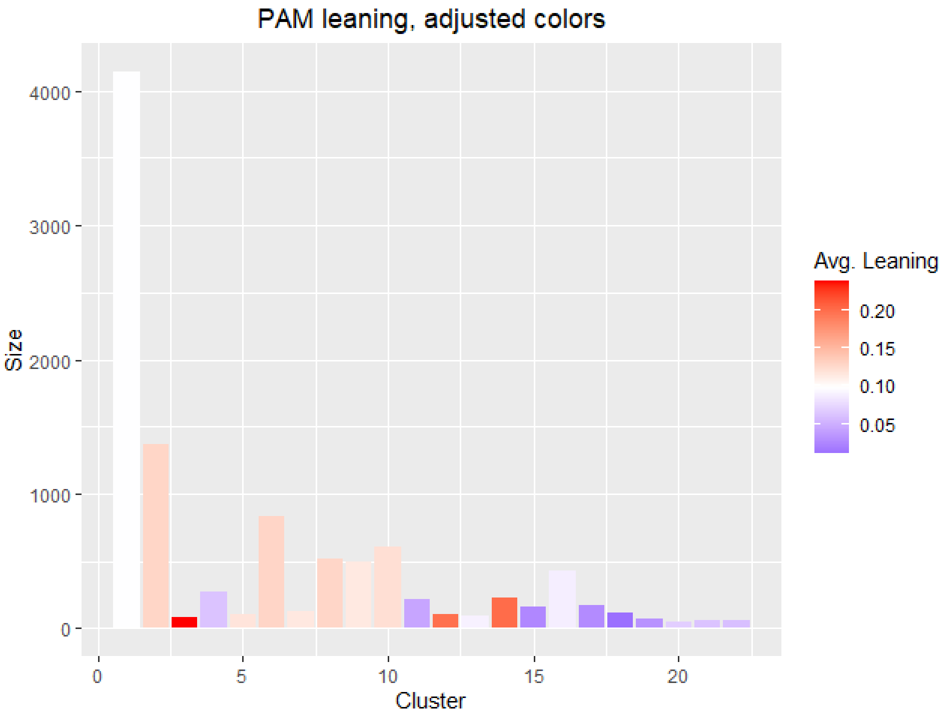


Figure 4.7: PAM clustering: sizes and average leaning

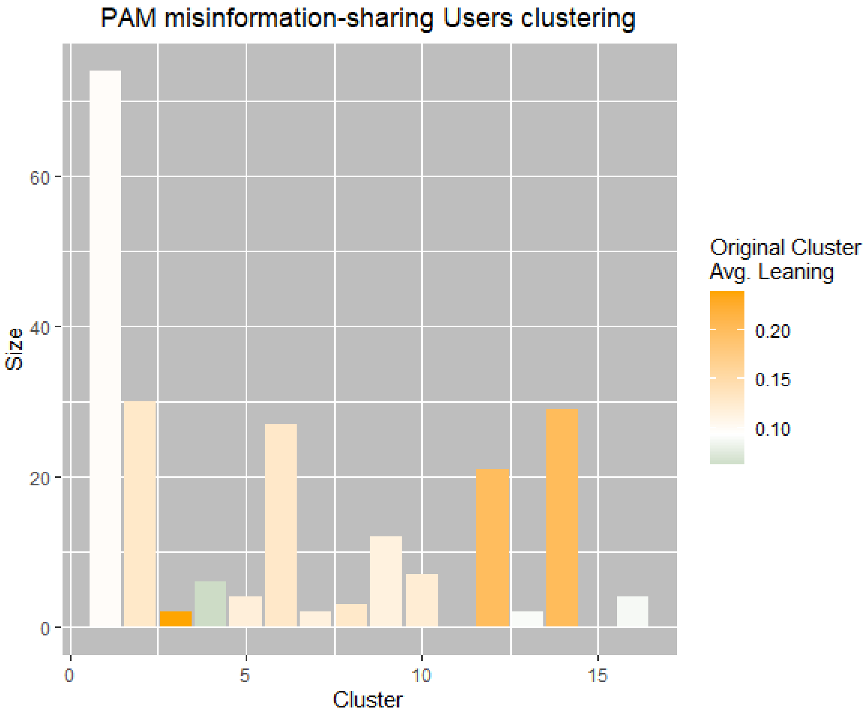


Figure 4.8: PAM clustering of users sharing URLs of misinformation sources; high and colored columns are sign of good clustering

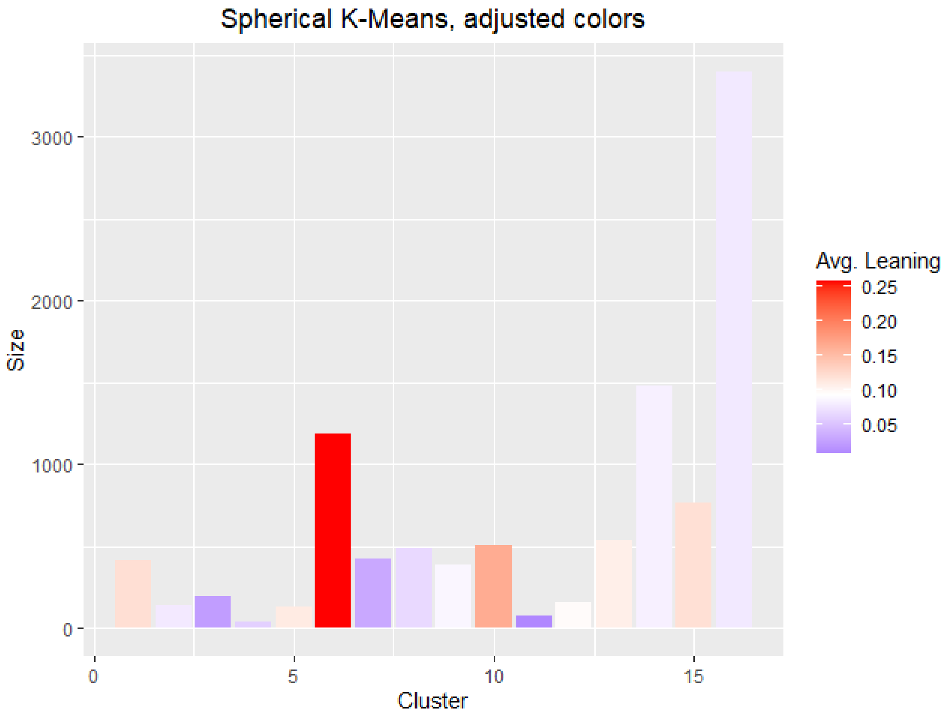


Figure 4.9: Spherical K-Means clustering: sizes and average leaning

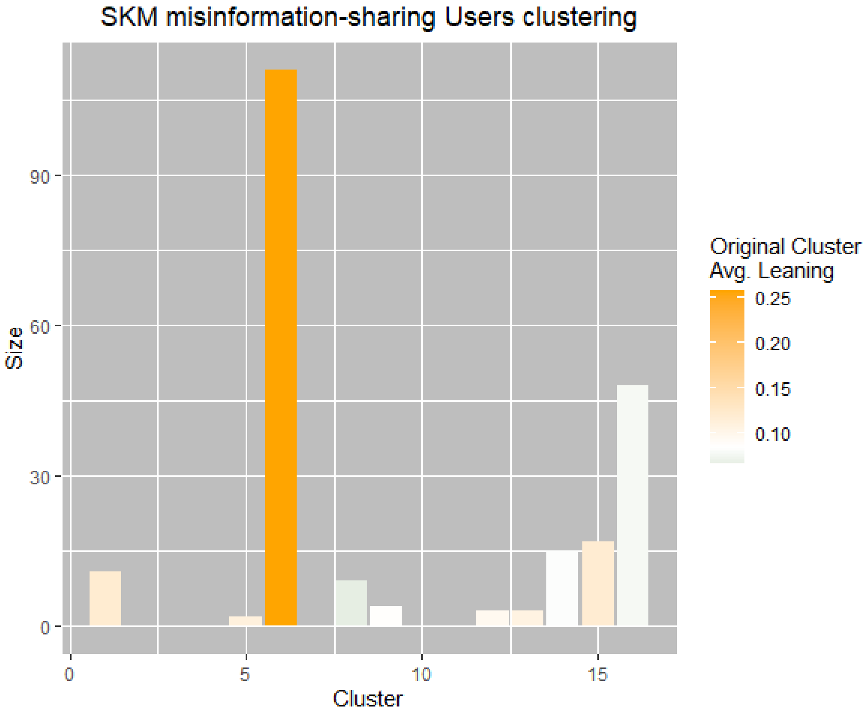


Figure 4.10: SKM clustering of users sharing URLs of misinformation sources; high and colored columns are sign of good clustering

We can see the results of clustering methods in Figures 4.5, 4.7, and 4.9. It is important to note that the range of Average *leaning* varies in each plot, and it presents information on the maximum value of the clustered users' average *leaning*, i.e. the higher the average *leaning*'s upper limit, the better.

Spherical K-Means (SKM) contains a large cluster with a 25% average *leaning*. PAM is interesting, because even if it contains the smallest maximum average *leaning* it contains three clusters with high average *leaning*, offering us a more diverse point of view on the contents of their members.

	SKM	PAM	HAC
SKM	1	0.79	0.79
PAM	0.79	1	0.75
HAC	0.79	0.75	1

Table 4.2: Rand Index between Lexical Clusterings; all values are at least 0.75.

It can be seen from their Rand index in Table 4.2 that the three clusterings possess similar characteristics, PAM and HAC being slightly less compatible.

The conclusion is that Lexical Clustering succeeded in clustering misinformation commenters possessing high *leaning* with varying degrees of success, and the algorithms seem to be somewhat interchangeable so far, as they lead to similar clusterings.

4.3.1 Word-clouds

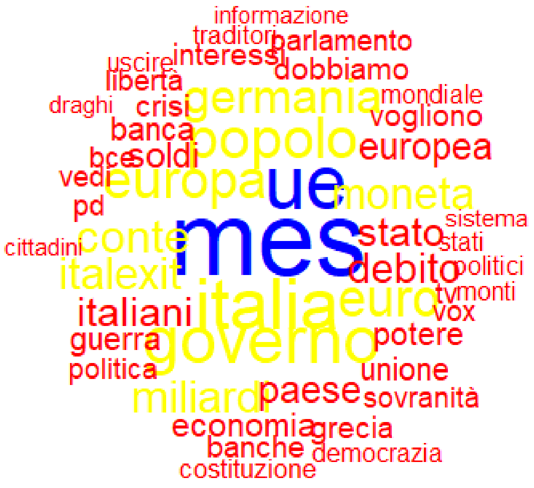


Figure 4.11: Word-cloud for HAC cluster n. 6

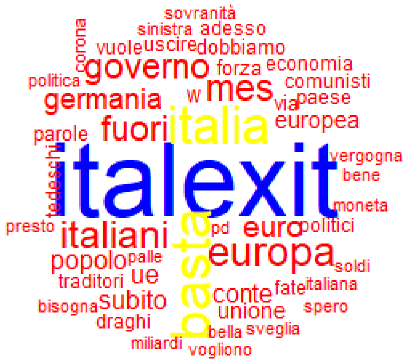


Figure 4.12: Word-cloud for PAM cluster n. 3

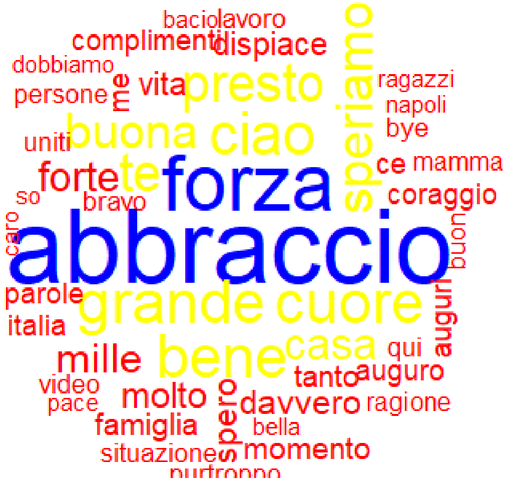


Figure 4.13: Word-cloud for SKM cluster n. 6

By looking at the terms used in the clusters from the Lexical Clustering process we obtain a set of term frequencies which can be visualized, for example as a word-cloud.

Even if this is far from an actual topic extraction, it still gives an indication on the topics of discussion, and helps formulating an hypothesis on why the users were grouped together.

For instance figures 4.12 and 4.11 clearly display the connection between anti-European narrative and YouTube misinformation narrative.

The word-cloud in 4.13 offers more room for interpretation: the cluster shown has the highest average *leaning* statistic, but its size is far greater than the others, and it does not seem to be about politics or misinformation in general, rather it contains words of support and consolation.

This could mean that the cluster groups several users which have nothing to do with misinformation; it could also mean that people engaging in misinformation seeks needy people on purpose, or that it is a needy person itself.

4.4 Community Detection Comparison

Here we try to answer the second question: "Are we able to tell if clustering users using lexical content gives us different information from a clustering based on commented videos?" The answer is given by comparing the results of the clusterings with the Rand Index. The end of this question is to to investigate the hypothesis that Lexical Clusterings simply reflect which videos are commented by users, which therefore are grouped together.

	WT	FG	LE	LP	ML
SKM	0.63	0.59	0.61	0.16	0.62
PAM	0.60	0.57	0.69	0.20	0.60
HAC	0.60	0.58	0.60	0.13	0.60

Table 4.3: Lexical Clusterings and Video Interaction Graph Network Clusterings Rand Indexes; all indexes are lower than their lexical-only counterpart

Method	# Clusters	Max cluster size	Min cluster size
SKM	16	3402	44
PAM	22	4147	49
HAC	18	3898	30
WT	98 (87)	4592	1
FG	19 (8)	5907	2
LE	14 (3)	5336	11
LP	14 (3)	10332	10
ML	19 (8)	5048	2

Table 4.4: Summary of Clustering Statistics. Numbers in parenthesis are the clusters in the network after isolated node removal.

Most community detection methods were able to find three large groups of ≈ 5000, ≈ 3000, and ≈ 1000 users, also their Rand index (not shown for brevity) shows high degrees of concordance between all clusterings (> 0.80) except label propagation, which clusters 99.7% of all users into a single group; a reason for this results may be that the network has too many connections between communities, leading to the prevalence of the label with most connections. The graph G_v had 11 isolated nodes, but this did not significantly affect clustering or Rand indexes; also it should be noted that users corresponding to the nodes had $leaning = 0$ and did not share any URL. In case of Walktrap there are 89 singleton clusters, and this is caused by small communities remaining trapped alone; however the method is able to find three large communities.

Whichever the reason, the relatively small Rand Indexes shown in Table 4.3 indicate that clusterings do not show particular similarities, the highest being PAM and Leading Eigenvector which show a slightly higher similarity then the others; for all other couples of values we can conclude that Lexicon and the Comment Interaction Network yield each different information on the behaviour of the users.

A possible interpretation of this behaviour is that users embracing misinformation narrative use words related to the narrative while commenting also other videos, as it is suggested by the prevalence of users with small *leaning* values inside clusterings. This information is independent from the location of the comments and reflects some kind of topic preference.

Here we encounter the limits of the BOW model: it strips words of their meaning and treats sentences using the same words but with opposite meaning as the same. Using this approach alone there is no accurate statement we can make on users' actual preference for one narrative or another without further information.

4.5 URL Preference

Method	Max Avg. *leaning*	Precision	Recall	F-measure
SKM	0.26	**0.47**	9.6%	**15.9%**
PAM	0.10	0.33	1.9%	3.6%
HAC	**0.13**	0.38	4.9%	**8.7%**
WT	0.21	0.54	3.4%	6.4%
FG	0.26	0.55	4.4%	**8.2%**
LE	0.22	0.41	3.3%	6.2%
LP	0.11	1.00	2.3%	4.5%
ML	0.22	0.54	3.7%	6.9%

Table 4.5: For each method we show statistics for the cluster with highest recall

This is why we need to use the most direct proxy for user preference: sharing a URLs pointing to a misinformation source is a clearer endorsement of misinformation narrative.

Considering the all 323705 users in the dataset, only 467 users, 0.14% of the total, shared at least one URL of a fact-checked misinformation source. In the reduced dataset considered for clustering there is a total of 239 users expressing their preference sharing misinformation in the comments; the relative percentage rises to approximately 2% of the considered 10353 users.

By looking at Figures 4.6, 4.8, and 4.10 we can see how effective were the clusterings at grouping actual user preference based on URLs. The colors are there to emphasize the original cluster's *leaning*. In the case of HAC clustering the three clusters with most URL-sharing users have a relatively low average *leaning*. Moreover the distribution of the two histograms 4.5 and 4.6 is similar, also to the naked eye, with the only exception of the first two clusters, which are also the two largest ones: this suggests that the clustering is not able to cluster misinformation users found using their shared URLs. The situation is slightly better for PAM, which is able to capture two numerous groups of misinformation-sharing users in clusters 12 and 14. It fails to capture most of the others, which are evenly spread in the other clusters, according to their size. Looking at Table 4.5 without any doubt SKM shows the best performance in terms of precision. The algorithm is able to group most URL-sharing users into cluster 6, which is also the one with the highest average *leaning*. The table also shows that all network clustering perform poorly in terms of recall. PAM's performance is last in terms of both precision and recall, surpassed even by LE, which clusters most users in a big cluster; network clustering is outperformed because of the greater size of its clusters, even if Fastgreedy is able to get close to HAC clustering. It is interesting to note that HAC was able to group together users with a low *leaning* in its cluster with most commenters sharing misinformation, which makes it interesting to study as the users are not

The fact that SKM is the best performing clustering is especially significative, because not only SKM is a relatively fast clustering method which does not require to compute any distance matrix, but also it is the one that is specifically designed for lexical clustering. This means that Lexical based Clustering is at least partly effective at identifying users endorsing misinformation.

A shortcoming of this clustering is that the word-cloud of the cluster with most misinformation-engaging users gives us an ambiguous interpretation, making it hard to tell that the cluster is related to misinformation without any more sources.

So we can try to answer to the question "Is it possible to cluster users which adhere to misinformation content?" Of the three methods, two could not cluster most of the users sharing misinformation; one method was able to group 111 of the 223 users. This amounts to almost 50% of the users grouped in a single cluster. Figure 4.14 shows that the method was able to cluster some users with *leaning* $= 0$, which is a good result. Admittedly the overall results of lexical clustering are not exciting per se; however, based on *leaning* and shared URLs lexical clustering leads to better results than those obtained through network clustering.

In conclusion, the method was barely successful: lexical clustering was able to find clusters of users identified as engaging in misinformation on YouTube, but many slipped past it, and one of the methods obtained the worst F-measure. Nonetheless two lexical clustering methods performed better than network clustering. The results look promising, but information on the actual performance is not at our disposal, therefore further research is necessary.

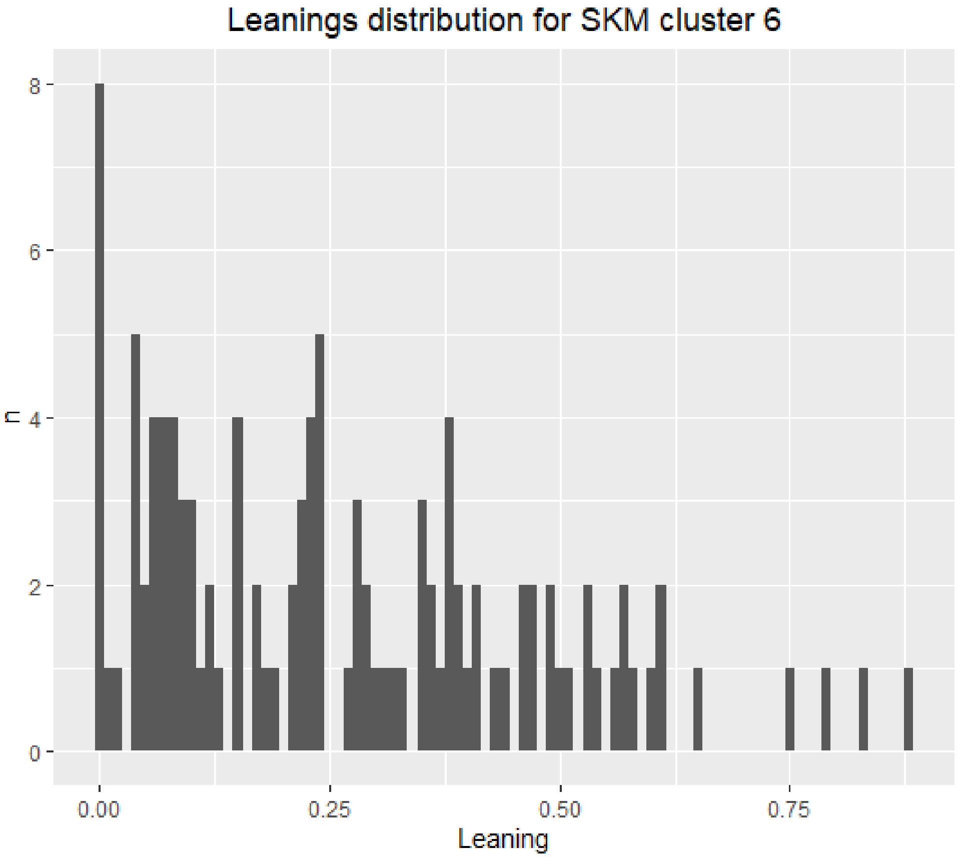


Figure 4.14: *leaning* of members of SKM cluster 6.

4.6 Limits

The approach presented in this thesis relies on many clustering methods that have some degree of randomness. Even though here we use a seed for result reproducibility[2] and each non-deterministic algorithm has been run multiples times to account for randomness[3], it is possible that more tries may lead to radically different results.

We did not perform analyses on the channels themselves, and did not study the video consumed by clustered users, but this could lead to the discovery of more misinformation sources, or on some insights on habits of YouTube misinformation consumers. Another limit is that we do not take into account that lexical usage changes with time, and it is data-specific.

But the greatest limit is that we do not possess any information on other classes of channels other than misinformation ones, and we do not have access to broad direct information on user preference like retweets or likes, so the results cannot be definitive: we cannot exclude the presence in the clusters of debunkers[4] by construction, and we possess very limited data on user preference, restricted to 2% of users.

[2] R seed set to 1

[3] K-Means - based methods took the best of at least 25 takes

[4] users that comment misinformation but do not endorse it.

Chapter 5

Conclusions

From the analysis of the YouTube comments on COVID-19 we note that users engaging in misinformation show the characteristics of a vocal minority: people commenting misinformation comments more, and, despite being only one tenth of the total population examined, they are up to four times more likely to share links in their comments. By analyzing the domains of the URLs pointing to misinformation sources it emerges that YouTube misinformation consumers show deeper knowledge of the misinformation landscape than that present on the platform, as they share sources which are not present in the platform.

Both URLs and frequent terms show that despite the topic of the videos being the COVID-19 outbreak, the comment's misinformation narrative is focused on geopolitical topics, and its political affiliation leans towards the right-wing. Also the numerous Facebook links suggest that there is a substantial connection between users in the two platforms.

Among the comments in the misinformation videos there were many URLs from institutional sources, but it has not been possible investigate more deeply on the reason why they were shared, leaving space for further studies on the issue.

This works has tried to measure the effectiveness of a Lexicon based approach in overcoming such obstacles, which is particularly helpful when other information is missing. Results of the clusterings, especially Spherical K-Means, show that it is possible to effectively group users that lean towards misinformation narrative using just BOW features of the commenters. It is unclear how many debunkers are contained in the clusters, but data suggests that they are indeed present.

The results could not be conclusive because of the lack of data, but they are promising, and further research could consist on evaluation of the approach on labeled data. Investigating the nature of channels commented by users in selected clusters, and exploring the net of shared URLs in the comments are other promising developments. Given the informative content that lexical analysis can yield, further research could focus also on ways to combine lexical and network information in context of lack of labels; more research is needed to explore effects of the time on lexical usage, which we deliberately neglected; another interesting option is that of improving the results given from the word-clouds and to apply a topic extraction method.

Finally, possible application of this work's results include a low-resource clustering to help preliminary analyses, and an heuristics for third-party fact-checkers to investigate on new misinformation sources and improve existing lists.

Chapter 6

Appendix

6.1 String matching

6.1.1 Exact String Matching

Problem Definition Given an alphabet Σ, a pattern $p = p_1 \ldots p_n$ and a text $t = t_1 \ldots t_m$, $p, t \in \Sigma^* \times \Sigma^*$, find a substring $t_{j',j} = t_{j'} \ldots t_j$ where $p_i = t_{j'+i-1} \forall i \in 1, \ldots, n$. There exist many algorithms solving the Exact String Matching problem in linear time with respect to pattern length and string length (e.g.: the Knuth-Morris-Pratt algorithm) [16]. One standard tool is Regular Expressions or Regex, a sequence of characters defining a search pattern.

6.1.2 Fuzzy String Matching

Given an alphabet Σ, a distance $d : \Sigma^* \times \Sigma^* \to \mathbb{K}$ a pattern $p = p_1 \ldots p_n$ and a text $t = t_1 \ldots t_m$, $p, t \in \Sigma^* \times \Sigma^*$, find a substring $t_{j,j'} = t_j \ldots t_{j'}$ which minimizes the distance $d(p, t_{j,j'})$. Usually $\mathbb{K}$ is $\mathbb{N}$, as the distances usually count the number of occurrences of symbols or operations; however some measures apply weighting, either for normalization purposes or to weight different coefficients, resulting in $\mathbb{K} = \mathbb{R}$ or $\mathbb{K} = [0, 1]$.

6.1.3 Distances

The topic of string distances contains many variations of two main families of distance measures which can be grouped as follows: edit distances, and set string distances.

Another way to group the distances is their co-domain: $\mathbb{N}$, $\mathbb{R}$, or $[0, 1]$.

Given an alphabet Σ and two strings $s, t \in \Sigma^*$, the edit distance $d(s,t)$ is a function of the the minimum number of a set of edit operations that transforms s into t. Many distance measures simply count the number of necessary edit operations, while others apply weighting for normalization purposes.

The simple edit operations usually considered are:

- Deletion: $cart \rightarrow car$
- Insertion: $cat \rightarrow cart$
- Substitution: $cat \rightarrow car$
- Transposition: $cats \rightarrow cast$

By allowing or disallowing different sets of edit operations one obtains different edit distance measures.

Implementations are usually $O(n^2)$ in the length of the strings.

Now, let us define Q-grams as subsequences of Q consecutive symbols. A Q-gram can be seen as a member of $\Sigma^Q, Q \in \mathbb{N}, Q \leq min\|s\|, \|t\|$.

Given an alphabet Σ and two strings $a, b \in \Sigma^*$, set string distances are distances that count the amount of Q-grams which a and b share, applying different weightings.

Set string distances may be considered as vector embeddings, or as literal set distances.

If $Q = 1$ and using supersets in stead of sets, these distances can be seen as edit distances allowing insertion and deletion with weight 1 and transposition with weight 0. Still they are more useful as vector embeddings than as edit distances.

Levenshtein

The Edit Distance by antonomasia. Allows Deletion, Insertion, and Substitution, and it usually applies unit weight to each operation.

A recursive definition is the following:

$$\operatorname{lev}(a,b)=\begin{cases}|a| & \text{if } |b|=0,\\ |b| & \text{if } |a|=0,\\ \operatorname{lev}(\operatorname{tail}(a),\operatorname{tail}(b)) & \text{if } a[0]=b[0]\\ 1+\min\begin{cases}\operatorname{lev}(\operatorname{tail}(a),b)\\ \operatorname{lev}(a,\operatorname{tail}(b))\\ \operatorname{lev}(\operatorname{tail}(a),\operatorname{tail}(b))\end{cases} & \text{otherwise.}\end{cases}$$

where $tail(a)$ returns the string a without the first character.

Example: $d_{lv}(\text{cast, acute}) = 4$

- *cast → deletion → ast*
- *ast → insertion → acst*
- *acst → substitution → acut*
- *acut → insertion → acute*

A property which is important for our purposes is that Levenshtein distance can be bounded:

- $mind(a,b) = ||a| - |b||\forall a, b \in \Sigma^* \times \Sigma^*$
- $mind(a,b) = 0$: indeed if $a = b$ the minimum number of operations to transform a into b is 0
- $maxd(a,b) = max(|a|,|b|)\forall a, b \in \Sigma^* \times \Sigma^*$

We can therefore see that it is possible to normalize Levenshtein distance by dividing by $max(|a|,|b|)$; by doing so however the distance loses other properties (it is not a metric anymore).

Damerau-Levenshtein

A variation of the Levenshtein distance, allows Transposition among the basic edit operations.

$$d_{a,b}(i,j)=\min\begin{cases}0 & \text{if } i=j=0\\ d_{a,b}(i-1,j)+1 & \text{if } i>0\\ d_{a,b}(i,j-1)+1 & \text{if } j>0\\ d_{a,b}(i-1,j-1)+1_{(a_i\neq b_j)} & \text{if } i,j>0\\ d_{a,b}(i-2,j-2)+1 & \text{if } i,j>1 \text{ and } a[i]=b[j-1] \text{ and } a[i-1]=b[j]\end{cases}$$

Example: $d_{dl}(cast, acute) = 3$

- $cast \to transposition \to acst$
- $acst \to substitution \to acut$
- $acut \to insertion \to acute$

Optimal String Alignment

Also known as Restricted Demarau-Levenshtein distance. It differs from the former in allowing the same substring to be edited only once. As an example let us show the DL distance versus OSA with the same two strings: d_{dl}(ca, abc) = 2:

- $ca \to transposition \to ac$
- $ac \to insertion \to abc$

d_{osa}(ca, abc) = 3:

- $ca \to deletion \to a$
- $a \to insertion \to ab$
- $ab \to insertion \to abc$

Longest Common Substring

Another variation of the Levenshtein distance which allows only two operations: Deletion and Insertion Example:

$d_{lcs}(\text{cast}, \text{acute}) = 5$:

- cast → *deletion* → ast
- ast → *insertion* → acst
- acst → *deletion* → act
- act → *insertion* → acut
- acut → *insertion* → acute

Hamming Distance

An Edit Distance which allows only for substitution. This string metric is not particularly useful for fraud detection and related tasks because it returns ∞ for strings with differing character number.

Q-gram Distance

Given two strings $s, t \in \Sigma^*$, let A and B the sets of Q-grams of s and t. The Q-gram distance is defined as

$$d_{Q-gram}(A, B) = \sum_i \|A_i - B_i\|$$

This is the simplest set string distance: it counts the number of Q-grams which are not shared.

For $Q = 1$ $d_{Q-gram}(s, t) = 0$ if t is a permutation of s.

there is a variation in which A and B are supersets.

Jaccard Distance

Let us define Jaccard similarity as

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Since $0 \leq J(A, B) \leq 1$, Jaccard distance is defined as

$$d_J(A, B) = 1 - J(A, B)$$

This distance is a normalization of the Q-gram distance.

Cosine Distance

The cosine between two vectors can be derived from the definition of dot product: $\mathbf{A} \cdot \mathbf{B} = \|\mathbf{A}\| \, \|\mathbf{B}\| \cos\theta$

Let $\mathbf{A}$ and $\mathbf{B}$ be vector embeddings of s and t, or their Q-grams. Cosine similarity is defined as

$$cos(\mathbf{A}, \mathbf{B}) = 1 - \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = 1 - \frac{\sum_i^n \mathbf{A}_i \mathbf{B}_i}{\sqrt{\sum_i \mathbf{A}_i^2}\sqrt{\sum_i^n \mathbf{B}_i^2}}$$

Since $0 \leq cos(\mathbf{A}, \mathbf{B}) \leq 1$, we can define cosine distance as

$$d_{cos}(\mathbf{A}, \mathbf{B}) = 1 - cos(\mathbf{A}, \mathbf{B})$$

It is in practice another way to normalize Q-gram distance: indeed if $\mathbf{A}, \mathbf{B} \in 0, 1^N$, then $cos(\mathbf{A}, \mathbf{B}) = |\mathbf{A} \cap \mathbf{B}|/(|\mathbf{A}||\mathbf{B}|)$.

Another property of cosine distance relates it with Euclidean distance: when $\|\mathbf{A}\|^2 = \|\mathbf{B}\|^2 = 1$ we have:

$$\begin{aligned} \|\mathbf{A} - \mathbf{B}\|^2 &= (\mathbf{A} - \mathbf{B})^\mathsf{T}(\mathbf{A} - \mathbf{B}) \\ &= \|\mathbf{A}\|^2 + \|\mathbf{B}\|^2 - 2\mathbf{A}^\mathsf{T}\mathbf{B} \\ &= 2 - 2\mathbf{A}^\mathsf{T}\mathbf{B} \\ &= 2(1 - cos(\mathbf{A}, \mathbf{B}) \\ &= 2 \cdot d_{cos}(\mathbf{A}, \mathbf{B}) \end{aligned} \tag{6.1}$$

Cosine similarity is more commonly used as an information retrieval tool than as string distance. The main drawback of its use as string metric is its invariance to permutations.

Jaro distance

The last metric proposed tries to address the problem of transposition invariance for set-based string distances.

The Jaro similarity between two strings s and t is defined as

$$sim_j(s, t) = \begin{cases} 0 & \text{if } m = 0 \\ \frac{1}{3}\left(\frac{m}{|s|} + \frac{m}{|t|} + \frac{m-T}{m}\right) & \text{otherwise} \end{cases}$$

where:

- $|s|$ is the length of string s
- m is the number of "matching" characters
- T is half the number of transpositions

Two characters "match" if they are at most $\left\lfloor \dfrac{\max(|s_1|, |s_2|)}{2} \right\rfloor - 1$ character apart.

Jaro similarity is bounded by $0 \leq sim_j(s, t) \leq 1$

Jaro-Winkler distance

Given a Jaro similarity between two strings s and t, the Jaro-Winkler similarity is defined as

$$sim_w(s,t) = sim_j(s,t) + lp(1 - sim_j(s,t))$$

where

- l is the length of the common prefix up to 4 characters
- p is a constant scaling factor $0 \leq p \leq 1/4$, usually $p = 0.1$

Then Jaro-Winkler distance is defined as

$$d_w(s,t) = 1 - sim_w(s,t) = lp(sim_j(s,t) - 1) - sim_j(s,t)$$

Jaro-Winkler is bounded by $0 \leq d_w(s,t) \leq 1 \iff 0 \leq p \leq 1/4$. Jaro distance can be seen as a particular case of Jaro-Winkler where $p = 0$.

Distance	Insert	Delete	Substitute	Transpose	Codomain
Levenshtein	1	1	1	∞	$\mathbb{N}$
DL	1	1	1	1	$\mathbb{N}$
OSA	1*	1*	1*	1*	$\mathbb{N}$
LCS	1	1	∞	∞	$\mathbb{N}$
Hamming	∞	∞	1	∞	$\mathbb{N}$
Q-gram	1**	1**	1**	0	$\mathbb{N}$
Jaccard	1**	1**	1**	0	$[0,1]$
Cosine	1**	1**	1**	0	$[0,1]$
Jaro	1**	1**	1**	$[0,1]$***	$[0,1]$
JW	1**	1**	1**	$[0,1]$***	$[0,1]$***

Table 6.1: Table of distances. *:only once per substring. **: not explicitly defined, depends on Q ***: depends on p

6.2 Information Retrieval Concepts

6.2.1 Bag Of Words

The Bag of Words model for a document in a collection $d \in \mathbb{D}$ is a vector embedding $v(d)$ of the set of term frequency weights tf, or other weightings $w(tf) \in \mathbb{R}^+f$; i.e. in a document vector $v(d)$ each component is a function of tf_t for a particular term t.

This is a quantitative representation of the document which disregards the order in which the terms appear in the document.

This representation allows vector operations on documents, such as *cosine distance*.

The underlying assumption of this model is that document using the same terms will score higher, and therefore receive a smaller distance. The assumption that the terms' order does not matter works in practice.

Instead the usage of raw tf weights assumes that all terms are equally important, and this does not hold in practice.

6.2.2 TF-IDF Weighting

This weighting is a mechanism that attenuates the weights of extremely common terms. To do so it introduces the document frequency df_t of a term t in a document collection $\mathbb{D}$ with $N = |\mathbb{D}|$ documents, and is the number of documents containing the term t.

Then the *inverse document frequency* is defined as

$$idf_t = log\frac{N}{df_t}$$

. Therefore the idf_t of a rare term is high, whereas the idf_t of a frequent term is likely low.

Now, given a term t, and a document $d \in \mathbb{D}$, the term frequency-inverse document frequency $tf - idf$ is defined as

$$tf - idf_{t,d} = tf_t \times idf_{t,d}$$

and assigns to term t a weight that is:

- high when the term is frequent in a small group of documents
- low when the term is uncommon or occurs in many documents
- lowest when it occurs in most documents

6.3 Clustering

Unsupervised learning is the branch of machine learning in which the algorithm is not provided with any explicit feedback on the result of its performance. The technique aims to discover patterns arising from the data

Clustering is the most known unsupervised learning task. Clustering algorithms can start either from a data matrix of n items and m features, or from an $n \times n$

(dis)similarity matrix in which some (dis)similarity measure is applied between each pair of items. Usually the distance (similarity) measure is required to be symmetric, but there are exceptions.

The underlying hypothesis when applying any clustering technique to information retrieval tasks is that “documents in the same cluster behave similarly with respect to relevance to information needs” [26].

Clustering itself is divided into flat and hierarchical clustering.

6.3.1 Hierarchical Clustering

This technique builds a hierarchy of clusters during the process.

These algorithms can be further divided into:

- Agglomerative clustering algorithms: following a bottom-up approach they start from the singleton clusters of isolated items and aggregate them until the final cluster of all items is reached.
- Divisive clustering algorithms proceed in a top-down approach, progressively dividing the whole dataset until it reaches n singleton clusters.

Hierarchical Agglomerative Clustering

The HAC algorithm was published by Johnson in 1967 in a statistics and psychometrics journal [21].

The algorithm starts at step 0 from n items with m features, and the $n \times n$ distance matrix, obtained with some measure of distance between them; this is the initial “hard” clustering.

Then at step $k+1$ it iteratively aggregates the two most similar items until step $n-1$, producing a new cluster with each iteration.

This procedure implies the choice of a measure of closeness between clusters; the most common options are:

- Complete-link: the distance between two clusters’ farthest members.
- Single-link: the distance between two clusters’ closest members.
- Average-link: the average of all pairs of distances between members of different clusters (UPGMA).

- Centroid method: the distance between the means of the two clusters' members.
- Ward-linkage: the distance between two clusters is how much the euclidean sum of squares will increase when we merge them [27].

Since the introduction of the algorithm the complete-link distance between clusters has been generally accepted, as single-link produces incoherent clusters, and average-link adds more computation but produces very similar or identical results to complete-link's. Its naive implementation's complexity is $O(n^3)$, but with the use of heaps it decreases to $O(n^2 log(n))$.

6.3.2 Flat Clustering

Flat Clustering algorithms are those that do not produce any hierarchy tree.

The single most important flat clustering algorithm is K-Means.

K-Means

K-Means starts from n items with m features, along the hyper-parameter k, indicating the number of desired clusters. It starts by randomly picking k items as cluster centers.

Then it alternates the two operations:

- assign the points closest to those centers to their respective clusters.
- compute the mean of each cluster and choose it as that cluster's new center.

K-Means benefits of some useful properties:

- It minimizes the distance from the mean of the i-th cluster following the objective function:
$$\underset{C}{\operatorname{argmin}} \sum_{i=1}^{k} \Big\{ \sum_{j \in C_i} \|x_j - \mu_i\|^2 \Big\}$$
- It is guaranteed to converge after a finite amount of iterations with minimal assumptions.
- Its time complexity is $O(kn)$, and does not require a distance matrix.

However it possesses a number of critical issues. Namely:

- requires the unknown parameter k
- sensibility to initialization
- sensibility to outliers
- it works only with spherical clusters
- performs poorly with high dimensional space

Some of these issues may be tackled with straightforward methods:

- several runs with a range of values for k
- multiple initializations

These expedients mitigate the effects of the issues, albeit harming the time complexity advantage.

The other issues are less straightforward.

Partitioning Around Medoids

A variant of K-Means is PAM, also known as K-Medoids. It consists on picking the item closest to the center of the cluster as medoid instead of calculating the mean each time.

This algorithm can be computed using the distance matrix instead of the original data. Its exact solution is NP-Hard, however there are many heuristics for which the time complexity is at least $O(n^2)$.

It is strictly connected to Voronoi tessellation. One advantage of PAM over K-Means is its ability to better handle outliers.

Spherical K-Means

This is a lesser known variation of K-Means, specifically designed for text clustering [20]; it tries to address the issue of over-representation of long texts by using cosine dissimilarity to cluster them, which is equivalent to use the Euclidean dissimilarities of the projections of the feature vectors onto the unit sphere. By design it is more suitable to highly dimensional spaces than K-Means.

The main difference from K-Means is that it minimizes the function

$$\underset{C}{\operatorname{argmin}} \sum_{i=1}^{k} \{ \sum_{j} 1 - cos(x_j, \mu_i) \}$$

. The algorithm is not guaranteed to converge to the optimum, and different heuristics may lead to better partitions.

The algorithm does not need a distance matrix, and its implementations are optimized for sparse matrices.

6.3.3 Network Clustering

Also known as community detection, network clustering clusters vertices of a graph according to its properties.

The clusters obtained using such methods are known also as modules, groups, and communities.

Graph Theory Concepts

A graph is a structure that used to abstract relationships. It is formally defined by a set of vertices and a set pairs of vertices called edges, or arcs. Graphs can be directed or undirected, depending on whether the edges are respectively an ordered or an unordered pair. Edges can also be weighted or unweighted.

Formally a weighted graph G is defined as:

$$G = \{V, E, W\}$$

where $E \in V \times V$ and $W : E \to \mathbb{A}$, usually $\mathbb{A} = \mathbb{R}$. A graph can be represented as adjacency matrix, whose elements are defined as $A_{i,j} = W(i, j)$, where $i, j \in V, (i, j) \in E$.

The degree $deg(v)$ of a vertex v is the number of edges passing through v; the weighted degree $wdeg(v)$ of a vertex is the sum of weights of the edges passing through v. By the handshaking lemma, the following equation holds for any graph $G = \{V, E\}$:

$$\sum_{v \in V} deg(v) = 2|E|$$

Given a random graph G and two nodes u, v the probability of an edge existing between them is $p_{ij} \approx \frac{deg(i)deg(j)}{2|E|}$

A (in)finite walk of length n is an ordered sequence of $n+1$ vertices $(v_1, \ldots, v_{n+1}) : (v_i, v_{i+1}) \in E \forall i = 1, \ldots, n$.

Bipartite Graphs

Bipartite graphs are graphs whose vertices can be divided into two subsets $V = X \cup Y, X \cap Y =$ where $\forall (u, v) \in E u \in X \wedge v \in Y \vee u \in Y \wedge v \in X$, i.e. no edge starts and ends in the same subset. Given a bipartite graph $B = \{V_b = X \cup Y, E_b\}$, a Bipartite Graph Projections is a graph $G = \{V_p, E_p\}$ whose vertex set V is either X or Y, and $(u, v) \in E_p \iff \exists z \in V_b : (u, z), (v, z) \in E_b$. For a bipartite graph there are two possible bipartite projections, one for X, and one for Y.

Bipartite projections can be unweighted or weighted. A simple method to weight an edge $(u, v) \in E_p$ for the bipartite projection of X is $w(u, v) = |\{z : (u, z) \in E_p \wedge (v, z) \in E_p\}|$, i.e. the number of elements in Y that connect u and v.

Modularity

Modularity of a graph is an intrinsic measure of goodness of a vertex clustering: high modularity corresponds to high number of edges starting leading to the same starting cluster, and small number of clusters starting from a cluster and ending into another.

This measure is sensible to the resolution of the graph, and it does not accurately capture small group's fitness of clustering.

Given c clusters we can define the community membership matrix $S \in \{0, 1\}^{n \times c}$ to have members S_{ij} set to 1 only if node i is member of cluster j. Then we define the modularity matrix B as

$$B_{ij} = A_{ij} - P_{ij}$$

where A is the adjacency matrix, and P is the matrix of expected number of edges between two nodes.

Finally modularity can be expressed as:

$$Q = \frac{1}{2|E|} Tr(S^T B S)$$

which more intuitively is the sum of expected number of edges of elements of the same cluster. For an unweighted undirected graph, the modularity is bounded by $[-\frac{1}{2}, 1]$.

Walktrap

A discrete random walk is a walk where at each discrete time step t a walker is on a vertex i and moves to another vertex j chosen randomly and uniformly among its neighbors. At each step the probability of the walker to move from i to j is

$P_{ij} = \frac{A_{ij}}{deg(i)} = (D^{-1}A)_{ij}$, where D is the diagonal matrix of of degrees $D_{ii} = deg(i)$, and A is the adjacency matrix.

A random walk process is represented by the matrix P^t having at each entry $P^t_{i,j}$ the probability that a random walk of length t starting at vertex i will end vertex j. If two verteces i and j are in the same community, P^t_{ij} will be also high, but the converse is not necessarly true. In fact the walks will tend to go towards high degree vertices.

Walktrap [31] exploits the random walk process' features: it first calculates the distance matrix according to the distance measure

$$d^{(t)}_{ij} = \|D^{-\frac{1}{2}}P^t_i - D^{-\frac{1}{2}}P^t_j\|$$

Then it employs a clustering algorithm to cluster the nodes (HAC is used in [31]).

Time complexity is $O(mn(H + t))$ where $H = O(n)$ is the height of the tree, and t is the length of the random walk. An heuristic is to choose $t = O(\log n)$, because of the exponential convergence speed of the random walk process, making the time complexity $O(mn \log n)$.

Given the nature of the random walk probability matrix it is tied closely to spectral clustering algorithms.

FastGreedy

FastGreedy [15] is a greedy hierarchical agglomerative clustering algorithm that tries to optimize modularity Q.

The algorithm starts with $|V|$ singleton clusters, and iteratively joins two clusters that produce the greatest increase in Q.

The algorithm is designed for sparse matrices, and has time complexity $O(|E|d \log |V|)$, where d is the depth of the resulting dendrogram.

Leading Eigenvector

This method [28] tries to maximize the modularity of the graph by finding the eigenvalues of the modularity matrix through spectral decomposition:

$$B = UDU^T$$

where $U = (u_1, u_2, \ldots)$ is the matrix of eigenvectors, and D the diagonal matrix of eigenvalues $D_{ii} = \beta_i$. Then Modularity can be rewritten as:

$$Q = \sum_{j=1}^{n} \sum_{i=1}^{c} \beta_j (u_j^T s_k)^2$$

.

Maximization of Q happens through the choice of c. This is proportional to the number of leading eigenvectors of B; i.e. c is the number of positive eigenvalues $+1$.

In practice the problem becomes a NP-hard N-cut problem, which is relaxed through an approximization parameter α and solved in polynomial time.

Label Propagation

This method, described in [32] initializes the nodes with $n = |V|$ unique labels at time $t = 0$; at time $t + 1$ the nodes synchronously choose their new label among their neighbors' according to a majority rule, using a uniform random tie breaking rule, eventually reaching stability.

This method favors densely connected groups and automatically determines the number of communities; however if the graph contains some topologies like bipartite and star graphs the labels do not converge, oscillating at each iteration, unless it is employed an asynchronous label.

The algorithm's advantage is speed and simplicity, given that each iteration takes $O(|E|)$ time, and that it may take as few as 6 iterations to terminate the algorithm on an Erdős - Rényi graph with 10000 nodes and average degree of 4.

Multi - Level

Also known as Louvain community detection [4], it is a hierarchical agglomerative algorithm that tries to maximize modularity Q. The algorithm alternates two phases.

Starting from a $n = |V|$ labelled graph, for each node i it considers each neighbour j of i and calculates the gain in modularity of removing i from its cluster and assigning it to j's cluster; then i is assigned to the community with the greatest ΔQ, and if $\Delta Q \leq 0$ i does not change community. This phase is repeated until it reaches a local maximum for Q. The order in which nodes are considered changes the result.

In the second phase the algorithm builds a new graph, whose represent are the communities found in the preceding phase, and the weights of the edges between the new nodes are the sum of the weights of edges of the old nodes between the communities. When the second phase ends, the first phase is repeated iteratively until there are no changes in the communities.

The algorithm has two main advantages: it has near-linear time complexity, and it circumvents the resolution limit of modularity thanks to its multi-level nature.

6.3.4 Clustering Evaluation

When inspecting a clustering result it is needed a measure of the goodness of the result. If there is no information on the objects being clustered then the only evaluation possible is through Intrinsic Evaluation measures; if we can count on information on the objects we can adopt Extrinsic Evaluation measures.

Intrinsic Evaluation Criteria

A typical Instrinsic Evaluation measure has two components: a measure of internal consistency of the cluster which has to be maximized, and a measure of extra cluster similarity which has to be minimized.

But good scores on internal criteria do not translate directly into good effectiveness in an application [26]. The validity of these criteria depend heavily on the assumptions that are made on the underlying data.

Furthermore, internal criteria introduce bias towards the algorithms that use the same cluster model (i.e. a K-Means intrinsic criterion will score higher with a distance based evaluation).

That being said, an intrinsic evaluation is typically used to score different runs of the same algorithm to tune its hyper-parameters. One example is the application of the Silhouette score on $K - Means$ to choose the "best" K, or to pick among different initializations.

Silhouette Score

Presented in [34], it is an Intrinsic Evaluation Criterion suitable for K-Means clustering evaluation . It assigns a value in the range $[-1, 1]$ to each item, indicating internal cohesiveness to its own cluster and dissimilarity to other clusters.

Assume N objects have been partitioned into K clusters, and let $d(i, j)$ be a measure of distance between two objects.

For an object $i \in C_i$, let

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, i \neq j} d(i, j)$$

be the average dissimilarity of i to all other objects of C_i. Then let

$$b(i, C) = \underset{C_k \neq C_i}{\operatorname{argmin}} \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j)$$

be mean distance of i to all points in the cluster that minimizes this distance.

Then the silhouette score of an item i is defined as

$$s(i) = \begin{cases} 0, & \text{if } |C_i| = 1 \\ s(i) = \frac{b(i)-a(i)}{\max\{a(i),b(i)\}}, & else \end{cases}$$

Then the silhouette coefficient heuristic consists in finding the value for K that maximizes the mean silhouette score:

$$SC = \underset{K}{\operatorname{argmax}}\, \bar{s}(K) = \frac{1}{K} \sum_{i=1}^{N} s(i)$$

where $\bar{s}(k)$ represents the mean $s(i)$ over all the dataset for k clusters.

Another heuristic is the graphical method called “knee” method, or “elbow” method. This consists in plotting the evaluation criteria for various values of K and estimate the optimal value of K as the point where successive decreases (increases) in the measure become noticeably smaller [26].

Extrinsic Evaluation

Extrinsic Evaluation is a measure of the quality of clustering in presence of information on the elements to be clustered. Some basic measures are precision and recall. Precision is the fraction of the relevant elements among those retrieved, while recall is the fraction of relevant elements that were retrieved.

The F-measure is a value in the range $[0, 1]$, defined as the harmonic mean of precision and recall:

$$F = 2\frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

F-measure weights precision and recall equally, and is positively related to a better clustering of the relevant elements.

Rand Index

Another measure which is useful to evaluate clustering is the Rand Index [33]. It can be used as either External Evaluation criterion or as a comparison between different clusterings.

This measure returns a number in the interval $[0, 1]$ indicating how similar two clusterings are the closer the measure is to 1. It is closely related to accuracy.

Given a set of N points $X = \{X_1, \dots, X_N\}$ and two partitions of it, $Y^{(1)} = \{Y_1^{(1)}, \dots, Y_N^{(1)}\}$ into k_1 partitions and $Y^{(2)} = \{Y_1^{(2)}, \dots, Y_N^{(2)}\}$ into k_2 partitions a definition of the Rand index is:

$$R(Y^{(1)}, Y^{(2)}) = \frac{\sum_{i<j}^{N} \gamma_{ij}}{\binom{2}{n}}$$

where γ_{ij} is 1 if there exist k and k' such that either both X_i and X_j are in both $Y_k^{(1)}$ and $Y_{k'}^{(2)}$, or X_i is in both $Y(1)_k$ and $Y(2)_{k'}$ while X_i is in neither $Y_k^{(1)}$ or $Y_{k'}^{(2)}$; 0 otherwise.

In other words $\sum_{i<j}^{N} \gamma_{ij} = a + b$, where a counts the number of elements of X which are in the same subset in $Y(1)$ and in the same subset in $Y(2)$, and b counts the number of elements of X which are in a different subset in $Y(1)$ and in a different subset in $Y(2)$.

Bibliography

[1] La storia di gladio, ilpost.it, 10 2020. URL https://www.ilpost.it/2020/10/24/gladio-stay-behind/. accessed: 23/02/2021.

[2] Alessandro Bessi. Personality traits and echo chambers on facebook. *Computers in Human Behavior*, 65:319–324, 2016. ISSN 0747-5632. doi: https://doi.org/10.1016/j.chb.2016.08.016. URL https://www.sciencedirect.com/science/article/pii/S0747563216305817.

[3] Cornelia Betsch and Robert Böhm. Detrimental effects of introducing partial compulsory vaccination: experimental evidence. *European Journal of Public Health*, 26(3):378–381, 08 2015. ISSN 1101-1262. doi: 10.1093/eurpub/ckv154. URL https://doi.org/10.1093/eurpub/ckv154.

[4] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, Oct 2008. ISSN 1742-5468. doi: 10.1088/1742-5468/2008/10/p10008. URL http://dx.doi.org/10.1088/1742-5468/2008/10/P10008.

[5] Michele Bocci and Elvira Naselli. Morbillo, muore una bambina di 9 anni a roma. *La Relubbpica*, 28-06-2017. URL https://www.repubblica.it/salute/prevenzione/2017/06/28/news/morbillo_muore_una_bambina_di_9_anni_a_roma-169388590/. accessed: 09/04/2021.

[6] David A. Broniatowski, Amelia M. Jamison, SiHua Qi, Lulwah AlKulaib, Tao Chen, Adrian Benton, Sandra C. Quinn, and Mark Dredze. Weaponized health communication: Twitter bots and russian trolls amplify the vaccine debate. *American Journal of Public Health*, 108 (10):1378–1384, 2018. doi: 10.2105/AJPH.2018.304567. URL https://doi.org/10.2105/AJPH.2018.304567. PMID: 30138075.

[7] Emanuele Brugnoli, Matteo Cinelli, Fabiana Zollo, Walter Quattrociocchi, and Antonio Scala. Lexical convergence and collective identities on facebook, 2020.

[8] Joanna M Burkhardt. History of fake news. *Library Technology Reports*, 53(8):5–9, 2017.

[9] Maicolengel Butac. The black list, la lista che vanta innumerevoli tentativi di imitazione! URL https://www.butac.it/the-black-list/. accessed: 5/11/2020.

[10] Guido Caldarelli, Michela Del Vicario, Walter Quattrociocchi, Antonio Scala, and Fabiana Zollo. Mapping social dynamics on facebook: The brexit debate. *Social Networks*, 50:6–16, 2017. ISSN 0378-8733. doi: https://doi.org/10.1016/j.socnet.2017.02.002. URL https://www.sciencedirect.com/science/article/pii/S0378873316304166.

[11] Matteo Cinelli, Mauro Conti, Livio Finos, Francesco Grisolia, Petra Kralj Novak, Antonio Peruzzi, Maurizio Tesconi, Fabiana Zollo, and Walter Quattrociocchi. (mis)information operations: An integrated perspective, 2019.

[12] Matteo Cinelli, Emanuele Brugnoli, Ana Lucia Schmidt, Fabiana Zollo, Walter Quattrociocchi, and Antonio Scala. Selective exposure shapes the facebook news diet. *PLOS ONE*, 15(3):1–17, 03 2020. doi: 10.1371/journal.pone.0229129. URL https://doi.org/10.1371/journal.pone.0229129.

[13] Matteo Cinelli, Walter Quattrociocchi, Alessandro Galeazzi, Carlo Michele Valensise, Emanuele Brugnoli, Ana Lucia Schmidt, Paola Zola, Fabiana Zollo, and Antonio Scala. The covid-19 social media infodemic. *Scientific Reports*, 10(1), Oct 2020. ISSN 2045-2322. doi: 10.1038/s41598-020-73510-5. URL http://dx.doi.org/10.1038/s41598-020-73510-5.

[14] Matteo Cinelli, Gianmarco De Francisci Morales, Alessandro Galeazzi, Walter Quattrociocchi, and Michele Starnini. The echo chamber effect on social media. *Proceedings of the National Academy of Sciences*, 118(9), 2021. ISSN 0027-8424. doi: 10.1073/pnas.2023301118. URL `https://www.pnas.org/content/118/9/e2023301118`.

[15] Aaron Clauset, M. E. J. Newman, and Cristopher Moore. Finding community structure in very large networks. *Physical Review E*, 70(6), Dec 2004. ISSN 1550-2376. doi: 10.1103/physreve.70.066111. URL `http://dx.doi.org/10.1103/PhysRevE.70.066111`.

[16] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, 3rd Edition.* MIT Press, 2009. ISBN 978-0-262-03384-8.

[17] Alessandro Cossard, Gianmarco De Francisci Morales, Kyriaki Kalimeri, Yelena Mejova, Daniela Paolotti, and Michele Starnini. Falling into the echo chamber: the italian vaccination debate on twitter, 2020.

[18] Michela Del Vicario, Alessandro Bessi, Fabiana Zollo, Fabio Petroni, Antonio Scala, Guido Caldarelli, H. Eugene Stanley, and Walter Quattrociocchi. The spreading of misinformation online. *Proceedings of the National Academy of Sciences*, 113(3):554–559, 2016. ISSN 0027-8424. doi: 10.1073/pnas.1517441113. URL `https://www.pnas.org/content/113/3/554`.

[19] Andrew Guess and Benjamin Lyons. *Misinformation, Disinformation, and Online Propaganda*, pages 10–33. 08 2020. ISBN 9781108835558. doi: 10.1017/9781108890960.003.

[20] Kurt Hornik, Ingo Feinerer, Martin Kober, and Christian Buchta. Spherical k-means clustering. *Journal of Statistical Software, Articles*, 50(10):1–22, 2012. ISSN 1548-7660. doi: 10.18637/jss.v050.i10. URL `https://www.jstatsoft.org/v050/i10`.

[21] S. C. Johnson. Hierarchical clustering schemes. *Psychometrika*, 32:241–254, 1967.

[22] Howell L. Digital wildfires in a hyperconnected world. wef report 2013., 2013. URL `reports.weforum.org/global-risks-2013/risk-case-1/digital-wildfires-in-a-hyperconnected-world`. accessed: 2/2/2021.

[23] William Langer. *Bismarck as a Dramatist*, pages 128–146. Harvard University Press, 2013. doi: doi:10.4159/harvard.9780674493308.c6. URL `https://doi.org/10.4159/harvard.9780674493308.c6`.

[24] Cathey Libby and Allison Pecorin. Mcconnell says trump 'provoked' capitol assault, 'fed lies' to mob, abcnews.go.com. URL `https://abcnews.go.com/Politics/mcconnell-trump-provoked-capitol-assault-fed-lies-mob/story?id=75349374`. accessed: 02/03/2021.

[25] Laura Locke. The future of facebook. *Time*, 2007-07-17. URL `http://content.time.com/time/business/article/0,8599,1644040,00.html`. accessed: 9/4/2021.

[26] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval.* Cambridge University Press, Cambridge, UK, 2008. ISBN 978-0-521-86571-5. URL `http://nlp.stanford.edu/IR-book/information-retrieval-book.html`.

[27] Fionn Murtagh and Pierre Legendre. Ward's hierarchical agglomerative clustering method: Which algorithms implement ward's criterion? *Journal of Classification*, 31:274–295, 10 2014. doi: 10.1007/s00357-014-9161-z.

[28] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. *Physical Review E*, 74(3), Sep 2006. ISSN 1550-2376. doi: 10.1103/physreve.74.036104. URL `http://dx.doi.org/10.1103/PhysRevE.74.036104`.

[29] World Health Organization. Novel coronavirus (2019-ncov): situation report, 13. Technical documents, World Health Organization, 2020-02-02.

[30] Catharine I. Paules, Hilary D. Marston, and Anthony S. Fauci. Measles in 2019 - going backward. *New England Journal of Medicine*, 380(23):2185–2187, June 2019. ISSN 0028-4793. doi: 10.1056/NEJMp1905099. Funding Information: From the Department of Infectious Diseases, Penn State University College of Medicine, Milton S. Hershey Medical Center, Hershey, PA (C.I.P.); and the Office of the Director, National Institute of Allergy and Infectious Diseases, National Institutes of Health, Bethesda, MD (H.D.M., A.S.F.).

[31] Pascal Pons and Matthieu Latapy. Computing communities in large networks using random walks. In pInar Yolum, Tunga Güngör, Fikret Gürgen, and Can Özturan, editors, *Computer and Information Sciences - ISCIS 2005*, pages 284–293, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg. ISBN 978-3-540-32085-2.

[32] Usha Nandini Raghavan, Réka Albert, and Soundar Kumara. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E*, 76(3), Sep 2007. ISSN 1550-2376. doi: 10.1103/physreve.76.036106. URL http://dx.doi.org/10.1103/PhysRevE.76.036106.

[33] William M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971. ISSN 01621459. URL http://www.jstor.org/stable/2284239.

[34] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, 20:53–65, 1987. ISSN 0377-0427. doi: https://doi.org/10.1016/0377-0427(87)90125-7. URL https://www.sciencedirect.com/science/article/pii/0377042787901257.

[35] Ana Schmidt, Fabiana Zollo, Michela Del Vicario, Alessandro Bessi, Antonio Scala, Guido Caldarelli, H. Stanley, and Walter Quattrociocchi. Anatomy of news consumption on facebook. *Proceedings of the National Academy of Sciences*, 114, 03 2017. doi: 10.1073/pnas.1617052114.

[36] Ana Schmidt, Fabiana Zollo, Antonio Scala, Cornelia Betsch, and Walter Quattrociocchi. Polarization of the vaccination debate on facebook. *Vaccine*, 36, 01 2018. doi: 10.1016/j.vaccine.2018.05.040.

[37] Ana Lucía Schmidt, Fabiana Zollo, Antonio Scala, and Walter Quattrociocchi. Polarization rank: A study on european news consumption on facebook, 2018.

[38] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *Science*, 359(6380):1146–1151, 2018. ISSN 0036-8075. doi: 10.1126/science.aap9559. URL https://science.sciencemag.org/content/359/6380/1146.

[39] Vv.Aa. The black list, la lista nera del web, . URL https://www.bufale.net/the-black-list-la-lista-nera-del-web/. accessed: 5/11/2020.

[40] Vv.Aa. Centro di monitoraggio della disinformazione sul coronavirus, . URL https://www.bufale.net/the-black-list-la-lista-nera-del-web/. accessed: 5/11/2020.

[41] Steven Lloyd Wilson and Charles Wiysonge. Social media and vaccine hesitancy. *BMJ Global Health*, 5(10), 2020. doi: 10.1136/bmjgh-2020-004206. URL https://gh.bmj.com/content/5/10/e004206.

[42] Jennifer Zipprich, Kathlen Winter, Jill Hacker, Dongxiang Xia, James Watt, and Kathleen Harriman. Measles outbreak - california, december 2014-february 2015. *MMWR. Morbidity and mortality weekly report*, 64:153–4, 02 2015.

[43] Fabiana Zollo, Alessandro Bessi, Michela Del Vicario, Antonio Scala, Guido Caldarelli, Louis Shekhtman, Shlomo Havlin, and Walter Quattrociocchi. Debunking in a world of tribes, 2015.