Modeling Users Preferences in Online Social Networks

Lorena Recalde

DOCTORAL THESIS UPF / 2018

THESIS SUPERVISORS

Prof. Dr. Ricardo Baeza-Yates
Prof. Dr. David Nettleton

Departament of Information and Communication Technologies

upf. Universitat
Pompeu Fabra
Barcelona

To my father, because of his love for life,
to my mother, because of her daily courage,
to my sisters, who are my journey lights,
to Jhoel, because he is the happiness for us,
to Alex, because he has sweetened my heart.

Esta tesis está dedicada a mi familia...

Acknowledgement

It has been a big privilege to learn from Ricardo Baeza-Yates, my supervisor. I want to thank you for opening the ‘research doors’ for me and giving me this opportunity that has changed my life. You believed in me and encouraged me to keep working (I hope never disappoint you). I want to thank my co-supervisor, David Nettleton, for your patience and advice and for the valuable hours that you spent reading my work or discussing about methods and results with me, I learned many things while working with you, thanks. I am very thankful to Ludovico Boratto. When we started working together, I saw research with new eyes because you taught me lots of things, not only about writing papers but also to keep confident and do not fall into desperation. My favorite times doing research were while we were working together and I could say that your mentorship has been as important as your friendship. I met Carlos Castillo in the last months of the PhD. I want to thank you because you took some time to know about my work, gave me great ideas to keep working on my topic and you are a scientist who inspires me hard work. I also want to thank Lydia, from the DTIC Secretary, for being very helpful, supportive and understanding with all the Ph.D. students. Thank you for helping me to solve all kinds of problems!

Doing the PhD allowed me to know great people who became my friends and have been my ‘academic’ family. Dear Ana, how many moments have we shared together? You are a woman with a huge strength, brightness and kindness that motivate young women around you, thank you for being an impact in my life. You are the one who has seen me struggling for years and unconditionally supported me, thank you. I do not see a better company than Maria and Diana in those uncountable hours of work. Thank you for all the laughs, talks, coffees and time we have shared together. In the DTIC there are people who I shared the office the first years, you were always nice and inspired me with your work and professionalism, thank you Toni and Gabriel, I keep nice memories of our time together. Thank you Luz, there were few times I went out with you, but in those times you gave me the best advice about writing the thesis and told me that I had to enjoy it; well, I did it...

Mezli and Leti, it was very great to work with you in the UPF course 'From draft to submission'. You took time to read my last paper and gave me feedback that improved my work. We enjoyed together the writing experience of articles which made us become friends. Lucy, we have lived together one year and I would like to say that you have been a great company, you just arrived when I needed someone more open and generous to share with. Thank you for making my days less lonely. Myriam, because of you I could completely focus on the thesis these years without worrying about problems at home, you really made my life in Barcelona very comfortable, thank you. Moni and Lesly, somehow we were together in this since 2012, thank you.

I spent three months as an intern in the Information Systems Department at the University of Fribourg, in Switzerland. The Professor Andreas Meier welcomed me in his research group and I could work in the article that is explained in Chapter 4 of the thesis. Thank you for being part of my formation as a researcher. During that time I had the opportunity to collaborate with Luis and Aigul. I consider you both as friends since we went to the amusement park in Vienna, RecSys 2015. Dear Aigul, our lunch times, coffees, supportive talks and trips to conferences together were very valuable for me, thank you.

Carmen, Gaby and Jonathan, it was a pleasure to collaborate with you. Thank you for giving your best and working with enthusiasm in our journal paper.

There are three families who let me be part of their homes during my studies. Paty and Marco, you will be always my second parents, my family and I are very thankful for the things you did for me. Luis and

Pitu, thank you for the time we were together, your support, our dinners and talks; you are very dear friends for me. Vero, Tiffi and Diego, my best moments in Barcelona and Brussels have been with you. Thank you for considering me as a sister and being so generous with your love.

Quiero agradecer a mis suegros, Angelita y Edgar, quienes han sido muy comprensivos, pacientes y solidarios. ¡Gracias por su apoyo y por cuidar de mí como a una hija más! A mis dos abuelas, Ana y Mercedes, muchas gracias. Sus abrazos siempre han sido los más cargados de cariño, tal vez no ha sido fácil ver que me fui "sola", y "tan lejos", pero ustedes educaron a sus hijos "solas", y ese ha sido el mayor ejemplo que tengo de mujeres valientes. A mi familia entera, siempre pendientes, perdón si no los menciono uno a uno, gracias por el apoyo constante y por no dejar solos a mis padres desde diciembre 2016. Gracias a mis amigas y amigos en Quito, sus mensajes me han acompañado en estos años.

Lourdes, Luis, Cristy, Carito y Jhoel, ustedes son el mayor impulso que tengo en la vida. Gracias por todo su amor incondicional. No cambiaría ni un solo minuto de nuestra vida juntos por nada, porque son mi mayor tesoro. Dios les bendiga y bendiga a nuestra familia.

Gracias Ale, has sido el pilar donde me sostengo, el refugio donde encuentro paz, el amigo que me da de su fuerza y el compañero que me anima a seguir sin mirar atrás. Gracias a Dios por tu vida y gracias a ti por tu amor.

Si decidí hacer el doctorado fue por mi país, por quienes fueron mis alumnos, por la gente que trabaja incansablemente para que el Ecuador sea un lugar mejor, por quienes guardan la esperanza de que un día termine la corrupción, por quienes son abiertos y dan buen trato a los immigrantes, por quienes no ven raza, género o religión. Gracias por ser una inspiración, espero no defraudarles.

Abstract

The objective of this thesis is to develop new and diverse methods to model the preferences of users in Online Social Networks. The proposed methods are intended to be applied in areas of research such as *personalization* or *recommendation* of items and the *detection of groups of users* who have similar preferences. These methods can be grouped into two types: i) methods based on text analysis techniques (Part I, Chapters 3 to 5) and ii) methods based on graph theory (Part II, Chapters 6 and 7).

With the methods proposed in Part I it is possible to determine the level of interest of users on topics that are shared on microblogging platforms. We have taken as a case study the digital participation of tweeters in politics. For example, we propose an approach that allows to quantify the degree of interest of users regarding political topics. Similarly, another of the proposals allows to define the political alignment of users. Our research shows that to model unstructured and short texts such as tweets, the techniques that implement word embeddings are highly efficient. Therefore, users' preference models based on the content extracted from their posts can represent their topics of interest in the short and medium term.

The methods proposed in Part II aim at defining a role for users in social networks, whether as 'creators' or content generators and 'distributors' or content 'consumers'. We have proposed a method where users with similar interests but with different roles, are grouped in the same community so that new content spreads more quickly. Unlike the approaches in Part I, these methods are based on connections (whether explicit or not) between users and not on content that has been previously shared. We end applying our methods to event-based communities to show that they extend to other social media data.

Resum

L'objectiu d'aquesta tesi és desenvolupar nous i diversos mètodes per modelar les preferències dels usuaris a les Xarxes Socials Online. Els mètodes proposats tenen com a finalitat ser aplicats en àrees de recerca com la Personalització Recomanació d'ítems i la Detecció de Grups d'Usuaris amb gustos similars. Aquests mètodes poden ser agrupats en dos tipus: i) mètodes basats en tècniques d'anàlisi de textos (Part I, Capítols del 3 al 5) i ii) mètodes basats en teoria de grafs (Part II, Capítols 6 i 7).

Amb els mètodes plantejats a la Part I és possible determinar el nivell d'interès dels usuaris en temes que són compartits en plataformes de microblogging. Hem pres com a cas d'estudi la participació digital de 'tweeters' a la política. Per exemple, plantegem un enfocament que ens permet quantificar el grau d'interès dels usuaris pel que fa a temes polítics. De la mateixa manera, una altra de les propostes permet definir l'orientació política dels usuaris. La nostra investigació demostra que per modelar textos desestructurats i curts, com són els tweets, les tècniques que implementen *word embeddings* són altament eficients. Per tant, els models de preferències dels usuaris basats en el contingut extret dels seus posts poden representar temes d'interès a curt i mitjà termini.

Els mètodes proposats a la Part II busquen definir un paper pels usuaris de les Xarxes Socials, ja sigui com a 'creadors' o generadors de contingut i 'distribuïdors' o 'consumidors' de contingut. Hem plantejat un mètode on usuaris amb interessos similars però amb diferent rols són agrupats en una mateixa comunitat, de manera que els nous continguts es propaguen més ràpidament. A diferència dels anteriors, aquests mètodes estan basats en les connexions (ja siguin explícites o no) entre usuaris i no en el contingut que ha estat compartit.

Resumen

El objetivo de esta tesis es desarrollar nuevos y diversos métodos para modelar las preferencias de los usuarios en las Redes Sociales Online. Los métodos propuestos tienen como finalidad ser aplicados en áreas de investigación como la *personalización o recomendación* de ítems y la *detección de grupos de usuarios* con gustos similares. Dichos métodos pueden ser agrupados en dos tipos: i) métodos basados en técnicas de análisis de texto (Parte I, Capítulos del 3 al 5) y ii) métodos basados en teoría de grafos (Parte II, Capítulos 6 y 7).

Con los métodos planteados en la Parte I es posible determinar el nivel de interés de los usuarios en temas que son compartidos en plataformas de microblogging. Hemos tomado como caso de estudio la participación digital de 'tweeters' en la política. Por ejemplo, planteamos un enfoque que nos permite cuantificar el grado de interés de los usuarios en cuanto a temas políticos. De igual forma, otra de las propuestas permite definir la alineación política de los usuarios. Nuestra investigación demuestra que para modelar texto desestructurado y corto, como son los tweets, las técnicas que implementan *word embeddings* son altamente eficientes. Por consiguiente, modelos de preferencias de los usuarios basados en el contenido extraído de sus posts pueden representar los temas de interés a corto y mediano plazo.

Los métodos propuestos en la Parte II buscan definir un rol para los usuarios en Redes Sociales, ya sea como 'creadores' o generadores de contenido y 'distribuidores' o 'consumidores' de contenido. Hemos planteado un método donde usuarios con intereses similares pero con distinto rol, son agrupados en una misma comunidad de forma que nuevo contenido se propague más rápidamente. A diferencia de los anteriores, estos métodos están basados en las conexiones (ya sean explícitas o no) entre usuarios y no en contenido que ha sido compartido previamente.

Contents

List of Figures

List of Tables

Chapter 1

INTRODUCTION

The need for better personalization in online applications has become increasingly more important due to the user generated content that grows in volume every day. The more users the social network has, the more content is generated, which may be seen as a factor that determines the platform success. However, the adverse effect is the presence of information overload. To prevent this, the implementation of algorithms in order to understand the users' topics of preference makes it possible to provide them with information that matches their interests and filters out the rest; thus, the emergence of the so called Recommender Systems.

1.1. Motivation

For people, creating mental models of their relatives or friends is not very hard. Actually, we would be able to say with certain precision if our parents are going to like a movie or not; and we could do this because we know their likes. The same idea requires to be replicated in a system that aims at personalization. Thus, the system has to create models of the users by employing their available data: their social ties or connections, explicit ratings (likes, reviews, star evaluations, among others), content that they share, contextual signals, implicit feedback (clicks, searches, bookmarks, etc.), and so on [1].

Social Networking Systems such as discussion forums, blogs, tagging applications and professional networks facilitate the exchange of information between users and establish relevant and direct connections. Users are not static entities for information consumption, instead they generate and share content, collaborate in communities, provide feedback, add content and create personal profiles. Virtual social connections and involved interactions have shown to be useful for creating new ways of extracting and modeling user's interests which are subsequently employed in Recommender Systems.

Mining the Social Web has favored new forms of recommendations. For example, diverse social entities like hashtags, people to follow, trends, groups to subscribe to, content, events to attend, multimedia, URLs to be added in posts as well as emoticons, etc., are suggested for the users. As a consequence, the expectations of the users have risen regarding the ability of such systems to present them with relevant, consistent and up-to-date information. Then, the need of implementing systems that build models to understand the users' interests remains.

Social media users profiling has become of interest in many fields: marketing, communication, job recruitment, e-democracy, among others. This thesis covers different approaches to model the users preferences and our methods can be applyied in diverse domains. Specifically, some of our approaches contribute in politics, which is one of the fields most affected by the use of OSNs.

1.2. Contributions

The specific contributions concerning the proposed strategies to model users preferences are:

- A method that combines 'word embeddings' and a 'probabilistic clustering approach' in order to define a multidimensional (or multi-topic) user model (Chapter 3).

- A method that allows to quantify the extent to which a user is interested in politics (Chapter 4). The generic application of this approach in fields like sports, culture, and science, among others, would allow to measure the level of interest of users in each of the mentioned domains.

- A set of strategies to specify the political ideology of users and determine which of them make wrong use of political-related hashtags (Chapter 5). The generic application of this approach in topics that generate debate where there are user groups with opposing views (such as life imprisonment supporters/opponents, immigrants' rights supporters/opponents, among others) would allow their detection, as well as the identification of the groups' characterizing hashtags.

- A model that detects topic-dependent communities in which content creators and consumers are linked in a way that facilitates information sharing. The approach is shown to be efficient in both Twitter and Meetup data (Chapters 6 and 7).

This work has produced the following publications:

- **Lorena Recalde**, Jonathan Mendieta, Ludovico Boratto, Luis Terán, Carmen Vaca, Gabriela Baquerizo. "Who You Should Not Follow: Extracting Word Embeddings from Tweets to Identify Groups of Interest and Hijackers in Demonstrations", 2017, in IEEE Transactions on Emerging Topics in Computing, Issue: 99 (Chapter 5).

- **Lorena Recalde**, Aigul Kaskina. "Who is suitable to be followed back when you are a Twitter interested in Politics?", in Proceedings of the 18th Annual International Conference on Digital Government Research (dg.o '17), Jun 7-9, 2017. ACM, New York, NY, USA, 94-99 (Chapter 4).

- **Lorena Recalde**, David F. Nettleton, Ricardo Baeza-Yates, Ludovico Boratto. "Detection of Trending Topic Communities: Bridging Content Creators and Distributors", in Proceedings of the

28th ACM Conference on Hypertext and Social Media (HT ’17), July 04-07, 2017, Prague, Czech Republic (Chapter 6).

- **Lorena Recalde**, Ricardo Baeza-Yates. “What kind of content are you prone to tweet? Multi-topic Preference Model for Tweeters” in Workshop on Social Aspects in Personalization and Search, collocated with ECIR 2018, Apr 26, Grenoble, France (Chapter 3).

1.3. Organization

The thesis is structured in two main thematic parts: Part I, which contains Chapters 3 to 5, presents three approaches to model users’ interests which are based on mining their posts, and Part II, containing Chapters 6 and 7, proposes an approach based on the detection of topic-dependent communities by identifying meaningful connections in the users’ social graph.

In detail, the rest of this thesis is organized as follows.

Firstly, in Chapter 2, we describe how the relevance of Online Social Networks (OSNs) to fields such as Web Science and Social Computing has motivated our research. Then, we introduce concepts such as *Personalization*, *Recommender Systems*, *Social Web Mining*, *Social Network Analysis* and *Groups Detection*.

In Part I (Chapters 3 to 5), we consider text modeling as an important strategy for extracting user’s interests and applying word embeddings to obtain a vector representation of tweets and/or users. First, with the aim of quantifying the extent of a topic participation in the user’s profile, we employ word embeddings together with an unsupervised soft clustering method, Mixture of Gaussians (Chapter 3). We demonstrate that the proposed method, named *MUM* (Multi-topic User Model), is more accurate than modeling tweets with the general TF-IDF model and supervised machine learning algorithms.

Next, as word embeddings have shown to work well for short text modeling (Chapter 3), we use *word2vec* to model tweeters by aggregating their time-line posts represented as vectors. After some experiments, we reinforce the social-related conception that says ‘we tend to be friends with similar people’ [2]. This validation consists of an approach to *i)* define the users’ political profile by quantifying their degree of interest in politics and *ii)* compare this measure with their friends’ political interest level (Chapter 4). Moreover, we demonstrate that users subscribe to lists (or groups) that fit in with their preferred information topics.

While in Chapter 4 we proposed a method to calculate the extent of interest in politics of any Twitter user, in Chapter 5 we propose a method to identify the political stance of tweeters (detected as active users during protests) and extract the hashtag hijackers from the recognized political interest groups. The aim of the approach is to recommend the groups not to follow, in this case represented by malicious users who introduce noise and confusion in political-oriented communities.

The aforementioned chapters propose three methods to model users according to the content they post. In contrast, in Part II, we study the users’ social network structure to facilitate the detection of topic/category communities in which creators of content and distributors/consumers co-exist. Chapter 6 proposes an approach to model short-term topic preferences without studying the entire users’ time-line and doing text mining as in Part I. Contrary, we consider social ties (Following/Retweeting) and the trending topics that are promoted thanks to two kinds of tweeters: the creators and the distributors. That is to say, those who create relevant content give meaning to a given hashtag, noun or noun phrase for a group of interest. Additionally, with the support of their corresponding retweeters or content distributors group, the cycle of topic propagation is completed. An approach based on the detection of trending topic communities where creators and distributors are put together is presented.

The approach in Chapter 6 is extended in Chapter 7 to an Event-Based Social Network (EBSN). In such way, we demonstrate its general applicability. Broadly speaking, in EBSNs the explicit links among users are limited to the membership of a user to a given meetup group. Then we infer the relationships between *organizers* and *members* to generate useful connections and consequently favor the visibility of new events and groups.

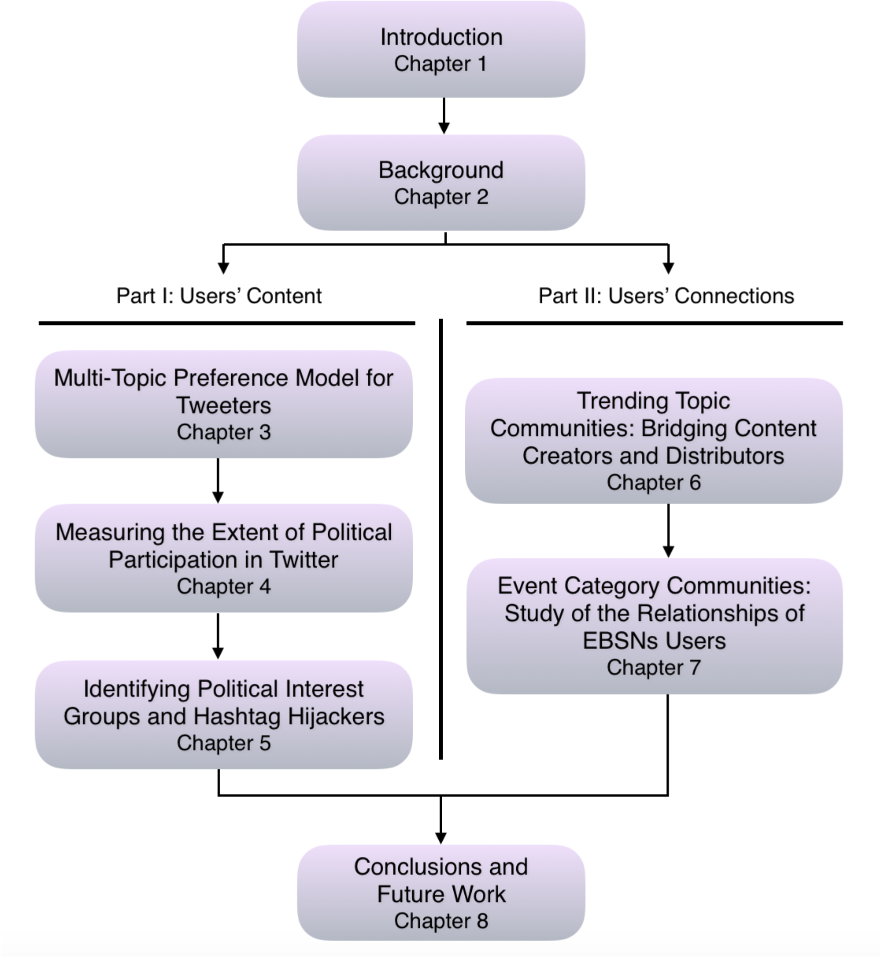

Introduction
Chapter 1
Background
Chapter 2
Part I: Users' Content
Part II: Users' Connections
Multi-Topic Preference Model for Tweeters
Chapter 3
Measuring the Extent of Political Participation in Twitter
Chapter 4
Identifying Political Interest Groups and Hashtag Hijackers
Chapter 5
Trending Topic Communities: Bridging Content Creators and Distributors
Chapter 6
Event Category Communities: Study of the Relationships of EBSNs Users
Chapter 7
Conclusions and Future Work
Chapter 8

Figure 1.1: Conceptual roadmap of the thesis.

The last chapter of the dissertation (Chapter 8) summarizes the work done and details how the aims of this thesis were achieved. Furthermore, future research directions and final remarks are also outlined.

Figure 1.1 illustrates the conceptual roadmap of the thesis.

Chapter 2

BACKGROUND

In this chapter we commence with an overview of Online Social Networks (OSNs), which is followed by how the user experience can be personalized in the context of recommender systems. Then, the detection of groups and information flows in OSNs is considered and how this can leveraged to recommend users and topics to follow to the members.

2.1. Online Social Networking Platforms

Technology is intended to augment diverse human abilities. For instance, since the emergence of social networking sites, the capabilities like remembering, thinking and reasoning have evolved to support social cognition [3]. In other words, users of such technology are aware of others, and have different social virtual ways to interact with each other. These online spaces provide functions to facilitate people coming together with others to satisfy the need of companionship, exchange information and other resources, learn, play, or converse. Concerning the OSNs' configurations, small knit groups or sites with millions of users may be found. From blogs to wikis, the common feature is (technology-mediated) ongoing interactions among people over time [4].

Research based on OSNs' data mainly model, mine and understand socially constructed knowledge structures and social information networks to identify, for example, expertise, information propagation, decision making, and collective intelligence, among others.

To develop and evaluate the methods proposed in this thesis we worked with Twitter and Meetup datasets. Next we present some definitions to contrast them.

2.1.1. Twitter

Twitter is a microblogging OSN whose activity is depicted by tweets, retweets, replies, likes and shares, and whose structure is defined by *follower* and *followee* unidirectional relationships. Tweets are short messages with a text size up to 140 characters. A tweet may contain text, pictures, video, as well as mentions to other Twitter users, URLs, hashtags and locations.

Users follow their friends, celebrities, news media accounts, or anyone/anything else they are interested in. Therefore, the users can access a personalized and filtered timeline of their *followees'* tweets. They can retweet them and promote the information diffusion over the network. Twitter users are able to explore domains and communities of interest and benefit from being part of the online social network.

2.1.2. Meetup

One kind of OSN are the so-called Event-Based Social Networks (EBSNs). These platforms offer the users the possibility to create, manage, share, schedule and join upcoming events. *Meetup.com* is one of the largest EBSNs available nowadays with thousands of meetup groups around the world. In the last years the dynamics generated in Meetup have attracted the attention researchers in the field of Recommender Systems. Specifically, the problems to be solved are related to event recommendation becaause they are short-lived, planned in the future and as they are always 'new' there is no trace of historical attendance; then, classic recommendation strategies are hardly applicable [5].

2.2. Personalization and Recommender Systems

On the one hand, the wide and quick spread of the use of OSNs has shown the need of users of not only to establish social connections, but also having access to information generated by others. On the other hand, the interaction of users with a huge number of people may affect the access to *relevant* information. Thus, mechanisms of personalization become necessary.

Recommender systems emerged with the aim of predicting users' interests, which is done by building the user preferences model and finding the item or ranked list of items that best fits their needs. Therefore, the level of personalization increases when the recommender system knows more about the user.

The term *item* is a general word used to make reference to the object that the recommender system suggests. Accordingly, an item to recommend would be a singer, a movie, a restaurant, a Twitter user to follow, a Meetup event to attend or a Facebook friend to add. However, the recommendation might be not only a ranked list of independent items, from which the user selects, buys or adopts any of the items presented. It may be an ordered set of elements, where one item recommendation signifies some elements provided in a specific order, or a bunch of items put together having the notion of *better together*, so a bundle of two or more objects conforms the item recommendation [6].

The nature of the recommendation or the type of item recommended is usually determined by the system *domain*. The domain guides the design of the recommender system because the approaches and techniques to implement may differ depending on whether the system recommends a recipe, a medical treatment or a car to rent.

The approaches applied in recommender systems have evolved since the mid-1990's. Many improvements to the algorithms and techniques have been published as a result of academia and industry research.

The main approaches are:

- Collaborative Filtering. The algorithms use historical rating information to compare how similar the users' preferences are. The search of neighbours of the current user allows to recommend him/her items with high ratings provided by his/her peers.

- Content-based techniques. The recommender bases its suggestions on the degree of high previous acceptance of items which have the same features or attributes as new items which have not been seen previously by the user. Therefore, because of their similarity they may be recommended.

- Knowledge-based techniques. These system contain knowledge bases concerning users and items. Most of the time the needs are elicited through conversational interactions between the user and a recommender assistant until discovering the item that has the desired characteristics.

The approaches mentioned have different variations and may be combined as a hybrid recommender system [7] in order to minimize their individual drawbacks. In recent years Context-Aware, Social-Based and Trust-Aware Recommenders have also emerged to present paradigms that the recommender systems developers may analyze to find which of the approaches best suits the requirements of the system.

Decisions about the design of the recommender have to be made after knowing which items are to be recommended, or more generally, once the domain is defined. However, knowing which target to recommend to has the same importance. The target might be a *single user* or a *group of people*. Thus, considering the type of target user, the recommenders are classified in *Recommender Systems* and *Group Recommender Systems* respectively. This classification has been proposed since modeling the interests of a person is not the same as modeling the interests of a family, a group of friends or a group of people who are sharing a room.

In the light of the above overview about the main function of a recommender system, we could say that the popularity of these systems has increased thanks to their usefulness. However, gathering and processing users' data introduce privacy risks that should also be taken into account when designing a recommender system. Indeed, commonly, users are not sufficiently aware of what kind of their data is collected, how securely it is stored, or if it is shared with third parties. To address privacy concerns in recommender systems, Jeckmans *et al.* discuss the associated risks to data privacy and relevant research areas for privacy-protection techniques and their applicability in [8].

Understanding Users' Preferences

Some information retrieval tasks (retrieving, filtering and recommending) need to continuously refine user profiles and one support for this are implicit feedback techniques to model users' interests [9]. The users could explicitly provide feedback by answering questions, choosing and rating items, or annotating emotions and tags, but those additional steps require extra users' time. Implicit feedback may be implemented in an unobtrusive way by obtaining information from the users interactions with Web applications and user-virtual friends (user preferences).

Reading time, using *find* while surfing in the web, formulating queries, bookmarking, copping and pasting, saving, selecting, subscribing, emailing, printing are all actions that contribute with implicit feedback. It could be valuable to collect implicit feedback information about users interests but the measures are not all useful in every case and sometimes they need to be combined to extract appropriate information, thus making it a challenge in practical terms.

Regarding users' data collection to infer their interests, there is an inherent trade-off between privacy and accuracy. For example, randomization techniques increase privacy by lowering accuracy. Users need privacy guarantee and have to be asked about data disclosure, how it is going to be used and for how long it is going to be saved.

User preferences analysis in OSNs is one of the areas of greatest interest to research communities. So much so, Social Network Analysis and techniques of graph theory, as well as Web Mining are employed in multidisciplinary studies, which involve psychologists, sociologists, business managers, mathematicians, data scientists, among others.

Social Web Mining

Web data mining aims at discovering the inherent relationships among Web data, which are expressed in the forms of textual, linkage or usage information. By analyzing the features of the Social Web with the use of data mining techniques, we may extract users behavior, personality and likes, know who interacts with whom, the topics they talk about, and accordingly, present them with personalized content. The objective of any data mining process is to build and efficient predictive or descriptive model of a large amount of data that explains it and can also be generalize to new data. Popular social networks such as Twitter, Facebook, LinkedIn, Google+, etc. offer an API to extract data. Then, techniques for data analysis can be employed to explore, preprocess, organize, structure, mine and visualize information.

Social Network Analysis

Social network analysis (SNA) focuses on the structure of ties within a set of social actors (persons, groups, organizations, and nations, or the products of human activity or cognition such as web sites, semantic concepts, and so on) to map and measure relationships and flows.

Mainly, related research has considered: *i)* the processes that link organizations, associations, cultural communities, social movements, and other social forms; *ii)* the impact of ties on the patterns of homophily; and *iii)* the dynamics of network change over time [10, 11].

In terms of personalization, SNA is applied to find the preferences of users by studying the relevant parts of their social networks and communities. Is is well known that users online activities and significant information from their social networks (user graph) provide solutions to address research related to trust, influence, reputation, privacy disclosure and temporal character of data. For instance, trust may be computed as a function of the path between the source and target user; and a high user's betweenness centrality may be an indication of his/her influence.

2.3. Detection of Groups in OSNs

Using platforms such as Twitter and Meetup can be a highly efficient way of connecting with like-minded people. Social Networking sites allow participants to join networks arranged around a topic, an activity or discipline areas. Users engage in discussions and share information within these networks, particularly if they find like-minded people. These ideas are motivated by concepts presented in studies about social identity [12]. Indeed, social identity is the knowledge that an individual has about his/her own membership of a social group or category, as well as the value and emotions associated to that membership.

Having a collective perception of their social unit is enough for users to act as a group and be socially interdependent. In a group, people have some knowledge about their membership and share an emotional involvement with the other people in the group. Their social identity allows them to classify, compare, recognize similarities and order their social environment (hence the efforts of OSNs' users to form groups, join forums of similars, subscribe in lists, etc.).

According so, social attraction is a factor that is present in OSNs and this represent an advantage in the formation of groups in the given social context. If groups of similar users can be detected and their common interests modeled, they can be targeted with group recommendations. Given this scenario, researchers have focused on answering these questions: What is the nature of a group? How are groups formed? How are recommendations computed for groups? What interfaces are best for sharing recommendations with groups? What are the privacy issues in showing recommendations to groups? [13, 14].

Members of a group match an agreed (implicit or not) group prototype and the more prototypical a member is, the more s/he tends to be consensually trustful or adopt the suggestions that arise in his/her community. Summing-up, OSNs makes group-forming easy which is conducive to the study of new groups and new kinds of groups.[1] Indeed, in scenarios when the number of recommendation lists that can be produced is limited, groups detection strategies are applied to address this issue [15].

2.4. Summary

In this chapter we have provided a general review of the main concepts which are the basis for the next chapters. We have explained the relevance of Online Social Networks (OSNs) and their study in fields that involve the analysis of new socially constructed knowledge structures. Then, we have described OSNs interdependence with personalization, recommender systems and social factors like group formation. In addition, we have presented the definitions of Social Web Mining and Social Network Analysis as research areas that support techniques for modeling users' interests.

Chapters 3 through 7 present their particular literature review. Thus, each chapter contains a section of *Related Work* which corresponds to its specific work.

[1] Theoretical foundations related to social and economic effects of Internet are proposed by Clay Shirky in his book 'Here Comes Everybody: The Power of Organizing Without Organizations'. `https://en.wikipedia.org/wiki/Here_Comes_Everybody`

Part I

Users Modeling based on their Posts

Chapters 3 to 5 of this thesis will present different approaches to model users' topic preferences. The proposed methods are derived from the analysis of the content posted by the users through 'word embedding' strategies. The approaches, experiments and results provide understanding of the use of *word2vec* and its varied ways of real-world applications. Most of the work done in this chapters was published in [16, 17, 18].

Chapter 3

MULTI-TOPIC PREFERENCE MODEL FOR TWEETERS

The problem we address in this chapter is the identification of users' implicit topic preferences by analyzing the content categories they tend to post on Twitter. Our proposal is significant given that modeling their multi-topic profile may be useful to find patterns or association between preferences for categories, discover trending topics and cluster similar users to generate better group recommendations of content. In the present work, we propose a method based on the Mixed Gaussian Model to extract the multidimensional preference representation for 399 Ecuadorian tweeters concerning twenty-two different topics (or dimensions) which became known by manually categorizing 68,186 tweets. Our experiment findings indicate that the proposed approach is effective at detecting the topic interests of users.

3.1. Research Problem

In the light of the massive digital information people are exposed to, they show interest in diverse topics to a greater or lesser extent. Quantifying and measuring a user's degree of interest in certain content and finding its correlation with his/her preference for another topic is a challenging task, especially in social media platforms where the user interests are not static. For example, people highly engaged to culture-related topics may often retweet posts about next concerts, but when their favorite soccer team wins a match, they generate posts according to that. Therefore, identifying this kind of topic preferences association represented as a multidimensional user model, (*MUM*), may be meaningful to define how much the user shows interest in content categories as well as to group like-minded users and address better recommendations for them.

In the context of Twitter, automatically classifying a tweet into a topic category is hard to achieve. Indeed, having a group of words that form a sentence of less than 140 characters[1] and that contains abbreviations, emoticons, URLs and mentions of other users, which in particular do not provide a relevant meaning by themselves, makes the semantic analysis a challenge. Then, during the classification work of a tweet, the capture of other words like hashtags, proper nouns, compound nouns and verbs lead to a better topic assignment. Accordingly, to make the implementation of the comprehension and classification tasks of a tweet possible (as the basic step to then associate topic interest to tweeters) we propose a method that merges language modeling techniques and the Expectation Maximization algorithm [19] (*EM* for Mixture of Gaussians).

The strategy is independent from the users' posts language which makes it feasible to take Spanish tweets posted by Ecuadorians as our case study. Respectively, aggregating the Mixed Gaussian Model (topic soft assignments) of the target users' tweets in order to find their *MUM* is useful to cluster them and find groups of users interested in the same topics and to the same extent.

There are loads of research works in the field of users' topic preferences modeling. However, to the best of our knowledge, our proposal represents the first attempt to quantify the degree of responsibility a topic has over a given tweeter. That is to say, the method allows to identify the percentage in which each category (*i.e.*, topic) takes part in the user profile.

[1] When the dataset was collected Twitter posts were limited to 140 characters. Currently, the length of a tweet may be up 280 characters.

Given this real-world application scenario, our scientific contributions are:

- a method to define the multidimensional user model *MUM* for tweeters, which can be further applied to cluster like-minded users and design group recommendations;
- an evaluation of the accuracy of the proposed method considering, in terms of a comparative analysis, a baseline approach which takes a *ground-truth dataset* of labeled tweets. In such way, the *MUM* approach is compared to the results of a traditional machine learning classifier.
- a detailed validation of our approach that shows its effectiveness in modeling users. We show that similar tweeters, whose profiles were modeled with MUM, are able to be grouped together.
- The work of this chapter was published in Lorena Recalde, Ricardo Baeza-Yates. “What kind of content are you prone to tweet? Multi-topic Preference Model for Tweeters” in Workshop on Social Aspects in Personalization and Search, collocated with ECIR 2018, Apr 26, Grenoble, France.

In summary, in this chapter we propose a novel method for unsupervised and topic-based “soft” classification of tweets. Such approach is used to model Twitter users. The remainder of the chapter content is organized as follows. Section 3.2 summarizes the context of the present research and related literature; moreover, we draw a comparison to our proposal; Section 3.3 describes our approach; in Section 3.4 we present the experimental framework and the obtained results. Finally, some observations, findings and future directions are discussed in Section 3.5.

3.2. Related Work

Human factors such as need for approval, acceptance of a community, reputation as an expert, friendship, among others are implicitly present in

Online Social Networks, OSNs [20]. Few of these factors have settled in a specific social media with more intensity than others, and human curiosity satisfaction is a widespread one. For example, *curiosity* to know about acquaintances' activities is prevalent in Facebook; on the other hand, *curiosity* to know (and learn) about new content related to one's topics of interest is seen in Twitter. Therefore, to meet user's curiosity it is necessary to present them with others' posts that are certainly of their preference.

Modeling users' profiles is essential to find the topics they enjoy consuming and provide the curious users with meaningful information. Accordingly, in this section we present related works considering *Tweeters Modeling for Recommender Systems* whose aim is to link tweeters with the corresponding content/items. Later, *Group Formation and Group Recommendation* is detailed due to the further application of our approach in this area. Finally, as our proposal is based on the use of EM to find the *degree of responsibility* a topic has over a tweet, *Tweets Classification* works are also described.

3.2.1. Tweeters Modeling for Recommendation

Recommender systems predict if an *unseen item* is going to be of interest of a target user. To address the problem of recommendation in the Social Web such systems mine people's interactions, trust connections, previously adopted suggestions, use of self-annotated content (*i.e.* through hashtags), groups subscription, among others [21].

Tweet recommendation has been studied due to the constant threat of content overload in the users time-line. In [22], the approach makes use of three components: tweet topic level factors, user social relation factors and explicit features like authority of the tweet creator and quality of the tweet to define if a tweet can be recommended. Unlike our proposal, this article bases the user model in the social connections and not in topics of interest.

Research presented in [23] proposes a URLs recommender system for tweeters based on content sources, topic interest models of users, and social voting. Their findings show that topic relevance and social interactions were helpful in presenting recommendations. As in our approach, [23] builds the user's profile from his/her own tweets. However, they work with the weighting scheme *TF-IDF* [24] to find the relevant topics for the user while we apply word embeddings.

In [25], Weng *et al.* propose an approach to identify and rank topic-influential Twitter users. A main step in the approach is the topics modeling per user. The authors apply Latent Dirichlet Allocation (LDA [26, 27]) to distill the topics that tweeters are interested in. To identify the topics that are related to the user, they aggregate the tweets posted by him/her so they can be seen as a *document*. Similarly, in our approach we need to aggregate the content generated by the user. However, instead of aggregating the user's tweets we aggregate the tweets' embeddings. Besides, unlike applying LDA for topic modeling, we use the Mixture Gaussian Model.

3.2.2. Groups Formation and Recommendations

From a general perspective, the benefits of using a microblogging platform such as Twitter emerge from the activity of the users themselves. This social and data-oriented phenomenon is known as collective intelligence [28, 29]. For example, a recommender system that tracks events liked by the users may infer that the users who attend musicals twice a month also attend plays once a month. This generalization may be done because the system learns patterns from the behavior of the whole community. In such a case, like-minded users need to be grouped and analyzed together.

A Group Recommender System supports the recommendation process by using aggregation methods in order to model the preferences of a group of people [30]. This is needed when there is an activity (domain) that can be done or enjoyed in groups [31]. For our proposal, it may be possible to detect groups of tweeters interested in the same topics and suggest for them, for example, lists to subscribe in.

3.2.3. Tweets Classification

In terms of tweets classification, in [32], 5 content categories (News, Events, Opinions, Deals, and Private Messages) are proposed in order to classify short text. In this work, tweets are modeled considering 8 specific features which lead to determine the class of a tweet. For example, one of the features is *presence of time-event phrases* that, in case it is true for a given tweet, might relate it to the Events category. On the other hand, considering the feature *presence of slang words, shortenings* as true for the tweet suggests a Private Message class. While, this method works with more general categories and a supervised classifier, our proposal allows a 300-dimension representation of tweets which are later classified (with soft assignments) considering 22 categories.

In [33], the problem of hashtag prediction is investigated to recommend the users proper hashtags for their tweets. As a first step, Naïve Bayes and the Expectation Maximization algorithm are employed to classify English and non-English tweets. Later, LDA with Gibbs sampling is applied to find the tweet-topic distribution. Like our proposal, EM was employed as a means of unsupervised classification of tweets. However, we used it to model the tweets depending on the hidden topics, to then seeing the tweet model as a percentage allocation per topic. On the other hand, the mentioned work uses EM to identify the probability of a tweet as being writing in English and later, they do a hard class assignment.

Topic modeling with LDA-based approaches has been broadly used as means of tweet classification [34]. However, supervised learning to classify tweets according to topics has been studied as well. In [35], the authors propose a method where a group of four classifiers are trained to learn the topics for tweet categorization. They define ten topics and with the help of annotators, they classify a set of hashtags into those topics. Once the hashtags are classified, they can label tweets (containing the hashtags) with the corresponding topic. In their experiments they try to find the features and feature classes relevant to maximize the topic classification performance. The baseline method employed to validate our approach follows the same strategy in terms of supervised classification.

In [36], a real-time high-precision tweet topic modeling system is proposed. 300 topics are considered, and the proposal is based on an integrative inference algorithm trough supervised learning as well. In contrast, we present a method to categorize tweets in an unsupervised manner.

Our method is effective in calculating the degree of participation of a topic in a given tweet (soft clustering) and no labeled data is required.

3.3. Approach

In this section we present the core phases that were implemented to *i)* identify the level of participation or responsibility that each category has over a tweet and *ii)* aggregate the user's tweets classification extracted in the former phase to then define his/her multidimensional user model *MUM*. The *MUM* approach, consists of:

1. **Tweets Modeling.** By using word2vec [37] we find a vector representation for a given tweet.

2. **Extraction of the Suitable Number of Topics.** A widely known technique to define the number of topics hidden in a corpus is the Elbow method [38]. We use it to decide how many dimensions our tweet/user model will have.

3. **Tweets Classification.** To define the topics' responsibility degree over a tweet we use EM. As a result, every tweet will have a vector with K dimensions where K depends on the number of topics. Every feature value of the vector is the percentage of the participation of the corresponding topic in the given tweet.

4. **Twitter Users Model.** Once the strategy to model a tweet is established as formulated in the previous phase, it is applied to the tweets of the target user. We aggregate the results to define the multidimensional user model.

5. **Grouping like-minded Users.** MUM provides a profile of tweeters who may be clustered in groups of homogeneous interests.

What follows presents the details of our approach considering each task.

3.3.1. Tweets Modeling

A collection of tweets is employed to build a vector representation model for the words (vocabulary). We use a word embedding strategy based on a neural language model, *word2vec*, and its implementation *skip-gram*. The model learns to map each word into a low-dimensional continuous vector-space from its distributional properties observed in the provided corpus of tweets[2]. To train the model, a file that contains a tweet per row is needed.

Other input parameters have to be provided: *size* or number of vector dimensions, *window* or maximum skip length between words, *sample* or threshold for how often the words occur, and *min_count* or minimum number of times a word must occur to be considered. The output of the trained model is a vector for each word in the corpus. Since the vectors are linear, we can sum several vectors to obtain a unique model representation (additive compositionality property). Therefore, in order to create a model of a *tweet* from the words in it, we sum its words vectors. Let W_t be the set of words in the considered tweet t. By taking their embeddings, w_t being the vector for a given word, we build the tweet model as follows:

$$w'_t = \sum_{w_t \in W_t} w_t \tag{3.1}$$

Then, the vector representation for t is w'_t.
The detailed methodology which covers tweets cleaning/pre-processing and text modeling is explained in Chapter 4. It is worth mentioning that the tweets are being represented as 300-dimension vectors. The values that the parameters took in this study are reported in the Section 3.4.3 to allow our experiments to be reproduced.

[2] https://code.google.com/archive/p/word2vec/

3.3.2. Extraction of the Suitable Number of Topics

To define the number of topics in which tweeters tend to get involved, we take the w_t' or tweets representation extracted previously and try to find the appropriate number of clusters of tweets. Therefore, we may find a meaningful topic per cluster by inspecting the tweets in it (in case the clusters need to be labeled). To separate the tweets into clusters, we applied *K-Means++* [39]. This method spreads out the initial set of cluster centroids, so that they are not too close together. By applying *K-Means++*, it is possible to find an optimal set of centroids, which is required to have optimal means to initialize EM.

The intuition behind clustering is that objects within a cluster are as similar as possible, whereas objects from different clusters are as dissimilar as possible. However, the optimal clustering is somehow subjective and dependent of the final purpose of the clusters; that is to say, the level of detail required from the partitions. The clusters we obtain may suffer from a wide variation of the number of samples in each cluster (*e.g.* few tweets talking about religion and lots talking about politics) so the distribution is not normal. Nevertheless, we can select the number of clusters by using the heterogeneity convergence metric as the *Elbow* method specifies. We are required to run tests considering different K values (*i.e. number of clusters*). To measure distances between observations we use the cosine distance metric. Then, having K, we measure the intra-cluster distances between n points in a given cluster C_k and the centroid c_C of that cluster.

$$D_k = \sum_{i=1}^{n} cosineDistance(x_i, c_C)^2 \;\; x_i \in C_k \;\wedge\; n = |C_k|$$

Finally, adding the intra-cluster sums of squares gives a measure of the compactness of the clustering:

$$het_k = \sum_{k=1}^{K} D_k \tag{3.2}$$

In the *Elbow* heuristic we need to visualize the curve by plotting the heterogeneity value het_k against the number of clusters K. At certain point, the gain will drop, forming an angle in the graph. Therefore, the graph where we have the heterogeneity versus K allows us to look for the “Elbow” of the curve where the heterogeneity decreases *rapidly* before this value of K, but then only *gradually* for larger values of K. The details of the analysis for the case of our study are presented in the experimental setup (Section 3.4.3).

While doing the experiments with different K values, we need to keep track not only the heterogeneity (used to apply the Elbow method), but also the centroids c_C calculated for the clusters.

3.3.3. Tweets Classification Using the EM algorithm

Mixture of Gaussians is one of the probabilistic models that can be used for observations soft-clustering. The model assumes that all the observations are generated from a mixture of K Gaussian distributions with unknown parameters. Then, after learning the properties of the observations, each mixture component represents a unique cluster specified by its weight, mean and variance. Mixture models generalize K-Means clustering by taking into account information about the covariance structure of the data as well as the centers of the latent Gaussians.

When the number of topics, specified by the number of clusters found in the previous phase is obtained, the next step is the implementation of the Expectation Maximization (EM) algorithm. EM is sensitive to the choice of initial means. With a bad initial set of means, EM might generate clusters that span a large area and are mostly overlapping. Then, instead of initializing means by selecting random points, we take the final set of centroids calculated before (suitable set of initial means). Indeed, the initialization values for EM will be: *i)* initial means, the cluster centroids c_C extracted for the chosen K; *ii)* initial weights, we will initialize each cluster weight as the proportion of tweets assigned by K-Means++ to that cluster C_k; in other words, n/N for $n = |C_k|$ and $N =$ total number of tweets; iii) initial covariance matrix, to initialize the covariance parameters, we compute $\sum_{i=1}^{N}(x_{ij} - \mu_{C_kj})^2$ for each dimension j.

When the initial parameters are set, the input for the algorithm will be the vectors which belong to the tweets that we want to model. The EM algorithm will be in charge of defining the degree of responsibility the topics will have over each tweet. Then, the output after running the algorithm will be the *responsibility matrix*[3] which cardinality is NxK. The rows of the matrix specify in which extent the observation x_i was assigned to the different K topics (columns). In other words, if the topic 0 (or cluster 0) has full responsibility over the observation the value is going to be 1. If we see shared responsibility between eight topics over another tweet, the sum of those values will be 1 (refer to Section 3.4.3 to see an example).

3.3.4. Extraction of the Multidimensional User Model

Having the responsibility matrix, we need to identify which tweets (rows of the matrix) correspond to the given user (noting t as a modeled tweet $\in T_u$). Whence, for the user being analyzed we will have a $|T_u|$xK submatrix, which will be noted as U. To establish the Multidimensional User Model (*MUM*), we apply next equations.

$$sum_j = \sum_{i=0}^{|T_u|-1} t_{ij} \tag{3.3}$$

For $j \in [0, K-1]$. Then, we sum the vector values j to obtain the total:

$$total = \sum_{j=0}^{K-1} sum_j \tag{3.4}$$

Finally, the model for the user (given by dimension j) will be represented as percentages:

$$MUM_j = (sum_j/total) * 100 \tag{3.5}$$

[3] Refer to the repository `https://github.com/lore10/Multidimensional_User_Profile` to access the code related to the EM algorithm (datasets and other files are also included).

In conclusion, MUM is going to be a vector of K dimensions that models the given user according to the topics he/she tends to tweet about. The j values will express the extent of topic participation in the user's Twitter profile.

3.3.5. Grouping like-minded Users

One of the applications of the multi-topic model of users would be clustering similar users to analyze audiences on Twitter, targeting certain groups of tweeters with recommendations, studying subtopics of interest given a group, among others. In the case of our study, this step was taken to evaluate the proposed approach performance. The clustering algorithm we used was K-Means++ [40], which implementation is provided in the tool Graphlab [41] for Python (K-Means with smart centers initialization). The feasibility and low cost of the algorithm to process partitions of big datasets allow the wide use of this clustering method oriented to many applications. To define the optimal number of groups of users, given the dataset in analysis we also applied the Elbow Heuristic.

3.4. Experimental Framework

In this section, we detail the experimental framework which validates our proposal. We present a case study based on a real-world scenario and have divided the section in the following. First, we describe the datasets employed during the experiments; then, we provide an explanation about the baseline approach used for comparison. Later, the experimental setup followed by the corresponding results are discussed.

3.4.1. Data Collection

To run the experiments and implement our approach we need some datasets:

- a set of tweets to train the word2vec model,

- a list of users and their tweets/retweets, and
- a list of users whose profile or preferred topic is well known in order to evaluate the performance of the baseline method and the proposed approach.

The detailed description of the data is provided next.

Training Corpus to obtain the Vocabulary Model

As it was said before, we collected datasets with the aim of applying word2vec. The trained model, which was the result of the research done in [17], was used in the present work because of the advantages the dataset presented: *i)* diverse nature of content because of a pool of 319,889 tweets posted by Ecuadorian users during a month, and *ii)* the authors have knowledge of the context involved, *i.e.* hashtags and their topics, meaning of referenced places and events, and public figures as well as the category their posts fall in.

The previous research explored and validated the quality of the training dataset. Indeed, the vocabulary extracted and represented as vectors covers most of the words Ecuadorian tweeters tend to use. Therefore, it suggests that the model can be generalized for similar scenarios as the one presented in this research. Besides, after doing some tests, it was found that the appropriate representation for this kind of input text (short sentences in Spanish) is of 300 dimensions.[4]

The trained model corresponds to the output of the approach phase presented in Section 3.3.1, Tweets Modeling. Once these tweets are modeled we identified the number of topics involved (Section 3.3.2) and the centroids to then initialize EM. Moreover, the vocabulary vectors are later used to define other tweet models.

[4] In addition to our experiments we want to mention that Google uses a 300 dimension vector to represent words and has published a pre-trained model. The pre-trained Google word2vec model was trained on Google news data (around 100 billion words) and contains 3 million words and phrases in the model vocabulary.

Sample of Users and their Timeline

A set of 360 users was sampled from the list of tweeters who created the tweets in Section 3.4.1. Every tweet in the corpus has meta-data that has information about of it, such as ‘text’ of the tweet, ‘creation date’, ‘list of hashtags’ contained in the tweet, ‘user’ (id number and screen name) who posted the tweet, among others. Given that we have a list of 37,628 users, we had to randomly sample 360 of them due to the Twitter API rate limits. To apply the proposed method, we extracted the last 3,200 tweets from their accounts. Finally, the amount of tweets collected from the users’ timelines is of 236,453.

Sample of Users for Approach Evaluation

We considered a list of 39 political figures who have worked in the government in decision-making positions or who were candidates for government positions during the 2017 elections. Besides that their tweets were collected in time of election campaigns (Nov 2016), we validated their political profile in the platform ‘Smart Participation’ (Participación Inteligente).[5] The official information published there confirmed their candidature as politicians and affiliation to a political party. We query their Twitter accounts and extracted a total list of 58,533 tweets. These tweets were added to the set previously obtained. Then, we will apply our approach (Section 3.3.3) considering a dataset of 294,986 tweets in total.

It is worth mentioning that those tweets belong to the 399 users. 39 of them are politicians intentionally added to test the accuracy of the proposed approach. In other words, the political figures help us to validate if after getting their *MUMs* and clusters (Sections 3.3.4, 3.3.5), they are going to be found as similar (homogeneous profile models) and put together. In such a case, we can assure that the tweets and users are being correctly modeled.

[5] Voting Advice Application in Ecuador, `https://participacioninteligente.org`.

3.4.2. Baseline Approach

To compare the performance of the MUM approach at modeling tweeters, a baseline method is proposed. We elaborate a strategy made of core techniques. What follows is a map of our approach phases and the decisions made to construct the baseline.

1. **Tweets Modeling.** The dataset of tweets presented in Sections 3.4.1 (training corpus) was modeled by applying *TF-IDF* [42].

   Such a strategy is one of the core information retrieval techniques used to create a vector representation of text.

2. **Extraction of the Suitable Number of Topics.** To build a ground truth about the topics hidden in the tweets dataset and get a subset of classified tweets, we extracted a list of the most frequent hashtags present in the tweets. We inspect the hashtags to identify keywords corresponding to a given category. For example, the hashtags *#ecu911, #routesecu911 and #ecu911withme* lead us to define the topic *Citizens Safety and Emergencies.* As a result, 22 topics were extracted and the corresponding tweets, which contained the studied hashtags, were labeled accordingly. Usually, this manual classification technique allows the categorization of 20% of the tweets. In our case, from 319,889 tweets we classified 68,186 which correspond to the 21.3%. The 22 categories define the number of dimensions the users model will have.

3. **Tweets Classification.** In our approach, EM is used to generate a topic-soft-assignment for each tweet (Mixture of Gaussians). For the baseline approach, we will predict the *topic* of the given tweet by applying a traditional machine learning algorithm. We did a series of tests to select an appropriate classification algorithm. First, we chose three machine learning approaches used to realize *multi-class* prediction. These were logistic regression, decision trees and boosting trees. Then, we took 80% of the previously label tweets to be the training dataset. The rest of the tweets were used to test the models.

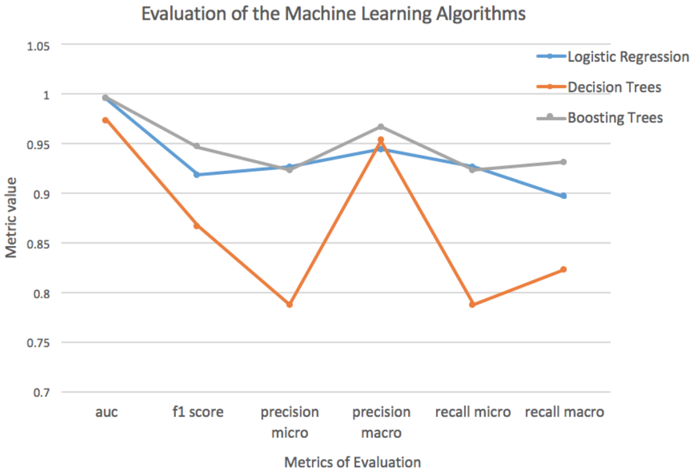


Figure 3.1: Comparison of the performance of the machine learning algorithms (multi-class prediction).

As it is shown in Figure 3.1, Boosting Trees algorithm [43] outperformed the others, so it was used to classify the users' tweets in next phase. The algorithm is based on a technique called gradient boosting, which combines a collection of base learners (i.e. decision tree classifiers) for predictive tasks. It can model non-linear interactions between the features and the target. It is worth clarifying that for precision and recall we calculated the micro and macro values [44]. Micro precision/recall calculates the metrics globally by counting the total true positives, false negatives, and false positives. On the other hand, the macro value calculates the metrics for each label and finds their unweighted mean (label imbalance is not considered). We use the trained boosted trees model to get the class/topic of the new observations (294,986 tweets of the 399 users with their TF-IDF representation). As output, we obtain the *class* and the corresponding *class-probabilities*.[6]

[6] `https://turi.com/products/create/docs/generated/graphlab.boosted_trees_classifier.html`

4. **Twitter Users Model.** According to our proposal, the MUM method aggregates the results of the EM algorithm applied over the tweets of a given user. On the other hand, considering the baseline approach, we take the tweets of the target user T_u and their probabilities associated to the class prediction P_t (results of the boosting trees classifier). At last, to define the user's model M for the baseline, we average the probabilities obtained for each of the 22 classes:

   $$M_j = avg(\sum_{i=0}^{|T_u|-1} P_t^{ij})$$

   For $j \in [0, 21]$.

   At the end of these baseline method's stage, the users will have a set of values (j) that quantify the level of preference of the user for the corresponding 22 topics.

5. **Grouping like-minded Users.** We take this phase to evaluate the performance of the baseline approach. In order to compare our method and the baseline, this step was identically applied in both MUM and M (refer to Section 3.3.5). More detail about the obtained results is given in Section 3.4.4.

3.4.3. Experimental Setup and Strategy

The parameters used to apply word2vec over the *training corpus* are: *size=300*, *window=5*, *sample=0* and *min_count=5*. Other parameters are not modified and take the default values. The output of the word2vec model contains a vocabulary of 39,216 words represented as vectors. Equation 3.1 is applied to have the vectors of the tweets in the training corpus. When the set of w_t' is ready we can move on to the next phase to define the number of clusters in which the tweets are classified. We run some experiments considering K (number of clusters to find) equal to several values. For each given K we apply K-Means++ to cluster the tweets and after that, we will be able to calculate the heterogeneity

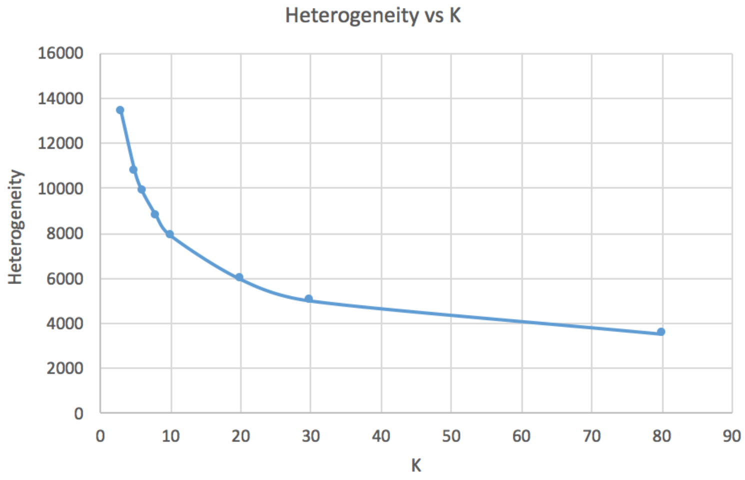
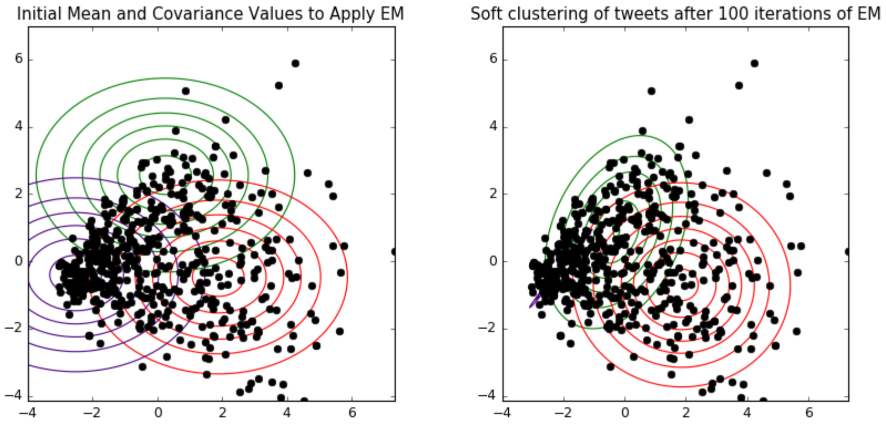


Figure 3.2: Elbow Heuristic: Heterogeneity vs K values.

(Equation 3.2).[7] The results are shown in Figure 3.2 where we have the heterogeneity vs K plot. The Elbow Heuristic specifies that by analyzing this plot, we can define the optimal number of clusters for the provided data points. The diagram shows that the gain reduces significantly from K=3 to K=20. Besides, we see a flattening out of the heterogeneity for $K >= 30$ (overfitting for larger values of K). So, it might indicate that the K searched is in a range of 20 and 30. To make a decision, we take into account the manual classification of the training tweets in the baseline method, where *22 topics* were found. Whereby, as the Elbow Heuristic also suggests, we consider 22 topics, or $K = 22$ to continue working on our approach. The centroids for the 22 clusters are calculated and used to initialize the $means$ for EM. When applying the EM algorithm in order to get a soft topic assignment per tweet, we will be using the dataset of 399 users' tweets (39 of the users are political figures, which results are employed in Section 3.4.4 for validation).

[7] It has to be mentioned that for the given K we run K-Means++ with some initialization seeds: 0, 20000, 40000, 60000, 80000. The considered seed to define the centroids for our work was the one which reported the minimum heterogeneity.



(a) Initial Clusters

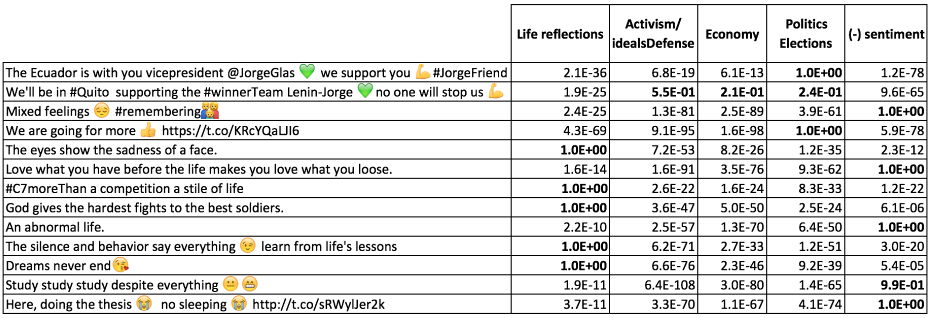
(b) Final Clusters

Figure 3.3: Visual Comparison of the Initial and Final states of 3 Clusters' Gaussians.

To visualize the work of the EM algorithm, we present Figures 3.3a and 3.3b. In order to facilitate graphics' data representation, we considered a sample of 500 tweets or data points that were transformed from 300 dimensions to two and we also defined K=3 instead of 22. Then, Figure 3.3a shows the 'shape' of the beginning clusters which have as centroids the *initial means* and the orientation of the *initial covariances*. After the EM algorithm runs and learns the new parameters from data the 'shapes' of the clusters change as it is shown in Figure 3.3a.

When EM converges, we will get the output of results. The resulting responsibility matrix is used to define the MUM of the users by implementing Equations 3.3, 3.4 and 3.5. As an example, Figure 3.4 shows 5 topics and the degree of responsibility they have over 13 tweets of a given user.[8] The user we took had 698 tweets and once we extracted his/her

[8] It is worth noting that, as other unsupervised methods, the names of the classes, categories or topics are not defined by the proposed clustering strategy. For the example in Figure 3.4, to provide the topic labels, we extracted and analyzed the tweets classified in the corresponding topic with a minimum value of 0.90. Doing so, we were able to



	Life reflections	Activism/ idealsDefense	Economy	Politics Elections	(-) sentiment
The Ecuador is with you vicepresident @JorgeGlas 💚 we support you 💪#JorgeFriend	2.1E-36	6.8E-19	6.1E-13	**1.0E+00**	1.2E-78
We'll be in #Quito supporting the #winnerTeam Lenin-Jorge 💚 no one will stop us 💪	1.9E-25	**5.5E-01**	**2.1E-01**	**2.4E-01**	9.6E-65
Mixed feelings 😔 #remembering👨‍👩‍👦	2.4E-25	1.3E-81	2.5E-89	3.9E-61	**1.0E+00**
We are going for more 👍 https://t.co/KRcYQaLJI6	4.3E-69	9.1E-95	1.6E-98	**1.0E+00**	5.9E-78
The eyes show the sadness of a face.	**1.0E+00**	7.2E-53	8.2E-26	1.2E-35	2.3E-12
Love what you have before the life makes you love what you loose.	1.6E-14	1.6E-91	3.5E-76	9.3E-62	**1.0E+00**
#C7moreThan a competition a stile of life	**1.0E+00**	2.6E-22	1.6E-24	8.3E-33	1.2E-22
God gives the hardest fights to the best soldiers.	**1.0E+00**	3.6E-47	5.0E-50	2.5E-24	6.1E-06
An abnormal life.	2.2E-10	2.5E-57	1.3E-70	6.4E-50	**1.0E+00**
The silence and behavior say everything 😉 learn from life's lessons	**1.0E+00**	6.2E-71	2.7E-33	1.2E-51	3.0E-20
Dreams never end😘	**1.0E+00**	6.6E-76	2.3E-46	9.2E-39	5.4E-05
Study study study despite everything 😐 😁	1.9E-11	6.4E-108	3.0E-80	1.4E-65	**9.9E-01**
Here, doing the thesis 😭 no sleeping 😭 http://t.co/sRWylJer2k	3.7E-11	3.3E-70	1.1E-67	4.1E-74	**1.0E+00**

Figure 3.4: Example of Topic assignment with EM algorithm.

MUM, the model presented a value of 49.1 for the topic '(-) sentiments' and 11.4 in 'life reflections' (highest category weights). The model of tweeters is finally obtained and may be used with many purposes.

Actually, to align the results with the goals of our research we cluster the users to define groups of tweeters with *similar profiles* or tastes about content topics (last phase of our approach, Section 3.3.5). By making use of the notion about heterogeneity and Elbow Heuristic we find that the users in our dataset form 5 clusters. To evaluate the behavior of our approach facing the chosen baseline, we introduced a set of politicians. The assumption behind this is that if their profile is well represented, they are going to be grouped in the same cluster. This validation is presented in next Section.

3.4.4. Validation of Results

The users we take to do this validation are well-known political figures who have a position in the government or were candidates in different democratic elections. The clustering algorithm we applied with the aim of validating the MUM approach as well as the results of the baseline method was K-Means++. The details about the results for both approaches are presented in Table 3.1.

annotate the category names.

Cluster ID	Total Size (Baseline)	Total Size (MUM)	Politicians Classification (Baseline)	Politicians Classification (MUM)
0	50	100	17	36
1	165	6	0	0
2	126	45	0	1
3	16	122	2	1
4	42	126	20	1

Table 3.1: Summary of Users Clusters: Baseline and MUM methods.

The Table also shows how the politicians were classified. In the case of the baseline implementation, we can see that there are two prominent groups of politicians. One group (cluster 0) covers 44% of them, while the other group (cluster 4) the 51%. By analyzing the centroids of the two clusters, we identified that *cluster 4*, differently from *cluster 0*, groups users who tend to talk more about economy. Compared to our approach, it is shown that MUM performance at clustering politicians has 92% of precision. From the 39 politicians, only 3 were left out of the political-related cluster. The 'screen_name' of these users are $lcparodi$, $ramiroaguilart$ and $mmcuesta$. By verifying their MUM (the 22 dimensions of the model) and their tweets, it is seen that their profiles are different from the rest of politicians who mostly talk about elections, economy and social issues. Instead, lcparodi tweeted about capital market and investment, ramiroaguilart posted about his interviews in radio media and talks directly to people loading his account of mentions (@); besides, our model separated mmcuesta because she talks about recipes/food and cooking, and she promotes few enterprises.

To visualize those results, we created two graphics which correspond to the clusters found after applying K-Means in the user data modeled with the *Baseline* and in the user data modeled with *MUM*. Figure 3.5 shows that the users modeled with the baseline approach are separated into two clusters. We have underlined the users lcparodi and mmcuesta

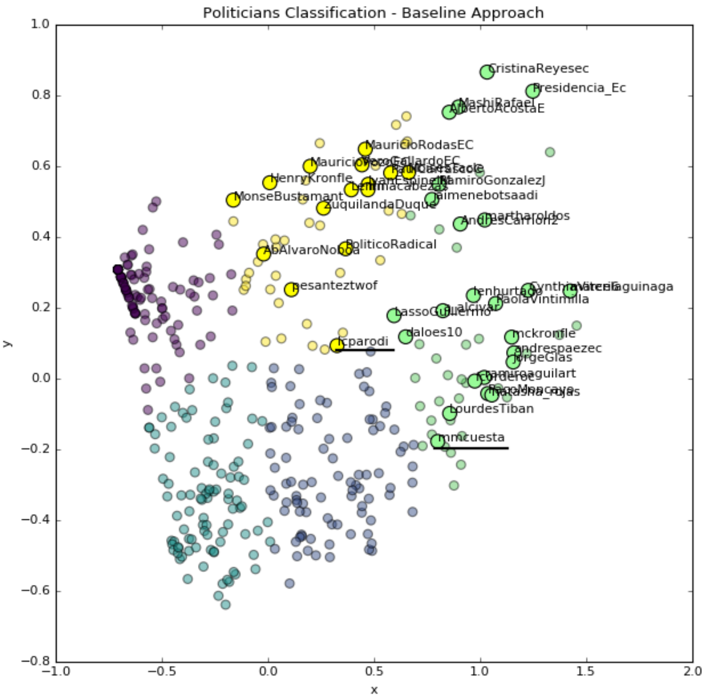


Figure 3.5: Clustering of Politicians who were Modeled with the Baseline Method.

because they are the ‘data points’ that belong to cluster 3 in the Politicians Classification - Baseline Method (Table 3.1).[9] Figure 3.6 presents the clusters obtained when the users were modeled with *MUM*. We see that among the 399 users, there is a clear cluster where the politicians gather (in yellow color). However, there are three of them, mentioned above, who are assigned other clusters.

[9] Results may be slightly different after applying the clustering algorithm in 2D data (fitted with PCA [45]).

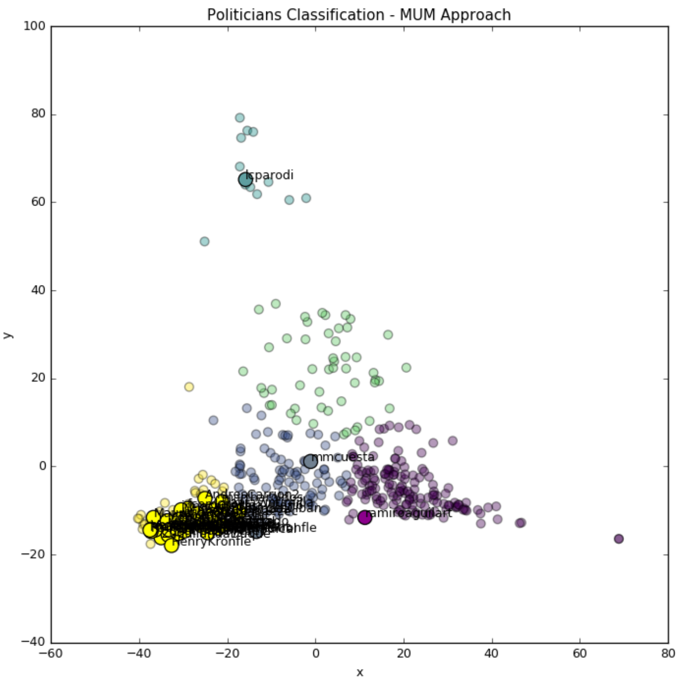


Figure 3.6: Clustering of Politicians who were Modeled with the MUM Method.

In order to make a deeper comparison of the politicians who were clustered together and the rest three, we did text mining over their Twitter accounts. As we already collected their time-lines, we consider every politician's tweets as a document; *i.e.*, there is a collection of 39 documents to be analyzed.

We apply TF-IDF over this corpus and found the most relevant words for the corresponding politicians' profiles. From among the most frequent words in the whole corpus, a list of meaningful words in the context of "politics" was extracted. The mentioned list contains 16 words: Ecuador,

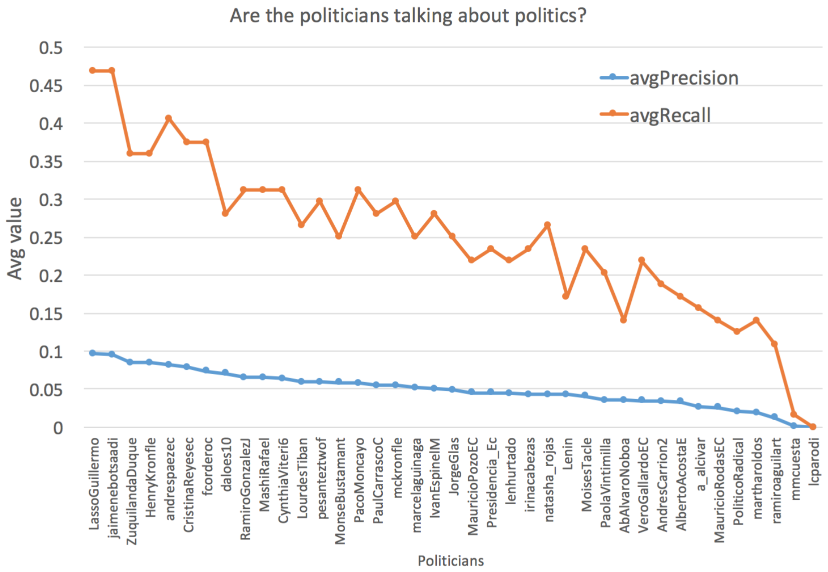


Figure 3.7: Relevance of "Politics" in the politicians' Twitter accounts.

government, country, Ecuadorians, president, 'the people' (pueblo), job, work, city, production, laws, taxes, congress, health, justice, and citizens.

In this experiment we try to find if the previous list was present among the relevant words extracted for the politicians. We worked with the 30, 50, 100 and 200 most relevant words taken from their profiles.

The results for the *average* precision and recall are shown in Figure 3.7.[10] As it is showed, the users $ramiroaguilart$, $mmcuesta$ and $lcparodi$ have the minimum values for both precision and recall; then, it is proved that they did not discuss about political issues as the rest of the politicians do.

[10] Note that the number of expected 'relevant' words to be retrieved is 16. Then, the reported values particularly for *precision* are low given that we average the calculation obtained by dividing the number of *True Positives* (with a maximum value of 16) by *30*, then by *50*, and so on.

3.5. Discussion

People may show preference for several topics to a greater or lesser extent. In this research, we have proposed a method that creates a vector representation of tweets by applying word2vec. Then, by using a Mixture of Gaussians through the EM algorithm, it calculates the degree of responsibility that a set of topics have over a tweet. Finally, we aggregate the results of the tweets which correspond to a given user to define his/her multi-topic preference model.

We have validated our proposal by comparing it with the results of a baseline approach. This evaluation showed that our method was able to cluster 92% of politicians in the same group, facing the results of the baseline method which divided the politicians in two clusters. In summary, we can conclude that our method is effective when modeling the topic interests of Twitter users.

Chapter 4

MEASURING THE EXTENT OF POLITICAL PARTICIPATION OF TWITTER USERS

This chapter presents an application of the use of word embeddings to quantify the *digital citizens'* participation on Twitter. We will explore the quality of results obtained when using *word2vec* and *Glove* for short text vector representation. After assessing both word embedding extraction strategies, we will detail the proposed approach which aims at measuring the "degree of interest in politics" (*DoIP*) of tweeters.

We address the problem of following-back recommendation and Twitter lists (to subscribe) recommendation by analyzing the users' DoIP, their friends' and the lists' where they are subscribed or belong as members. The results are meant to be used in the design of recommender systems. Besides, we will provide evidence about the positive association of users' topics of preference with their friends' interests and the kind of content to which they subscribe. We will conclude discussing the application of our method, benefits and possible new research challenges.

4.1. Research Problem

Online Social Networks ($OSNs$) have shown to be helpful to build a citizen identity in users. For instance, Twitter has proved to be a useful media platform that facilitates forms of political expression for their users. However, considering the extent of content published every second, the dynamic linkage among users and the wide purpose-oriented nature of Twitter, it is difficult to define the degree of political participation or interest of a *digital citizen*. Being able to solve this issue is of interest in recommender systems research, since identifying who the user is, what their interests are and which context is involved alleviates the problems of personalization in political-related suggestions.

In order to enhance personalization, a recommender system may be in charge of discovering the user's meaningful followers to suggest him/her to correspond reciprocally to this following relationship. Depending on the Twitter user's degree of interest in politics, *DoIP*, we may assume that not all of his/her followers are suitable to follow back. For instance, political figures in Twitter may have lots of followers but not all of them could be relevant to become *followees*. Besides, this kind of social recommendation could be appropriate because political actors may need to be aware of significant content (posted from others) to enhance their posts, receive important tweets they should retweet/reply or create a network of influencers in politics. Likewise, there are some users who subscribe to Twitter lists with the aim of accessing 'categorized' content.[1] Depending on the kind of content being shared in the lists and the topics of interest of the users, a Twitter lists recommender may be implemented.

In order to address the potential of Web-based recommendations and *digital citizens'* participation analysis, we present a method to identify the degree of interest in politics (DoIP) of Twitter users, considering Ecuador as a case of study. The time of data collection has co-occurred with political campaigns for Ecuadorian presidential elections (end of the year 2016) which made this context appropriate for our research.

[1] For instance, users may join a list that collects economy-related news and/or a list which groups the tweets of congressmen.

To the best of our knowledge, this work represents the first attempt to quantify the degree of interest in politics of digital citizens by analyzing their tweets. Accordingly, our scientific contributions are:

- a method to detect users' *DoIP*;
- the correlation of the users' *DoIP* facing their Twitter friends' *DoIP*[2] and their lists' *DoIP* which can be further applied to design recommendations.
- The contributions of our work were published in Lorena Recalde, Aigul Kaskina. "Who is suitable to be followed back when you are a Twitter interested in Politics?", in Proceedings of the 18th Annual International Conference on Digital Government Research (dg.o '17), Jun 7-9, 2017. ACM, New York, NY, USA, 94-99.

In summary, given this real-world application scenario, our objective is quantify the extent to which a user is interested in politics. However, the generic application of this approach in fields like sports, culture, and science, among others, would allow to measure the level of interest of users in each of the mentioned domains.

The chapter is organized as follows: Section 4.2 summarizes the context of the present research and the related literature; Section 4.3 describes our approach; in Section 4.4 we present the experimental framework and the obtained results; finally, in Section 4.5 we discuss about our main findings and present future directions to be considered.

4.2. Related Work

Digital citizenship is the ability to use technology to obtain political information. Besides, the frequent use of it elicits online participation of individuals in society [46]. New models of citizenship are arising due to the ways of interactions and communication provided by OSNs [47, 48].

[2] The term "friend" is used along the chapter to denote bidirectional or mutual relationship between two Twitter users.

Their in civic engagement, democratic participation, political party supporters interaction and voting is evident in last years. People enjoy from Internet use and may benefit from it through the opportunity it offers of letting them participate fully in society. For instance, the way how Facebook has changed political participation is analyzed in [49]. The researchers base their study on the level of users' engagement during the primary and general elections in the U.S. in 2008.

Another important social platform used in political communication is Twitter. It has been broadly considered in literature because of its effect on the last nations' socio-political changes. In [50] the researchers present an analysis of 28,695 tweets collected in 2011 during the Danish parliamentary election. They categorize those who enhance political communication and also test different democratic theories. The authors of [51] consider political discussions on Twitter during the Italian general election in 2013. They analyze in which degree online actions like getting political information and expressing oneself politically are associated to more demanding activities such as direct communication with politicians via e-mailing or offline meetings with political party supporters.

Differently, the study in [52] presents the *PoliTwi* system which aims at quickly detecting emerging political topics to be used to extend existing knowledge bases which may improve concept-level sentiment analysis methods. Ausserhofer *et al.* [53] analyze the relations between political actors and citizens in Twitter, the way they use the platform, and the political communication networks formed within Austrian political Twittersphere. In [54], the researchers study the Twitter activity of 380 members of the U.S. Congress (winter of 2012). Their findings show that officials use Twitter as a broadcast mechanism rather than as a way to engage in dialogue with the public. The presented works suggest that, by facilitating the access to political content as well as means to let citizens establish connections with like-minded users, social media can significantly contribute to political participation. Hence, our research is of importance in political related studies in online platforms.

The number of scientific works related to opinion mining and sentiment analysis in Twitter to potentiate government intelligence (or predict elections) grows increasingly. For instance, surveys about political opinion mining and political orientation classification in Twitter are presented in [55] and [56]. The goal of our project is not vote prediction or political ideology monitoring (as the aims in [57] and [18] respectively), but the application of methods to evaluate the DoIP of citizens in Twitter depending on the quantity of political-related tweets the user has published. Once this measure is found, it may be used as input to model the user political profile. Our contribution differs from the mentioned works because the proposed method employs word embeddings to classify words, find those usually employed in the context of politics, and then, automatically identify the tweets that contain those words.

In the context of recommendation, generally a system has to discover the interests of the target user in order to have an overview of his/her eventual needs and meet them. The Social Web has shown to be one of the richest sources for mining people's interests, personality and social interactions [58]. Research works like [22] and [59] propose methods to solve the feeds filtering and ranking problem (tweet recommendation). "Users to follow" recommendation is addressed in [60] and [61], but unlike our proposal the recommended users to follow are not necessarily followers of the target user.

Concerning the inference of user interests through the analysis of Twitter lists, Kim *et al.* [62] propose a method based on feature extraction to determine the relevant and common words that describe the "members" of a list. This work, in contrast to our study, analyses the content shared only by people added in certain lists to profile them. Our goal is to compare the tweeters' interest in politics with the quantity of political-related content that is shared in the lists which they subscribe and also where they are members.

To understand the difference of being a *subscriber* and being a *member* of a list we take an example. User *u* is able to create *n* number of lists. If s/he wants to group the tweets of her/his favorite story writers, s/he has to add them as "members" of the list l_u. In that way, instead of navigating form one writer's profile to another, *u* can access their posts by navigating the list l_u. If there is another user *y* who wants to be aware of the posts of those writers, he may “subscribe” to *u*’s list l_u as long as it is public. Thus, with our proposal we *i)* see how likely the users are friends with like-minded people; *ii)* study the tendency of users to subscribe to similar content to that they generate; and *iii)* discuss the grouping decision making of list creators when adding members to their lists.

4.3. Approach

This section describes the strategies used to automatically detect the *degree of interest in politics (DoIP)* of Twitter users. Accordingly, we make an analysis to identify if their followers are relevant to be followed back and whether or not to ‘follow’ the existing lists. The proposed approach consists of four phases:

1. Text Modeling. A corpus of tweets is employed to build a model with a word embeddings strategy. The model generates a vector representation of the corpus’ words. The vectors are assigned to clusters where one of them groups the words associated to politics. Later, its centroid will be used to measure the proximity of the tweets to it.

2. Calculation of $DoIP$ of Twitter Users. The tweets of citizens are converted into vectors by using the model so that we can apply the proposed method and measure in which extent they are interested in politics.

3. Following Back Recommendation. The association of users’ *DoIP* to the *DoIP* of their Twitter friends can be further used to build a following back recommender system.

4. Lists Recommendation. The association of users’ *DoIP* to the *DoIP* of the lists where they are subscribed can be further used to build a lists recommender system.

Next, we present the tasks that have been implemented per phase.

4.3.1. Text Modeling

Word embedding tools are able to give a vector representation to a word depending on the context it is commonly used (semantic sensitive). The purpose of these neural language strategies is to create a distributed model that can be used to get a vector representation of a given word, where words with similar meanings have similar representation. Word embeddings extract the syntactic information in a text corpus being then, independent from the language or the corpus content itself.

Correspondingly, through the application of vector operations it is possible to figure out if two words have been used in the same context or, for example, how distant a hashtag is with respect to a given tweet. To do so, primarily, the model needs to be trained by learning the words and their context which is introduced by the text of the training corpus. Details about the training corpus used are provided in Section 4.4.1. In this phase, we applied three steps presented next.

Text Preprocessing

In order to be analyzed, unstructured text found in OSN's needs to be preprocessed. For instance, punctuation or prepositions found in a document do not provide meaning or any context; therefore, they need to be removed. To clean up the tweets which shape the training corpus we take next actions:

- Capital letters conversion into lower-case letters.
- Special letters conversion. Vowels with accents and special characters, such as *'ñ', 'á', 'é'* (part of Spanish grammar) will take the form of *'n', 'a', 'e'*, respectively. (Use of Unidecode library[3]).
- Spanish stop words and punctuation removal. The list of Spanish stop words such as *'yo' ('I'), 'de' ('of'), 'en' ('in')* is provided by the *stopwords* NLTK library for Python.

[3] https://pypi.python.org/pypi/Unidecode

Models Training

When the training corpus is ready, we use it as the input for the model. We trained two word-embedding models, *word2vec*[4] and *GloVe*[5], in order to choose the one that performs the best given our context (short sentences in Spanish).

Word2vec was developed by Tomas Mikolov, *et al.* [37] to make the neural-network-based training of the embeddings more efficient. For this reason, it has become the *de facto* standard to extract word embeddings. It has two implementations, the Continuous Bag-of-Words model (CBOW) and the Skip-Gram model. *CBOW* predicts target words (*e.g.* 'pizza') given source context words ('Italian food like ...'). *Skip-gram* works in an opposite way and predicts source context-words (or sourrounding words) given the target word. While *CBOW* treats an entire context as one observation, *skip-gram* treats each context-target pair as a new observation, which tends to do better when we work with large datasets.

On the other hand, *GloVe*, developed by Pennington, *et al.* [63], employs techniques such as Latent Semantic Analysis (LSA) to learn the word embeddings. It uses global text statistics of matrix factorization instead of using a window to define the word context. In other words, GloVe constructs an explicit word-context or word co-occurrence matrix using statistics across the whole training text.

The models require some parameters to be indicated before training: in *word2vec* we specify (i) *size*, to define the number of dimensions of the vectors; (ii) *window*, to set the distance between the current and predicted word within a sentence; (iii) *sample*, to set the threshold for configuring which higher-frequency words are randomly down-sampled; (iv) *negative*, which represents how many noise words should be drawn; (v) *min_count*, to set the minimum frequency of occurrence a word in the corpus to be considered; and (vi) *alpha*, which sets the initial learning rate. Similarly, *GloVe* needs parameters to be determine in advance.

[4] https://code.google.com/archive/p/word2vec/

[5] https://nlp.stanford.edu/projects/glove/

The values required at the moment of instantiating the GloVe object are (i) the *number of components* or dimensions of the vector representation and (ii) the *learning rate* which scales the magnitude of feature weight updates in order to minimize the network's loss function. Later, when fitting the model we have to define (iii) the *epochs* or number of iterations over training corpus during training and (iv) the *number of threads* which sets number of worker threads to be used in calculations.

For this study, the employed parameters in the two models are reported in the experimental framework (Section 4.4). By the end of this step the models are ready to be used and generate embeddings for the words in the corpus.

Models Performance Comparison

Both qualitative and quantitative analysis may be applied in order to choose the appropriate model. Depending on the size of the training data set, the quality of the raw text, the prediction tasks performed, among other conditions, a model may outperform the other in terms of calculation of word analogies, word similarities or named entity recognition tasks. Given our study case, the evaluation of the behavior of the models and ground truth definition were done with the support of collaborators who are familiar with the Ecuadorian scenario (Communication Faculty, Universidad Casa Grande, Guayaquil-Ecuador).

For the assessment, two experiments were run: *Task 1.* given 5 words, retrieval of the *N* most similar neighbors; and *Task 2.* given 5 analogies, retrieval of the word that matches the corresponding relationships. Reports presenting the performance of the two models are detailed in next Section. It is worth mentioning that the vector dimension and window (parameters to train the models) were the same to execute the two models comparison.

Clustering and Centroids Extraction

A part of our approach is to compare a user's tweets to the political vector space in the model. Thus, to delimit the political-oriented vector space the words in the model associated to politics have to be grouped together. In other words, the political-related cluster needs to be defined. Some techniques may be used to do clustering depending on the vector dimensions and the expected performance [64]. For example, to achieve the goals of our research, the cluster assignment for words in the context of politics is important. In fact, the results of our approach will depend on finding out a centroid which efficiently represents the political-vector space.

The aim in this phase is to identify the words which are usually employed in Twitter posts related to politics and then, to calculate the corresponding centroid in their vector space. Generally, the centroid represents the mean vector of the observations in that cluster. In iterative clustering algorithms, to initialize the data points assignment, K observations of the dataset are chosen. The selected observations represent the initial cluster centroids. Then, in the first iteration every data point is going to be assigned to a cluster depending on its closest centroid (according to a specified distance function). With the first iteration done, new cluster centroids are calculated by finding the *average* of the vectors in each cluster.

The practice is repeated iteration by iteration until the same points are assigned to each cluster in consecutive rounds. Therefore, given our scenario, the political-related centroid may be considered as a general representation of the political context and provides a point of reference to say if other elements in the space (words, tweets, users) are near or far from it.

After having building the model and defining the categories or clusters of words we are able to calculate the cluster centroids. Next, we can extract the users' *DoIP* by applying the second phase steps described next.

4.3.2. Calculation of the Twitter Users' DoIP

To understand how users' DoIP is calculated, first we present how to find the similarity of a tweet with respect to clusters' centroids. Afterward, the proposed approach to find a user's DoIP is explained.

Similarity of a Tweet and Cluster Centroids

In our method, identifying the centroids of clusters (Section 4.3.1) is required because it is a baseline for the further classification of tweets.

Let C be the set of vectors that represent the centroids c for the clusters ($c \in C$) in the model. C will be used to measure how similar a given tweet t is to the different clusters. That is to say that we will be able to find the distance of a tweet with respect to the *politics centroid* c_p. It may perhaps be observed that after clustering, the cluster with the words in the context of politics can be easily identified. For instance, we may conduct a manual inspection of clusters and their words.

Given a tweet t, let W_t be the set of words it contains. Since the words in W_t are represented or 'modeled' as vectors (w_t), to find the vector that models t we average the vectors in W_t.

$$w'_t = avg(w_t), \forall w_t \in W_t$$

To measure how similar the tweet, represented as the vector w'_t, and the cluster centroids $c \in C$ are, we calculate the angle between the two vectors using the cosine similarity.

$$s_{tc} = cos(w'_t, c), \forall c \in C \tag{4.1}$$

After applying Equation 4.1, we will have an overview of how similar a tweet is to the c_p cluster (s_{tc_p}). The resulting values will range from 0 to 1, with 0 as the least similar vectors and 1 for the most similar pair of vectors.

User's DoIP Extraction

Given a Twitter user u we need to extract the tweets the user has posted in his/her timeline (T_u). The search function of the Twitter API supports this task retrieving the last 3,200 tweets for a given account. Every extracted tweet $t_u \in T_u$ is preprocessed (Section 4.3.1) and the steps described in Section 4.3.2 are followed in order to register the similarity of the tweets t_u facing the political cluster c_p.

Finally, the User's DoIP will be calculated as the average of similarities $s_{t_u c_p}$:

$$DoIP_u = avg(s_{t_u c_p}), \forall t_u \in T_u, c_p \in C \tag{4.2}$$

That is to say that if a user's tweets are mostly similar to the c_p cluster, the average calculation will present a DoIP near to 1.

4.3.3. Following Back and List Recommendation

Once the DoIP of a user is obtained, it is possible to evaluate whether a user is a highly politically active or a poorly politically active citizen. Depending on the amount of tweets in the context of politics published by the user, we can define to what extent he/she is interested in this topic. Given that both users' profiles and lists are depicted by a collection of tweets/retweets it would be reasonable to compare their models. Actually, the proper application of our approach to model the users' DoIP, their friends' DoIP and their Twitter lists' DoIP by evaluating their tweets, respectively, creates an advantage. This users/lists characterization may be useful when modeling their political profile and then determining if a recent follower is significant to be followed back or if a Twitter list may provide with content that meets the user's needs.

The recommender engine may be built based on the experiments results which are presented in Section 4.4.2.

4.4. Experimental Framework

This section presents the experiments done to validate the proposed approach. The datasets collected represent real Twitter users' activity and interactions. Next, we describe the data collection strategies and the experimental setup.

4.4.1. Datasets: Ecuador Case

We employed five different datasets to support our proposal.[6] In this section we explain their collection and usage.

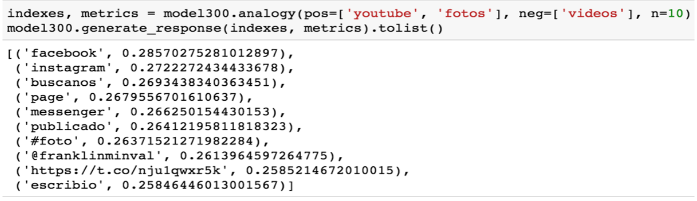
Training Corpus

As it was seen before, word embedding algorithms allow us to model text resulting in a vector per word. The corpus used to train the models contained 281,338 unique tweets (retweets are not included) collected from Ecuador during November 2016 by using the searching option for geo-localized posts provided by the Twitter Rest API[7]. To be sure about having political-related content in the corpus a pool of 38,551 tweets posted by Ecuadorian political actors was added. The total number of tweets in the training corpus was 319,889. It is worth mentioning that the success of the approach depends on a well defined corpus. Tests like fact checks through *analogies* in the model may validate the quality and ability of generalization of the corpus. Therefore, we conducted several qualitative validations following the fact checks technique. Indeed, this kind of tests let us deduce that applying ‘stemming’ to preprocess tweets in Spanish (Snowball algorithm) in our specific task affected the quality of the training corpus.

Some validation examples in the contexts of *media, politics, and football* are presented in Figures 4.1, 4.2, and 4.3. In the provided examples (model trained with word2vec), part of the code instructions is shown. Analogies may be tested to see the behavior of the model and the results (for example $n = 10$ displays the first 10 results extracted) will provide the word that matches the analogy (indexes), as well as the score of that match (metrics). By analyzing these results and comparing the scores obtained when training the model with/without applying stemming, the values were lower if stemming was employed. Thus, this preprocessing strategy was avoided. It has been explained the way we evaluated the

[6] The datasets used in our work are available in `https://github.com/lore10/Who_is_suitable_to_be_followed_back`.

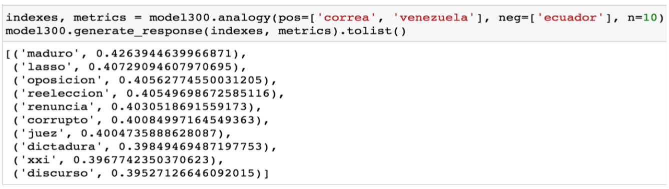
[7] `https://dev.twitter.com/rest/public`



```
indexes, metrics = model300.analogy(pos=['youtube', 'fotos'], neg=['videos'], n=10)
model300.generate_response(indexes, metrics).tolist()
```

```
[('facebook', 0.28570275281012897),
 ('instagram', 0.2722272434433678),
 ('buscanos', 0.2693438340363451),
 ('page', 0.2679556701610637),
 ('messenger', 0.266250154430153),
 ('publicado', 0.26412195811818323),
 ('#foto', 0.2637152127198228 4),
 ('@franklinminval', 0.2613964597264775),
 ('https://t.co/nju1qwxr5k', 0.2585214672010015),
 ('escribio', 0.2584644601300156 7)]
```

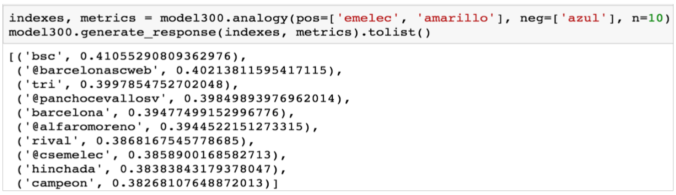
Figure 4.1: Analogy validation: ‘videos’ is to ‘Youtube’, as ‘pictures’ is to ‘Facebook/Instagram’.



```
indexes, metrics = model300.analogy(pos=['correa', 'venezuela'], neg=['ecuador'], n=10)
model300.generate_response(indexes, metrics).tolist()
```

```
[('maduro', 0.4263944639966871),
 ('lasso', 0.40729094607970695),
 ('oposicion', 0.40562774550031205),
 ('reeleccion', 0.40549698672585116),
 ('renuncia', 0.4030518691559173),
 ('corrupto', 0.40084997164549363),
 ('juez', 0.4004735888628087),
 ('dictadura', 0.39849469487197753),
 ('xxi', 0.3967742350370623),
 ('discurso', 0.39527126646092015)]
```

Figure 4.2: Analogy validation: ‘Ecuador’ has as its president ‘Correa’, while ‘Venezuela’ has as its president ‘Maduro’.



```
indexes, metrics = model300.analogy(pos=['emelec', 'amarillo'], neg=['azul'], n=10)
model300.generate_response(indexes, metrics).tolist()
```

```
[('bsc', 0.41055290809362976),
 ('@barcelonascweb', 0.40213811595417115),
 ('tri', 0.3997854752702048),
 ('@panchocevallosv', 0.39849893976962014),
 ('barcelona', 0.39477499152996776),
 ('@alfaromoreno', 0.39445221512733 15),
 ('rival', 0.3868167545778685),
 ('@csemelec', 0.3858900168582713),
 ('hinchada', 0.38383843179378047),
 ('campeon', 0.38268107648872013)]
```

Figure 4.3: Analogy validation: ‘blue’ is the color of the t-shirt of the Ecuadorian soccer team ‘Emelec’, then ‘yellow’ is the color of ‘BSC’.

quality of the training corpus. After the decisions made about preprocessing it, we used the dataset to let the models learn the embeddings to then assess their performance.

Sample of Politicians

31 Politicians (candidates for presidential/congress elections in February, 2017) were selected and their tweets were extracted. This dataset was

used to test the accuracy of the proposed approach. By using this data, we were able to identify the appropriate number of dimensions for the vector representation, as well as the number of clusters that group similar words.

The accuracy was defined as the number of politicians classified in the political-related cluster c_p over the total number of politicians. The greater the number of politicians classified in the c_p cluster, the better the adjustment of the parameters and the behavior of the model.

The cluster assignment was done by following the steps in Section 4.3.2. Then, we verified if the $DoIP_u$ (Equation 4.2) per politician was bigger than the rest of politician-cluster similarities, that is

$$avg(s_{t_u c_p}) > avg(s_{t_u c}) \forall c \in C, c_p \in C, c \neq c_p$$

The best result (*i.e.* 28 of 31 politicians classified in the politics vector space) was obtained setting a model of 300 vector dimensions and working with 5 clusters of words. In more detail, the combination of next parameters were tested: size=100 with 4, 5 and 6 clusters (worst results: $15/31$, $16/31$ and $16/31$ respectively); size=300 with 4, 5 and 6 clusters (results: $27/31$, $28/31$ and $27/31$ respectively); and size=500 with 4, 5 and 6 clusters (results: $21/31$, $22/31$ and $21/31$ respectively).

Sample of Ecuadorian Twitter Users

To calculate the DoIP of *digital citizens*, the users who generated the tweets used to train the models were collected. The intuition under this dataset collection strategy was to have *Ecuadorian users* who might fit in our study and preserve the context. In fact, the way of expressing oneself in posts, topics of interest and issues concerning a population may be community-dependent. Then, we made the context of the training data ‘agree’ with the context of the sample of users in analysis. In total, 37,628 users were identified, but due to the Twitter API rate limits, 3896 *digital citizens* were randomly sampled. To apply the proposed method and get the sampled users’ DoIP, we extracted the last 3,200 tweets from their accounts.

Twitter Users' Friends

To demonstrate the positive association of the user's DoIP and his/her friend's DoIP, we collected a list of 22,523 *friends* (bidirectional following) of the Twitter users in analysis. As it is required by our approach, we needed to extract the friends' tweets to apply the DoIP calculation method.

Twitter Users' Lists

As part of our experiments we analyzed the association of the users' level of interest in politics with the extent of political-related content shared by the lists where they are 'subscribed' or where they are 'members'. To verify the correlation of the user's DoIP and his/her lists' DoIP, we collected the tweets of 981 lists where the users in analysis were subscribed and the tweets of 1988 lists where the users participate as members. We calculated the DoIP of the mentioned lists by aggregating the results obtained by evaluating the corresponding tweets with our approach. The details are presented in next Section.

4.4.2. Experimental Setup

In this Section we detail the performed experiments and the corresponding results that we obtained.[8] First, we present the assessment of the behaviour of the word embedding strategies when modeling the provided corpus of tweets. This steps justify the selection of *word2vec* to be used in our approach. Second, we specify the details of the implementation of the approach and results. Finally, we demonstrate the positive relationship of the DoIP of users and entities they interact with like *friends* and *lists*.

[8] Toy examples and code corresponding to this section may be found in the repository `https://github.com/lore10/ICEDEG_tutorial`.

Query	Associated Terms
emelec	barcelona (or bsc) bombillo monumental
dictator	tyrant genocide criminal
nice	cute beautiful pretty
science	learning methodology technology
hope	faith strength optimism

Table 4.1: Queries and nearest terms (Task 1 annotated data).

Model Selection

The training corpus (Section 4.4.1) helped to model Twitter posts in the context of Ecuador events. The input parameters in *word2vec* were *size=300*, *window=10*, *sample=0*, *negative=0*, *min_count=5* and *alpha=0.025*. The number of word vectors obtained was 39,216. In contrast, *GloVe* took the values *no_components=300*, *learning_rate=0.05*, *epochs=30*, *no_threads=4* and *window=10*. Unlike word2vec, the implementation of GloVe for Python[9] does not allow to specify the minimum number of times that a word should appear in the training corpus to be considered (min_count=5 in word2vec). Then, the size of the vocabulary represented as vectors was of 356,305.

[9] https://github.com/maciejkula/glove-python

Analogy			Solution
youtube	videos	photos	facebook
trump	donald	hillary	clinton
winter	rain	sun	summer
correa	ecuador	venezuela	maduro
emelec	blue	yellow	barcelona

Table 4.2: Analogies and Appropriate Solutions (Task 2 annotated data).

To select the model to be used in our approach we performed two tasks that take the advantage of the semantic and syntactic regularities [65, 66] exhibited by these models. For Task 1, we collected the three most similar terms to a given word. The annotator was provided with 5 different words and a list of 30 related terms (per word) of which she should obtain the three nearest ones. Table 4.1 presents the input word or ‘query’ and the chosen associated terms. For Task 2, we collected the most suitable term that matches a given analogy. The annotator was provided with 5 analogies to be solved. Table 4.2 shows the analogies and the selected solution term. The results of the models evaluation are shown in Figure 4.4 (models performance at Task 1) and Figure 4.5 (models performance at Task 2). As it can be seen, word2vec outperformed GloVe in grouping similar words and solving accurately mathematical relations among words.

Finally, in a global and visual way we analyzed the relationship between certain groups of words. We applied “principal component analysis (PCA)” [45] and represented the words in a graphic. For example, we included words related to Ecuadorian soccer (names of teams), names of cities, terms related to research, terms associated with social media and words in the context of feelings. Figure 4.6a shows the word vectors modeled with *word2vec* and Figure 4.6b presents the words modeled with *GloVe*. It is not possible to draw conclusions through this visualization strategy which considers few samples. However, we may say that the models present a proper interpretation of the semantic role of words.

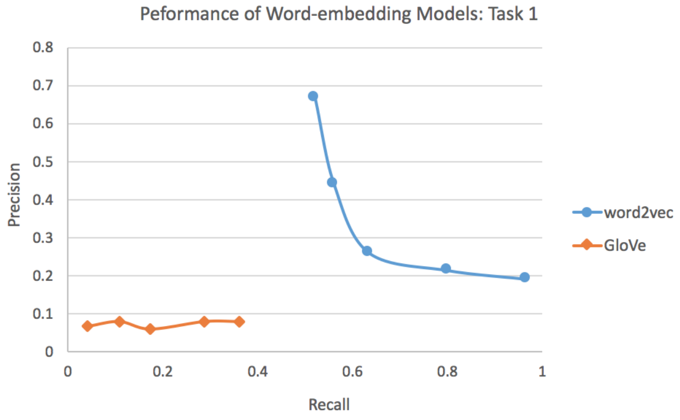


Figure 4.4: Average Precision and Recall @ {3, 5, 10, 15, 20} obtained for Task 1 (correspondence to dots from left to right).

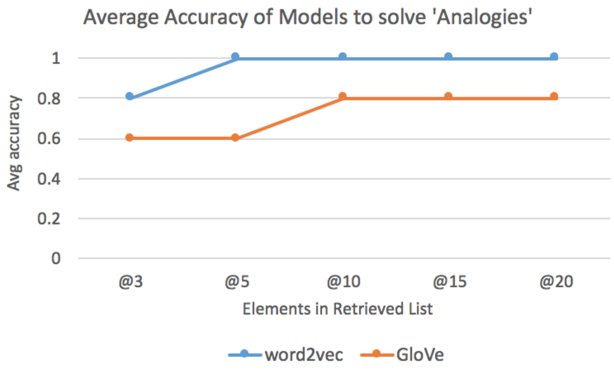
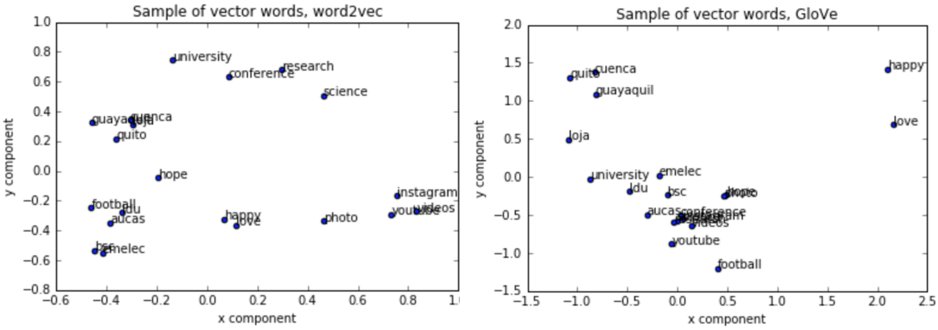


Figure 4.5: Average Accuracy @ {3, 5, 10, 15, 20} obtained for Task 2.

Indeed, similar words are found nearby and this reasoning is better supported by the bahaviour of word2vec (Figure 4.6a).

By the results presented above, the model chosen to continue with our work was *word2vec*.



(a) Words modeled with word2vec

(b) Words modeled with GloVe

Figure 4.6: Visual Comparison of the Behaviour of Word Embedding Models.

Identification of Clusters and Approach Implementation

Word2vec creates clusters of semantically related words in the provided corpus; therefore, the cluster assignment for words in the context of politics or other contexts like sports or news can be efficiently done. To group words, word2vec has as part of its Python implementation a *clustering* fuction. It needs as input parameters the corpus and the desired number of clusters. Once the clusters are defined with the model words grouped according so, a technique to get a representative vector for each of the clusters is to calculate their centroids[10].

Those vectors were grouped into 5 clusters. Each of the clusters were identified by an *id number*, where *2* was the id for the political-related cluster. It contained 2,326 words. Having the clusters' words, we were able to calculate their centroids c. By analyzing *cluster 2* that one may inspect its quality, among the words closest to its centroid c_p, we have 'illegally', 'impartiality', 'responsibilities', 'populism','fromGuayaquiltothecountry', 'bankers','pseudo','build', 'bankholiday', and 'spokesmen'.

[10] A cluster centroid is the mean vector of the observations in that cluster. The Python *scipy.cluster.vq* library may be used to find the centroid of a cluster.

With the centroids defined, the tweets of the Ecuadorian users (Section 4.4.1) were modeled into vectors to later measure its s_{tc}. For instance, showing an example we illustrate the job of Equation 4.1 (tweet-clusters similarity calculation):

- *Original Tweet t:* ‘Soplan vientos de cambios para el pas ..’
- *Original Tweet Translation:* ‘Winds of change blow for the country ..’
- *Preprocessed Tweet:* ‘soplan vientos cambios pais’
- *Preprocessed Tweet Translation:* ‘winds change blow country’
- s_{tc} *similarity values*: [0.594, 0.544, *0.881*, 0.73, 0.668] where metric in position *2* (list with range from 0 to 4) represents s_{tc_p}. Then, from the results we can infer that the tweet talks about politics.

Examples of the tweets with the largest s_{tc_p} are *i.* The disastrous Secretariat of Communications ask that the protesters show their faces. We ask them to remove the masks of civilized democrats they have; *ii.* He speaks against private ports. Make them pay more taxes. It has legitimacy. But do not say that they compete with the ports of the state; and *iii.* We demand professionalism, objectivity, impartiality. Practice of journalism as it was taught in the classrooms. Having s_{tc_p} calculations, we find the $DoIP_u$ established in Equation 4.2, Section 4.3.2, for each of the sampled users. Getting the users' $DoIP$ is our main contribution.

DoIP of a Twitter User vs their Friends' and Lists' DoIP

It is well known that friends share some similar interests [67] and that the implicit preferences of users may be detected by observing the content they consumed in the past [9]. These assumptions were verified by finding *a.* the correlation between *digital citizens*' $DoIP$ and their friends' and *b.* the correlation between *digital citizens*' $DoIP$ and their lists' $DoIP$ (for

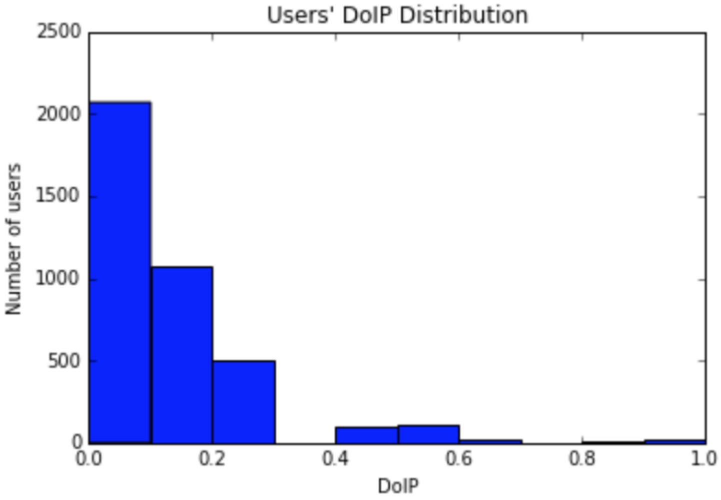


Figure 4.7: Histogram for the Users' DoIP Distribution.

subscription and membership lists). To better understand this, we separated the *digital citizens* into 4 groups: *i)* those whose $DoIP$ was equal to 0, *ii)* those whose $DoIP$ was bigger than 0 but less or equal to 0.3, *iii)* those whose $DoIP$ was bigger than 0.3 but less or equal to 0.7, and *iv)* those whose $DoIP$ was bigger than 0.7. The chosen ranges are delimited based on the DoIP distribution seen for the users who were analyzed (Figure 4.7). Next, the friends of the users of a given group categorised by the DoIP values were put together. Then, the friends' $DoIPs$ were calculated. Figure 4.8 presents the results of the correlation between the DoIP of the citizens and the DoIP of their friends. The plot shows that citizens in the *iv group*, whose DoIP is bigger than 0.7, tend to be friends with other users whose average $DoIP$ is 0.35, which is not the case for the users not interested in politics (*i and ii groups*), whose friends' average DoIP is less than 0.05. The same process was applied to do the analysis of Twitter lists. Therefore, according to their DoIPs, we created the four groups of users who were subscribed in at least one list. Then, their lists were placed in the users group.

Finally, we extracted the lists' DoIP to see the distribution. As we

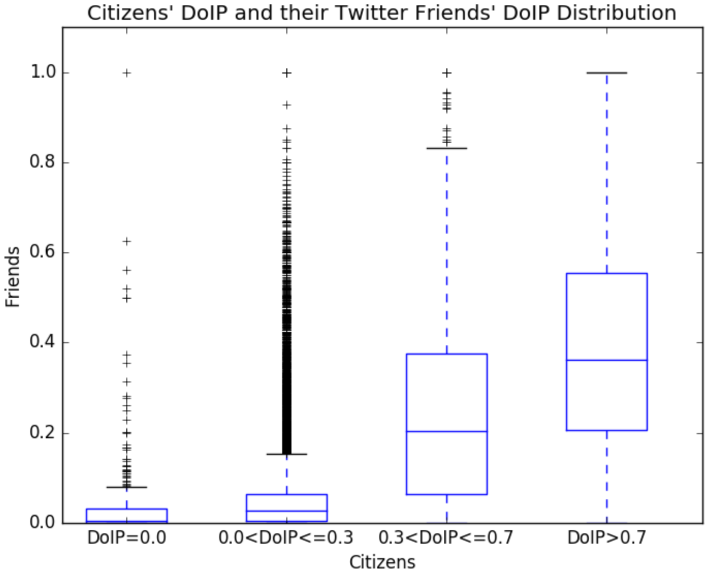


Figure 4.8: Digital citizens' DoIP and their friend' DoIP correlation. Users whose DoIP is bigger than 0.7, tend to be friends with other users whose average DoIP is 0.35.

did these steps for both subscription lists and membership lists the results are presented in Figures 4.9 and 4.10 respectively. Concerning 'subscription', it is shown that people who is not interested in politics subscribe to content that is not related to politics either. This behavior is different from the users whose DoIP is high (> 0.7), because they seem to be interested in accessing other sources where there may find posts that talk about politics. Similarly, regarding lists where the users are 'members', we see the same patterns. For example, users with DoIP equal to 0 tend to be added in lists where the rest of members do not talk about politics either. Meanwhile, users who post political-related content tend to be added in lists with the aim of collecting tweets discussing about politics. In other words, the tweets of the users highly interested in politics are gathered in lists whose content is usually 40% about politics.

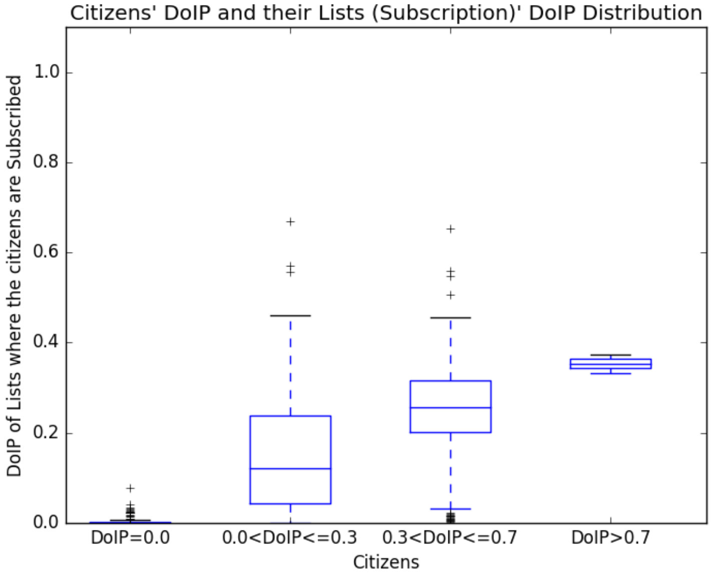


Figure 4.9: Correlation of the Digital citizens' DoIP and the DoIP of the Lists where they are subscribed.

The reasoning behind the categorization of citizens is to be able to infer the general preferences of similar users (those in the same category) instead of analyzing them individually. It is significant for our study to separate the citizens who are not interested in politics (*i*) from those who are more involved (*iv*) and differentiate the tendency of their friends with respect to politics. The ranges chosen for groups *ii* and *iii* divide into two the users with a DoIP different from 0 but less than 0.7. This shows that the difference concerning their friends' DoIP is big comparing both groups. The results presented in this section may guide the design of following back recommendations and lists to subscribe/create recommendations. For instance, if the political profile of the target user indicates a DoIP of 0.71 and the system detects a new follower of his/her account, it may follow the rule of recommending the follower to be followed back if his DoIP is bigger than 0.35; if the follower's DoIP is bigger than 0.2 but

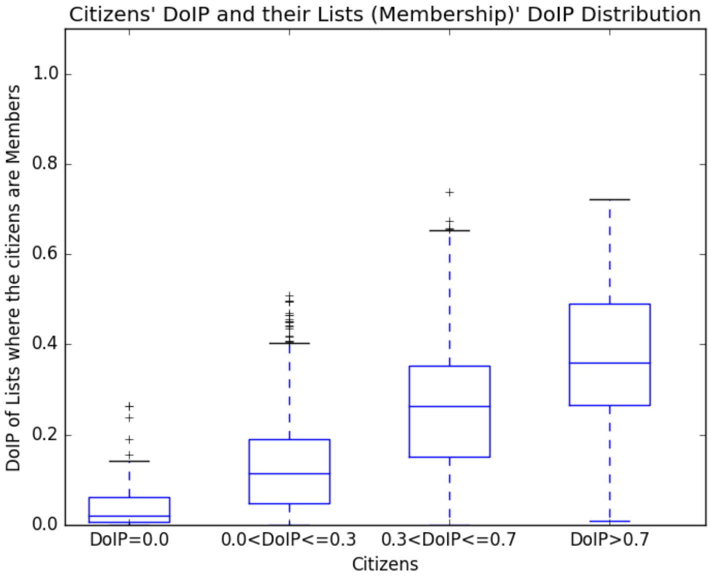


Figure 4.10: Correlation of the Digital citizens' DoIP and the DoIP of the Lists where they are members.

less than 0.35 other elements like number of friends in common may be considered to decide whether recommend or not; however, if the DoIP is less than 0.2 the follower is not suitable to be followed back.

4.5. Discussion

Community-featured technologies highlight communication and collaboration among citizens and nowadays they are essential to establish *digital citizens* interactions. Therefore, it is important to provide support to Twitter users about what is relevant and meets their information needs. For example, users may be suggested whether it is relevant to follow back a user or not.

In this chapter, we proposed an approach that can be used to identify the degree of interest in politics of a given user based on the semantic analysis of their tweets.

After evaluating the performance of two models at learning word embeddings, we chose to work with *word2vec* which was more appropriate than *GloVe* given our scenario. We observed that the quantity of vocabulary included in the model may affect (for better or worse) its performance. For instance, unlike *GloVe*, *word2vec* allows to restrict the selection of terms to be modeled by specifying a minimum number of times the term should appear in the training courpus. This reduces the number of words in the trained model avoiding the least frequent. In terms of our evaluation and considering that the proposed approach is based on how well the clusters of words are defined, we decided to build the model with *word2vec* (linguistic regularities were more precise). However, we claim that, depending on the methods and research goals, it may be needed a model which is able to represent *all* the words in the training dataset. Then, for different scenarios, a variety of experiments for comparison could be done.

In our approach, *word2Vec* was employed to extract the words associated to politics (vector model and clustering) and give them a vector representation. Our results show that:

- Users with high level of interest in politics tend to be friends with other users who are interested in politics as well.
- Users with high level of interest in politics seem to be in touch with additional posts related to politics through their subscription to Twitter lists.
- Lists tend to be created by aggregating members who convey like-minded ideas. Then, a political figure's tweets may be found in a list that groups other politicians.

The experiments allowed us to determine thresholds that may be considered when designing a recommender systems for tweeters interested in politics. The thresholds identify, for example, if a follower is or not relevant to become a followee depending on his/her DoIP and the DoIP of the user to be recommended. The method proposed may be the base for defining following back and lists recommendation strategies.

The context of our research was bounded by the political environment in Ecuador at the time of data collection. However, we consider that with a well defined training corpus that covers tweets related to technology, sports, culture, social events, among other categories, the proposed approach is able to be used without modifications to train a model and extract the degree of interest of a given user in every of the aforementioned contexts. Hence, our method would work by quantifying the amount of user's tweets that correspond to every category.

Chapter 5

IDENTIFYING POLITICAL INTEREST GROUPS AND HASHTAG HIJACKERS

In this chapter we propose a framework for identifying political interest groups as well as possible hashtag hijackers. Specifically, this work focuses on the problem of giving recommendations to groups in which a group of users with the same political view receives suggestions of users they should not follow because they have opposing political views but use hijacked hashtags. Experiments on real-world data collected from a series of demonstrations in Ecuador show the effectiveness of this approach in automatically identifying hijackers so that they can be effectively recommended to a group as people they should not follow. The generic application of this approach in topics that generate debate where there are user groups with opposing views (such as life imprisonment supporters/opponents, immigrants' rights supporters/opponents, among others) would allow their detection, as well as the identification of the groups' characterizing hashtags.

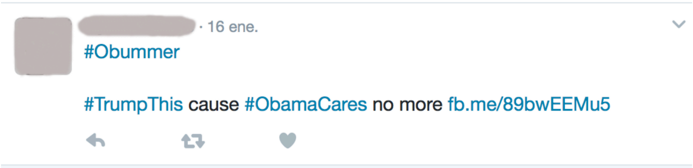
5.1. Research Problem

The use of social networks for social participation and social mobilization of offline demonstrations is having a big impact on society. Researchers have found evidence that social media affects citizens' political participation [68, 69]. Various offline demonstrations have been studied. An example is illustrated in the work of Segerberg and Lance Bennett [70]. The authors focus on Twitter and the hashtags #TheWave and #cop15 used in the protests around the 2009 United Nations Climate Summit in Copenhagen. In the same way, other citizen movements have used new communication technologies to spread their thinking and mobilize citizens. On the other hand, governments threatened by the spread of massive demonstrations are making the decision to censor Internet access to some of the most popular social networks [71, 72, 73].

One of the problems with using social media platforms for political participation is the existence of *hijacked hashtags*. A hashtag is considered as hijacked when it is used for a purpose different from the original one. More specifically, in the political context, the use of hijacked hashtags allows users to reach their political opponents by using hashtags that are familiar to them. For example, suppose that on Twitter, a user wants to know what others are saying about a specific topic and clicks on a hashtag of interest. The platform would display all tweets that contain that hashtag, and the user might see content generated for the purpose of starting arguments and generating confusion.

To illustrate this, consider the hashtag *#obamacares*, that emerged some time ago with the appearance of the Medicare program created by Obama's government in the U.S. and as an evolution of its trending hashtag #obamacare. At the end of 2016, most of his supporters and followers were saying goodbye to the outgoing president by using the hashtag *#obamacares*. However, Figure 5.1 shows an example of the mentioned hashtag and how it was misused.

Being able to analyze and contextualize the content of the messages posted in social media and identify users who hijack hashtags (*hijackers*) is of central interest. Indeed, starting from the assumption that in political



[redacted] · 16 ene.
#Obummer

#TrumpThis cause #ObamaCares no more fb.me/89bwEEMu5

Figure 5.1: Hijacked Hashtag in the context of Politics.

events active users can be associated to political parties, being able to automatically identify which party a user is associated with, and the political party that usually employs specific hashtags, would enable the detection of the hijackers. Moreover, identifying the hijackers would make it possible to give group recommendations to the other political party, in the form of *who you should not follow* lists. Indeed, alerting a group of users who belong to one political party of those with opposing views that hijack their hashtags, would allow them to take actions and avoid getting into pointless discussions or getting confused by false/twisted information.

In this chapter, we discuss the use of Twitter for the demonstrations in Ecuador that took place in the years of 2015 and 2016. We consider Ecuador as a case study because it has a long history of demonstrations against different governments, which makes this scenario very attractive for more in depth study. Besides, the analyzed demonstrations present the same patterns, *i.e.*, they were given at different periods of time in similar scenarios in which there were two groups of participants: government supporters and the opposition.

We present a framework for finding users who belong to each of the groups as well as a method to identify possible hashtag hijackers. The purpose of this study is to focus on a real-world application of the framework presented and create *who you should not follow* group recommendation lists, by identifying the political parties of both the targeted group of users and hijackers. More formally, the problem is stated as follows:

Let e be an event (or a series of related events that occur on different days for the same purpose). Let $T = \{t_1, \ldots, t_N\}$ and $H = \{h_1, \ldots, h_M\}$ denote the sets of tweets and hashtags associated with that event. It is denoted as $U = \{u_1, \ldots, u_J\}$, the set of users who wrote those tweets,

$T_u \subseteq T$, the tweets written by the user $u \in U$, and $H_u \subseteq H$ the hashtags employed by the user. Let $C = \{c_1, \ldots, c_K\}$ denote the classes of users involved in the event (*e.g.*, in this domain, C denotes the two classes of citizens' political orientations, namely government supporters and opponents). The first objective is to classify each hashtag, $h \in H$, in order to associate it to the party (or parties) that use it; let $H_k \subseteq H$ denote the set of hashtags employed by a class of users, c_k.

The second objective is to determine whether the hashtags and tweets of each user are hijacked based on the semantic content of the tweets in which the hashtags are used. The third objective is to assign a score s_{uk} that contains the relevance of a class c_k for a user u and to perform an automatic classification of the users in order to associate them to the class they belong to (*i.e.*, that with the highest score); it is denoted as $U_k \subseteq U$, the set of users who belong to a class of users, c_k. Given a class c_q to which a user u does not belong (*i.e.*, $u \notin U_q$), the final goal of our work is to create a score x_{uq}, which indicates the user's tendency to employ hashtags normally adopted by the users in U_q. The obtained values of x_{uq} (one for each user who does not belong to U_q) will be ranked in descending order to recommend to the users in U_q who they should not follow.

The scientific contributions in this chapter are:

- a semi-supervised method of classifying hashtags based on the context in which they appear;
- an approach to automatically detect groups of interest in a given event, based on a semantic analysis of the tweets they post;
- an automatic approach to classify each of a user's hashtags and tweets as being either hijacked or not based on the group of interest to which the user belongs and the content of the tweets;
- an approach that recommends to a group of users those they should not follow, based on a score that indicates how likely it is that the user is a hijacker; and

- the evaluation of this approach on real-world Twitter data using both quantitative and qualitative experiments that validate the effectiveness of this proposal.

- The work done in this chapter was published in Lorena Recalde, Jonathan Mendieta, Ludovico Boratto, Luis Terán, Carmen Vaca, Gabriela Baquerizo. "Who You Should Not Follow: Extracting Word Embeddings from Tweets to Identify Groups of Interest and Hijackers in Demonstrations", 2017, in IEEE Transactions on Emerging Topics in Computing, Issue: 99.

This chapter is structured as follows. First, Section 5.2 presents background information and related work. In Section 5.3, we describe the different algorithms used in this work. Then, in Section 5.4, we detail the case study of the demonstrations in Ecuador that are used in this research and the experimental framework that was developed to validate our proposal. Section 5.5 presents further work and other application areas for the framework. Finally, we discuss particular implications related to our research in Section 5.6.

5.2. Related Work

In this section, we describe different research studies and projects related to social networks and social effects, the use of Twitter for political participation, and hashtag hijackers.

The use of social networks for political participation and demonstrations is a new and evolving phenomenon occurring in events such as the Middle East and North Africa, also know as the *Arab Spring* [74]. It plays a key role in organizing and diffusing public protests and demonstrations. Some works have focused on the structures of social networks and/or information propagation given a politically defined hashtag or group of hashtags [75, 76]. The academic literature also contains a number of case studies related to the adoption of these alternative communication channels to promote mobilizations and demonstrations around the globe

[70, 77, 78]. In their work, Varnali and Gorgulu [79] analyze the case of the Gezi Park protests (#OccupyGezi) from the context of social influence for political participation in Twitter. In the same way, Ramirez and Guilleumas[80] present a quantitative research in which they analyze the trends of tweets related to the 15M movement (May 15, 2011). In [81], the authors analyze the citizens' communication through hashtags during protests that took place in Brazil in June 2013 and describe the different kinds of political hashtags (emotive, conative, and meta-lingual). The researchers found that hashtags containing information about streets and other places for demonstrations (referential hashtags) characterize a distinct behavior in the dataset.

One of the problems that activists are facing is known as hijacking. In the sphere of social networks such as Twitter, there are different types of hijacking, the most common being *account hijacking* [82, 83, 84] and *hashtag hijacking* [85, 86, 87, 88, 89]. The former is a process through which a user account is stolen or hijacked by a hacker. The latter is a research domain that is catching the attention of the academic community, and refers to the misuse of hashtags for a different purpose that the one that was originally set, generating confusion to users interested in the topic associated to the original hashtag.

Group recommender systems (GRS) are used when the context of the recommendation is defined by a group of users who are being suggested items that can be of interest for the group as a whole (*e.g.* a restaurant to dine together, a movie to watch, etc.), instead of having the scenario of a single user to recommend. GRS aggregate information from individual user models [90] in a way that the needs of all the members of the group are equally satisfied. Compared to the current literature in GRS that promotes items *adoption*, the presented work is the first attempt in recommending users *not* to follow.

This work focuses on the case of *hashtag hijacking* and develops a framework for determining tweeters' political stances and the identities of users who are prone to hijacking hashtags on Twitter, and to be displayed as a *who you should not follow* list to the users who normally employ those hashtags. As this analysis of the literature has shown, no other approach that detects hijacked hashtags was ever applied to the recommendation domain, no study that analyzed the demonstrations under the social media perspective ever provided services and recommendations to the participants in these demonstrations, and no GRS was developed to recommend users to a group instead of items to consume together. All these elements give novelty to our proposal in multiple research areas, by developing a unique solution that aims at facilitating the experience of the users on Twitter, for topics that are of strong interest for them (i.e., those that are related to the demonstration in which they are participating). To achieve this goal, a set of steps should be performed, which includes: tweet preprocessing, text modeling, hashtag classification, tweet political tendency detection, user political tendency detection, and group recommendation. These steps are presented in more details in the next section.

5.3. Approach

This section describes the approach used to automatically detect groups of interest in political events and give users recommendations as to who they should not follow, based on who hijacks hashtags they normally use. The approach works in six steps:

1. **Tweet preprocessing.** The content of the tweets is preprocessed in order to remove characters that would make it hard to process the text for knowledge extraction purposes.

2. **Text modeling.** The corpus of preprocessed tweets is employed to build a model with the *word2vec* tool, developed by Mikolov *et al.* [37]. *Word2vec* returns a vector representation of each word that is later used to classify hashtags, tweets, and users in the remaining steps.

3. **Hashtag classification.** Based on a ground truth available for a subset of hashtags, the word embeddings are used to classify the remaining hashtags and associate them to the group of users that employs them.

4. **Tweet political tendency detection.** Based on the hashtags contained in a tweet and the classification performed in the previous step, in this phase we detect a score for the tweet of interest, which represents the extent to which the tweet contains hashtags employed by the group to which the author of the tweet belongs.

5. **User political tendency detection.** The user's interest group is identified on the basis of the tendency scores of each of the user's tweets.

6. **Group recommendation.** For each group of interest, a subset of users they should not follow is ranked and presented, based on a score that indicates how likely it is that the users on the list hijack group hashtags.

What follows is a systematic account of how the tasks have been implemented.

5.3.1. Tweet Preprocessing

Most on-line social networks (OSNs) give users the freedom to express their ideas in an informal way. For example, on Twitter, users can enrich posts using hashtags, URLs, mentions, and locations. However, the use of emoticons, sequences of special characters, and language-dependent abbreviations make machine-learning tasks for understanding and processing text harder. In order to clean up the text in the tweets, the following preprocessing steps [91] were executed for each tweet:

1. Capital and special letters are changed (using the Unidecode library[1]). The library transforms a tweet into a sequence of lowercase letters and changes letters with accents and special characters, such as *'ñ', 'á', 'é', 'í'* that are often used in the Spanish language. These letters will take the form of 'n', 'a','e', 'i', respectively.

[1] https://pypi.python.org/pypi/Unidecode

2. A so-called tokenization is executed (using the TweetTokenizer of the NLTK Python library[2]). It keeps the hashtags, mentions, and URLs. Additionally, it removes the rest of the special characters and presents a list of tokens per tweet.

3. Spanish stop words are removed from the list of tokens. The list of Spanish stop words that get removed is provided by the Python NLTK library as well. However, an extended version of the stop words list was created through the addition of the punctuation characters presented in the punctuation module of the Python String library. For Spanish punctuation, characters such as ‘’, ‘’ and abbreviations like ‘q’, ‘xq’ and ‘d’ are treated as stop words, too.

4. Stemming is done using the *snowball* method[3] [92] for all the tokens in the tweet that are not hashtags, mentions, and URLs.

5.3.2. Text Modeling

The preprocessed tweets will be the input used to create a model of the words in them. In this study, a neural language model is used, since it is able to provide a better representation of the words with respect to classical models [65]. Indeed, neural language models extract the syntactic information in a text corpus instead of a simple bag-of-words and perform some nonlinear transformations. Moreover, they generate a low dimensional vector space in which semantically similar words are close. In this study, we employed the so-called neural word embeddings and built them using Google's *word2vec*[4], which is the most widely used implementation in the current literature.

In order to compute the vector representations of words, the *word2vec* tool provides two implementations, known as *continuous bag-of-words (CBOW)* and *skip-gram*. The skip-gram architecture uses the current word to predict the surrounding window of context words, so it builds the model

[2] `http://www.nltk.org/api/nltk.tokenize.html`

[3] `http://www.nltk.org/api/nltk.stem.html`

[4] https://code.google.com/archive/p/word2vec/

more slowly but does a better job of representing infrequent words. For this reason and validations already motivated in Chapter 4, we used the skip-gram architecture in this study. In building the model, every tweet is treated as a sentence. The *word2vec* tool also requires the same parameters as inputs which are detailed in Chapter 4, Section 4.3.

The values that these parameters took in this study are reported in the experimental framework, to ensure the repeatability of the experiments. The output of this step is a vector for each word in the corpus.

5.3.3. Hashtag Classification

The objective of this task is to provide a classification of each hashtag so that it can be associated with a class c_k that represents a group of interest. This task is semi-supervised in the sense that it is necessary to start from a ground truth that associates a small subset of hashtags with a group of interest (e.g., with the help of a domain engineer). For the rest of the hashtags, their embeddings (*i.e.*, the vector representations generated in the previous step) are used in order to classify them. All the hashtags in the ground truth are associated with a class c_k and added to a set H_k. The hashtags classified using the ground truth are denoted by H^G.

Since each hashtag is represented as a vector, the most straightforward form of comparison (which is also the most employed in the literature) is to calculate the angle between them using the cosine similarity. This step calculates the cosine similarity between each hashtag $h \in H \setminus H^G$ that has yet to be classified and each hashtag $h' \in H^G$ that has already been classified. If the similarity between the two vectors is greater than or equal to 0.5, h is added to the set H_k in which h' is included. The output of this step is the classification of each hashtag in the dataset, which are each associated with one of the groups of interest involved in the event.

5.3.4. Tweet Political Tendency Detection

This step measures the extent to which a tweet can be associated with a group of interest (this metric is called the *tweet political tendency*). Given a tweet $t \in T$, let H_t denote the set of hashtags it contains. For each class c_k, a set H'_k is built that contains all the hashtags employed in the tweet that are associated to a group of interest c_k:

$$H'_k = H_t \cap H_k$$

Since the vectors handled are linear, it is common to sum several vectors to obtain a unique model representation (additive compositionality property) [37].

Therefore, in order to create a model of the hashtags in a tweet that belongs to a class c_k (denoted as h'_k), all the vectors in H'_k are summed up as follows:

$$h'_k = h'_k + h_k, \forall h_k \in H'_k$$

The model is then used to evaluate whether the content of the tweet is related to the hashtags of a class c_k that the user has employed in the tweet. In other words, the objective is to understand whether a user has misused the hashtags of a class c_k in the context of a tweet (*i.e.*, whether this is a tweet that contains hijacked hashtags).

In order to understand whether the tweet contains hijacked hashtags, it is first necessary to model its content. Let W_t denote the words that are not hashtags in the considered tweet t. By taking their embeddings into consideration, one can build a unique model of the words that appear in the tweet that are not hashtags (denoted as w'_t) as follows:

$$w'_t = w'_t + w_t, \forall w_t \in W_t$$

Given the words in the tweet (modeled as the vector w'_t) and the hashtags related to each group of interest c_k (modeled as the vector h'_k), the task evaluates how similar the contexts in which the words and hashtags used in the tweet are by calculating the cosine similarity between the two vectors. The result is a tweet political tendency score, denoted as s_{tk} (*i.e.*, the tendency of a tweet t with respect to the hashtags employed by the group of interest c_k).

$$s_{tk} = cos(w'_t, h'_k), \forall c_k \in C$$

The range of values for the cosine similarity is between -1 (dissimilar items) and 1 (similar items). The intuition behind this operation is that if a hashtag has been used in the tweet in a semantic context similar to that of the other tweets in which it has been used, the political tendency score of the tweet with respect to the class c_k is high. Otherwise, it is low.

5.3.5. User Political Tendency Detection

The users in the dataset are those who posted at least one tweet containing one or more hashtags identified as meaningful to the event. Based on all the tweets a user posted (denoted as T_u), the user's political tendency score for each class c_k can be calculated as follows:

$$s_{uk} = \frac{\sum_{t \in T_u} s_{tk}}{|T_u|}$$

In other words, the political tendency score of a user for a class c_k is the average of the tendency scores of all the tweets the user posted that contain hashtags associated with c_k. Of course, if the score is high, the user has made proper use of the hashtags related to that class in the tweets. At the end of the process, the user is associated with the group of interest that corresponds to the highest s_{uk}:

$$u \in U_k \ iff \ \nexists c_q \neq c_k \ s.t. \ s_{uq} > s_{uk}$$

5.3.6. Group Recommendation

Once a user is associated with a group of interest, it is possible to evaluate whether a user has hijacked the hashtags of other groups based on that user's tweets. If a user has made heavy use of hijacked hashtags, the group whose hashtags are being hijacked should be aware of the presence of this user's tweets. Therefore, for each group to which the user does not belong, it is important to evaluate how much the user hijacked the group's hashtags. This is done by using a score x_{uq}, which indicates the tendency of a user $u \in U_k$ to employ hashtags that are normally adopted by users in another group U_q.

Let T_{uq} be the set of tweets posted by a user u that contain hashtags associated with a class c_q to which the user does not belong. In order to evaluate the tendency of the user to hijack hashtags of class c_q, the previously defined score s_{uk}, which defines the tendency of a user to employ hashtags used by her group, is also considered (*i.e.*, we consider how much the user embraces the ideology of the group to which she belongs).

The score is calculated as follows:

$$x_{uq} = \frac{|T_{uq}| * s_{uk}}{|T_u|}$$

This score represents the percentage of hijacked tweets posted by the user, multiplied by the user's political tendency score. The higher the value of x_{uq}, the more likely it is that the users who belong to U_q should not follow the hijacker. Therefore, for each user $u \notin U_q$, the values of x_{uq} are ranked in descending order and presented as a recommendation to the group of users U_q, indicating who should not be followed.

5.4. Experimental Framework

This section presents the experimental framework developed to validate this proposal. In this section, we present a real-world case study related to a set of demonstrations that happened in Ecuador between 2015 and 2016. This work is based on the datasets collected in this study. We then describe data collection followed by a description of the approach used as a baseline for the comparison. Then the experimental setup and strategy are presented, and the section will end with a presentation and a discussion of the results.

5.4.1. Case Study: Demonstrations in Ecuador

Mobilizations are self-summoned by citizens through social networks and mobile phones. They are citizen protests that bring to mind the Arab Spring, social movements in Iceland, Tunisia, Syria, and Egypt, *Indignados* in Spain, and the Venezuela strike in Latin America.

Public and political debate in Ecuador is supported by the idea that social protests themselves are capable of overthrowing governments, as was the case in the dismissal of former Ecuadorian presidents Bucaram, Mahuad, and Gutierrez, which occurred because of mobilizations and demonstrations. From 1996 to 2006, Ecuador had eight different presidents [93], which shows that demonstrations in Ecuador had a real impact. These public manifestations are a form of democracy understood as the occupation of public spaces [94], which does not resemble protests organized by social institutions or political movements. According to De la Torre [94], these encounters differed from others in terms of organization. Given Ecuador's long history of demonstrations against different governments, this scenario is very attractive for more in-depth study.

In recent years, social networks had a big impact on the organizations of demonstrations worldwide. In the case of Ecuador, most of the latest mobilizations and demonstrations were organized by citizens via social networks, such as Twitter, Facebook, and Whatsapp, among others. The latest demonstrations happened by determination of Ecuadorians who took to the streets in protest of the government of the president at that time, Rafael Correa, which happened on June 8, 2015. However, everything began on May 24, 2015 during the national report submitted by President Correa to the National Assembly, where President Correa presented two bills on taxes that would be sent on an urgent basis to be reviewed and approved or disapproved within thirty days. The bills were related to an increment in the inheritance tax and the capital gains tax; then, among the population, mainly middle class, there was the common idea that they would be economically disadvantaged.

The political impact of these bills led to the mobilization of civil society in major cities: Guayaquil, Quito, and Cuenca. According to different media outlets in Ecuador, nearly 20,000 demonstrators participated. These mobilizations were different to protests that took place in 2014 because of their permanence in time[95], in such a way that the volume of protesters increased everyday. Besides, these strikes "were not demonstrations called by specific political actors but born and spread via social networks"[96]. Political debate in the Ecuadorian public sphere has a visible presence on social networks because users take advantage of hashtags to connect with people with the same interests and the trending topics on Twitter allow them to mobilize around the same cause. This situation sometimes enables users to hijack popular hashtags in order to distribute their views to other users [86].

Ecuador is a country where the penetration rate of micro blogging platforms has grown in recent years. The use of Twitter for political debate, for instance, has increased in response to the communication strategy of the current government, whose leader has 2.7 million followers on the platform at the time of writing [97, 98, 99]. This is also reflected in the call made to march August 2015, in which both supporters and government opponents used their virtual audiences to mobilize citizens.

In this work, we analyze the marches that took place on March 19, June 25, July 2, and November 11, 2015 and April 7, 2016. A day of national resistance to claim the economic measures of the regime was convened on March 19, 2015, and it generated massive protests in at least eight cities. The largest concentrations were in the cities of Guayaquil, Quito, and Cuenca. On June 25, three opposition marches and a government concentration took place in Quito. Five months later, a new national mobilization called by the Unit of Indigenous Social Movements and National Collective Workers happened on November 11 to push the government to file drafts of constitutional amendments. In response to a tweet with the phrase "I protest", on April 7, 2016, civilians took to the streets without regard for age, race, religion, or political affiliation to express their displeasure. The aim of the current study is to determine users' political tendencies–either opposition or support–as well as their hijacked hashtags in these demonstrations.

5.4.2. Data Collection

The datasets used in this work are extracted from a collection of tweets published on the dates of the strikes: March 19, Jun 25, July 2, November 11 (2015), and April 7 (2016). The bag of words formed based on the list of hashtags used in those tweets was analyzed (according to each hashtag and its frequency). From these datasets, 2,378 hashtags were found, but only 730 had a frequency higher than 1. The 730 hashtags were manually checked to verify their relation to the strikes. Then a list of 308 hashtags meaningful to the strikes was filtered. Additionally, using the list of strike hashtags, a new search of the tweets was performed. The number of tweets collected was 126,667, while the number of unique users was 18,960. These datasets were used to identify the political tendencies of those users who published tweets during the five sociopolitical events, particularly the strikes for and against the current government of Rafael Correa that occurred in Ecuador in 2015 and 2016. For future studies, the dataset used in this work is available online[5]. As shown in Section 5.3, k classes to which the users belong are identified. It should be clear that, in this scenario, there are only two classes to which the users can belong, i.e., *supporters* and *opponents*. For the sake of simplicity, instead of the generic c_k notation, the more intuitive S and O notations are employed to help identify specific classes of users.

5.4.3. Baseline Approach

In the research presented in [86], the authors propose a method to detect tweets that contain hijacked hashtags. It stands on the use of one of the core information retrieval techniques, *TF-IDF* [42], which reveals the ground nature of the approach. *TF-IDF* provides a score that quantifies how *relevant* a word is to a document in a corpus. The presented method consists of a manual classification of a sample of hashtags into five categories: Technology, Entertainment, Politics, Brands, and Others.

Then, an analysis is done taking into consideration each hashtag by extracting one thousand tweets that contain the given hashtag. Once the tweets are collected, in order to apply *TF-IDF*, a tweet is seen as a document and the set of tweets shapes the corpus. Consequently, a list of meaningful words (those with highest *TF-IDF*) that represent each of the categories may be obtained. Later, when a tweet needs to be evaluated to see whether it is hijacking a hashtag of a particular category or not, a

[5] https://github.com/lore10/Who_you_should_not_follow_Datasets

weight is calculated by counting the number of the tweet's words that appear in the category list of meaningful words. In other words, a value of 1 is given per tweet's word found in the category list. Low scoring tweets, those with a weight of 0 or 1, are treated as suspicious or irrelevant.

The addressed problem in the baseline approach is the detection of tweets that most likely include hijacked hashtags. To be reproduced and implemented considering the context and datasets of our method, *"Who you should not follow"*, the baseline was adapted with regard to the categories (*i.e.*, we divided the hashtags into supporters and opponents) and the analysis of hashtag hijacking, that was oriented towards *users* instead of *tweets*. To do so, the *evaluated tweet* in [86] is replaced by the aggregation of the set of tweets posted by the *evaluated user*. Accordingly, the user, compounded by his/her tweets, is seen as a document. The score to measure how relevant the user is to the category is calculated by counting the number of the user's words that appear in the category list of meaningful words. A user is considered as a non-hijacker when his/her total score is bigger than 5. More detailed about the results obtained when applying the baseline method using our datasets is given in Section 5.4.5.

5.4.4. Experimental Setup and Strategy

The framework used in this work involves the use of the *word2vec* approach. In order to generate the model based on the textual data, the tool was run with the following parameters:

- $size$ is set to 100;
- $window$ is set to 5;
- $sample$ is set to 0;
- $negative$ is set to 0;
- min_count is set to 5;
- $alpha$ is set to 0.025.

Query	Similar Word Retrieved	Similarity 100 dim	Similarity 300 dim
#correa	correa	0.680	0.677
#getoutcorrea	#getoutcorreagetout	0.859	0.851
#revolution	#ecuadoriansrevolution	0.848	0.844
#rafaelcorrea	#citizensrevolution	0.727	0.713
#inheritancelaw	#inheritancetax	0.831	0.827

Table 5.1: Comparison of 100dim Model and 300dim Model.

We worked with the default values for the parameters except for $size$.Tests considering $size = 100$ and $size = 300$ were implemented to define an optimal value for the parameter.

To see the behavior of the two models, we extracted the most similar term for a list of hashtags. For the hashtags provided, the closest word retrieved was the same in the two models. However, the similarity calculations varied, giving the highest similarities for $size = 100$. Table 5.1 shows some results where we present the *query or hashtag*, the *most similar word retrieved*, the *cosine similarity* for the model with 100 dimensions and the *cosine similarity* between the words for the model with 300 dimensions.

Since this work is based on real-world data and the “who you should not follow” perspective cannot be measured using classic metrics, quantitative and qualitative experiments are performed with the goal of validating this proposal by showing how this approach is able to classify the hashtags, tweets, and users as well as how it can highlight a group of those users who could actually be classified as hijackers. In order to validate this proposal, six sets of experiments are performed:

1. **Classification analysis.** In order to evaluate the choices made to build the classifications of the hashtags, the tweets, and the users made in steps 3, 4, and 5 of this proposal (Section 5.3), some interesting cases of outputs returned by the tasks are analyzed.

2. **Influence of the hijacked hashtags in a profile.** This experiment evaluates the percentage of tweets with hijacked hashtags in a user profile.

3. **Impact of the hijacked tweets in each group of interest.** The percentage of users in each of the two groups of interest who wrote tweets with hijacked hashtags is analyzed.

4. **Distribution of the user political tendency scores.** For each user, two user political tendency scores (supporting and opposing) are analyzed to directly evaluate the two types of behavior that characterize the user.

5. **Analysis of the group recommendations.** In this experiment, the recommendations that the system would make are evaluated by measuring the scores returned by the group recommendation tasks and how they are distributed among the two groups.

6. **Validation of the results.** In order to validate the results, in this set of experiments we calculate the average distance between the user and a hashtag to see how "cohesive" are the users who belong to a group with respect to the hashtag[6]. To calculate the distance between the user and a given hashtag, s/he was modeled by summing the vectors of the hashtag s/he used, to later being compared to the hashtag in analysis. The cosine distance will generate a value of 1 for the *most similar* user, and a value of -1 for the *least similar* one. The users that are classified as supporters/opponents are further from the hashtags used by their counterpart. The analysis will be performed for those who have been classified as hijackers or not.

[6] This strategy was inspired by the classic validation of unsupervised classification (clustering), in which the average distance from the observations to the cluster centroid is measured.

5.4.5. Experimental Results

Next, we detail the experiments used to do the qualitative and quantitative validation.

Classification Analysis

For the three types of *classifications* performed in this approach for each hashtag, tweet, and user, the qualitative results that show the outputs returned by the three steps is provided.

Hashtag classification. The ground truth of this approach is represented by a manual classification performed by a specialist (an Ecuadorian political scientist). The seeds given by this semi-supervised approach are 49 hashtags identified as opposition hashtags and 61 hashtags identified as support hashtags.

To show an example of a hashtag that this approach classified as belonging to the opponents due to these seeds, consider `#shirys`, which is the misspelled name of a place that cannot be classified as belonging to one group or the other without external sources of knowledge. However, the classification task returned a 0.81 similarity with one of the hashtags employed by the opponents, `#fueracorreafuera` ("get out, Correa, get out"). This tells us that the two hashtags have been used in similar contexts. Indeed, Shyris was the meeting point where the protests against the government started. Note that even though the word was misspelled in the hashtag, this approach was still able to correctly classify it.

Tweet classification. The tweets political tendency score is measured to define the tweet class. As an example, let's consider a tweet t:

> '@MashiRafael CARA DE TUCO!!Tienes que OBLIGAR a los empleados publicos a MARCHAR A LA FUERZA!!#FueraCorreaFuera http://t.co/7rNXvwza37'

Its translation is:

> '@MashiRafael IMPERTINENT!!You have to make the state employees gather together supporting you, against their will!! #GetoutCorreaGetout http://t.co/7rNXvwza37'

The following results were obtained:

$$\begin{aligned}
W_t &= [`@mashirafael', `car', `tuc', `oblig',\\
&\quad `emple', `public', `march', `fuerz',\\
&\quad `http://t.co/7rnxvwza37']\\
H'_S &= []\\
H'_O &= [`\#fueracorreafuera']\\
s_{tS} &= -1\\
s_{tO} &= 0.471
\end{aligned}$$

The words that give meaning to the tweet suggest confrontation (*CARA DE TUCO!*) and disagreement with the government, particularly with the president, who is mentioned by the use of his official Twitter account (*@MashiRafael*) and blamed for forcing the state employees to attend his calls. Thus, note that the tweet can be correctly classified as belonging to the opponent group ($s_{tO} > s_{tS}$). However, probably because of the presence of a link and other unusual words, the score is not that high (i.e., the hashtag has been used in a context slightly different from the usual one).

User classification. To validate the capability of this approach to classifying the users according to their *political tendency score*, let's consider a user who posted the following two tweets:

Tweet t_1

- t_1: 'Dos #MedallitaParaCorrea luchito ch. y galito ch.'

- t_1 translation: ‘Two #LittleMedalForCorrea luchito ch. and galito ch.’

- $s_{1S} = -1$

- $s_{1O} = 0.590.$

Tweet t_2

- t_2: ‘#MedallitaParaCorrea Porque si hubiera tenido moneda nacional nos hacia m....’

- t_2 translation: ‘#LittleMedalForCorrea because without the dollar as currency he would make us ....’

- $s_{2S} = -1$

- $s_{2O} = 0.423$

Based on the user’s previous tweets, the following computations are made:

$$s_{uS} = Avg(s_{1S}, s_{2S}) = -1$$
$$s_{uO} = Avg(s_{1O}, s_{2O}) = 0.507$$

Then, $u \in U_O$ (user tendency is ‘opponent’, since $s_{uO} > s_{uS}$). The example shown above reflects most of the data, which means that it is possible to detect the political orientation of users by analyzing the hashtags they employ. In other words, the political orientation obtained is not vague at all. That means, the users expressed their opinions about the government, and these are equivalent to the hashtags used.

However, the problem arises when a) some users see one trending hashtag and start using it in their tweets to make their posts visible or popular even though the content is not political; b) some users want tweeters with opposing views to be aware of their opinions; for instance, when those who are against the government tweet content criticizing it, but use hashtags proposed by government supporters and/or mention (@) them;

or c) in the last case, the hashtag that started being used by one political group evolved into one being used by the antagonist group as a result of a proper *evolution* or as *sarcasm*.

The following examples illustrate this effect:

Case a)

- Tweet: “Yo solo quiero ser popular. #FueraCorreaFuera”.
- Tweet translation: “I just want to be popular. #getoutCorreagetout”.

Case b)

- Tweet: “Que pena que haya gente que haga estos hashtags #YoCreoEnRafael se nota que lo que abunda en Ecuador son IGNORANTES”.
- Tweet translation: “It is a pity that there are people who post #IbeliveinRafael it is notorious that ignorance abound in Ecuador”.

Case c)

- Tweet: “por destruir el pais pero saber que ya te vas #felizcumplepresi #nosheabruto #ecuadorunidoenresistencia @cata_l_b_n @shababaty @friega_again”.
- Tweet translation: “To destroy the country but now knowing that you will leave #happybirthdaycorrea #dontbestupid #ecuadorunitedforresistance @cata_l_b_n @shababaty @friega_again”

Even though the problems associated to these three special cases occur in our data, due to the fact that word embeddings are able to capture the context in which a word appears, a clear classification of the users was possible thanks to our approach.

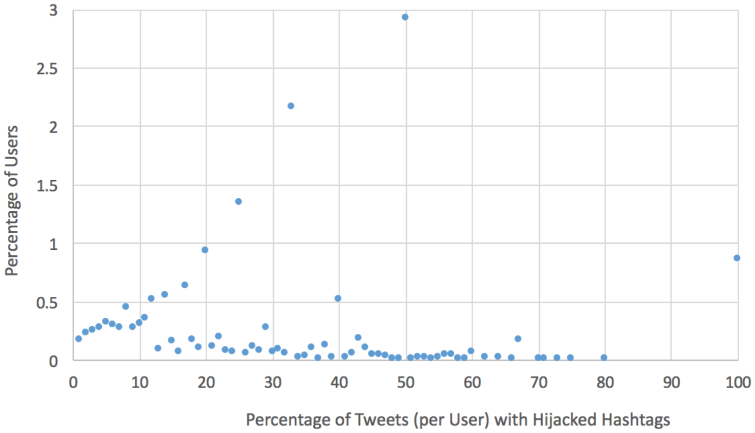


Figure 5.2: Percentage of Users who Hijacked Hashtags. Outlier (0,83X.Y) which corresponds to the 83% of users with 0 hijacked tweets is not shown.

Influence of the Hijacked Hashtags in a Profile

Figure 5.2 shows how heavily users hijacked hashtags. The results show that 83% of the users (15,730) did not use any hashtags from the opposite political tendency. Instead, almost 17% of the users in the dataset used at least one hashtag that did not correspond to their political orientation or the content of their tweets. The percentage of users almost remains constant as the percentage of hijacked tweets increases (i.e., there are both users who make light and heavy use of hijacked hashtags).

Among the most preferred hashtags to be misused, we present in Table 5.2 a list of them where it is specified the number of times that the hashtag was misused, the group of users who hijacked it and an example of tweet where the hashtag was hijacked.

Impact of the Hijacked Tweets in Each Group of Interest

Figure 5.3 shows the percentage of users in each group who posted tweets with hijacked hashtags. The results show that very few users

Hijacked Hashtag	Frequency of Misuse	Hijacker Group	Tweet
#GetoutCorreaGetout	220	Supporters	Today it is #GetoutCorreaGetout, later it'll be the same for the next president and we'll be like in the past. It's not democracy.
#GuayaquilProtests	177	Supporters	83K families with a member with disabilities left extreme poverty in 2014 #GuayaquilProtests? Seriously? #weAreRevolution
#Ecuadorprotests	68	Supporters	What is it that #Ecuadorprotests?? It is the wealthy who bother and delay the country. Long live the #inheritancelaw
#BlackSunday	61	Supporters	No to the #BlackSunday, we have to maintain the #HappySunday because the past politicians won't come back
#Morethan1000reasons (to support 'correism')	214	Opponents	#Morethan1000reasons to say #GetoutCorreaGetout
#EcuadorPeaceful	191	Opponents	Watch the President speeches on Saturdays to see if #EcuadorPeaceful is true, @MashiRafael #hypocrite
#PresidentialDialogue	95	Opponents	#PresidentialDialogue is lie after lie #noonebelievesyoucorrea
#GoOnCorreaGoOn	82	Opponents	#GoOnCorreaGoOn but go on offering sandwiches for your sheep while in the streets there are children, mothers and old people begging #GetOutCorrea

Table 5.2: Examples of Tweets where there are Hijacked Hashtags.

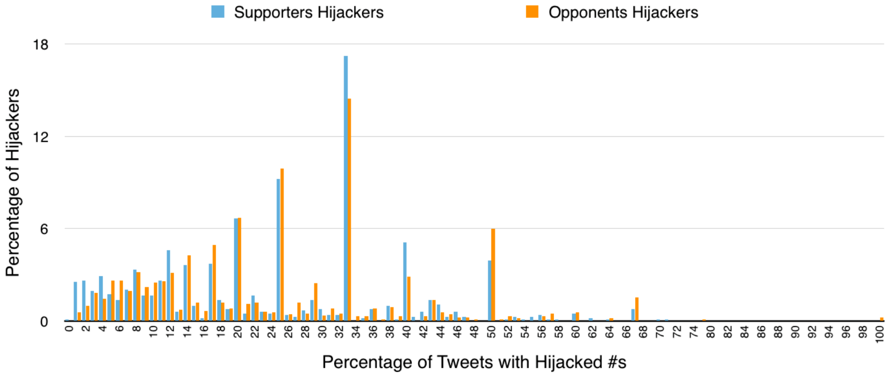


Figure 5.3: Percentage of Supporters and Opponents who Posted Tweets with Hijacked Hashtags.

posted more than 50% of their tweets using hijacked hashtags. Furthermore, a higher incidence of hijacking is found in tweets of opponents of the government (orange color reference). A depth insight of the resulting distribution showed that 33.5% of the analyzed hijackers posted few tweets (2, 3, 4, or 5). Those users tend to be more subtle about hijacking a hashtag; indeed, among their tweets only one showed to contain a hashtag that was not employed by the group to which the user belongs (i.e., the hashtag was hijacked). This behavior is reflected in Figure 5.3 where the biggest quantities of hijackers are gathered together for the percentages of tweets with a hijacked hashtag of 50, 33, 25 and 20.

Distribution of the User Political Tendency Scores

Figure 5.4 shows the values of the user political tendency scores for the detected hijackers. Users classified as supporters (38% of the total number of hijackers found), had a higher supporter tendency score, as shown on the x axis. The same notion is applied for the opponent hijackers (62% of the total number of hijackers found), with a higher value for the opponent tendency score in the y axis. It is clear that a supporter who hijacked opposition hashtags should have a s_{uS} value much bigger than the s_{uO} value. However, some peculiar cases include those users whose

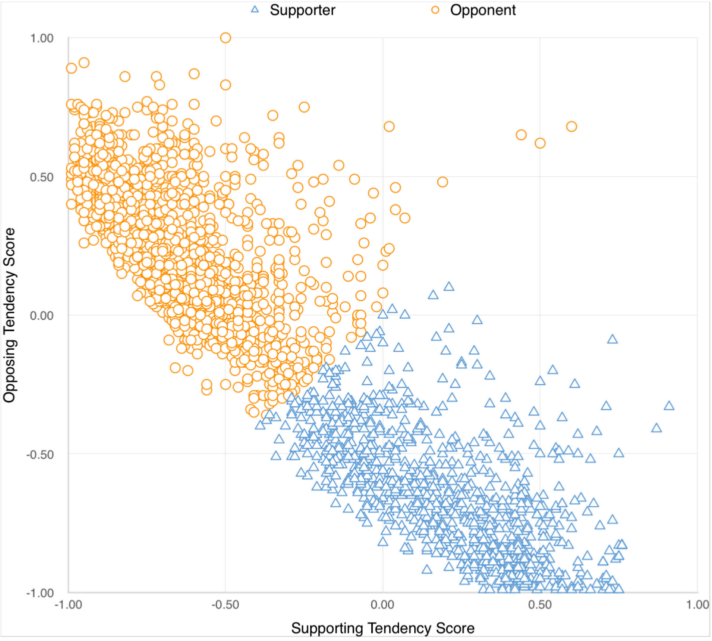


Figure 5.4: Political Tendency Score for Hijackers.

range (the difference) between s_{uS} and s_{uO} is not big enough, e.g., an opponent with the scores (0.60, 0.68). A similar situation occurred for users located in the third quadrant such as the supporter hijacker user whose scores are (-0.39, -0.40). It should be noted that the users in the third quadrant would hardly be recommended by this approach, since their tendency to be either supporters or opponents is low. Despite those users, 73% of the supporter hijackers have a positive supporter score and a negative opponent one, while 82% of the opponent hijackers have a negative supporter score and a positive opponent one.

Analysis of the Group Recommendations

The x axis of Figure 5.5 shows a score that indicates the tendency of a user from a group to use hijacked hashtags, while the y axis shows the percentage of users who received that score. When analyzing the figure,

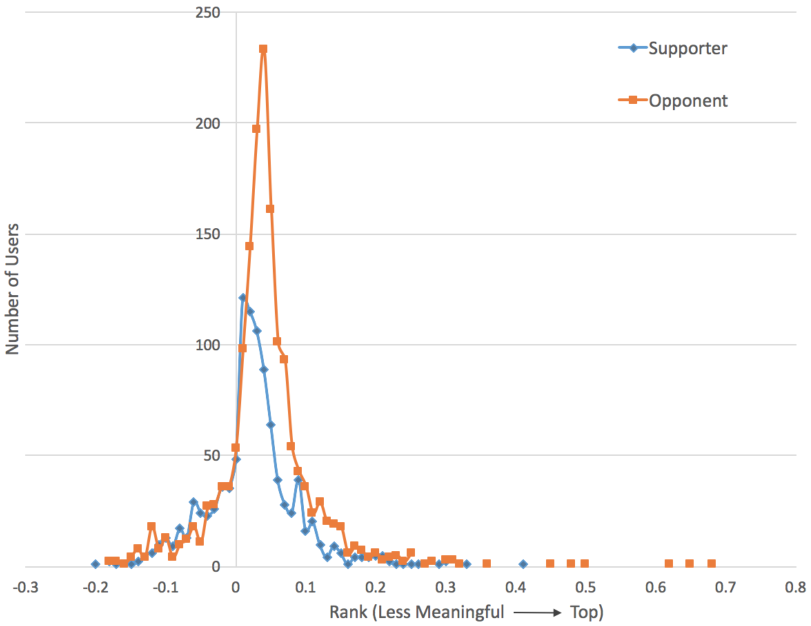


Figure 5.5: Who Not to Follow by Political Tendency and Rank.

it should be noted that recommendations of users that should not be followed would be shown from right to left (*i.e.*, those with a higher score would be recommended first). These results show an interesting phenomenon, which is that opponents tend to use more hijacked hashtags in more tweets than their counterparts. Indeed, their scores are higher, and a lot of them also have a score close to zero, which means that they would be very hard to detect manually. Supporters hijacked hashtags much less, with the vast majority of them scoring close to or even lower than 0. We suspect that this behavior may be associated with the exposure of the opponents to "official" hashtags promoted by the government which has a main figure who is the President. Indeed, for the opponents, Correa is a real target to be attacked. On the other hand, the supporters of the government have no specific hashtags to misuse or a particular popular person to be against.

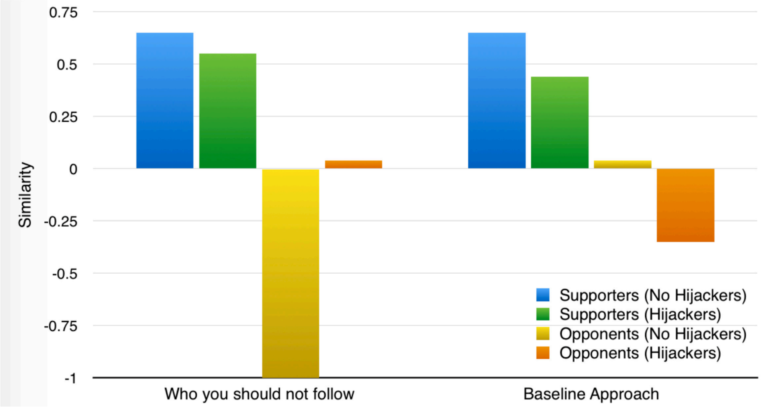


Figure 5.6: Similarity of *Supporting Hashtags* to User Categories, Results of the Proposed Method and the Baseline.

Validation of the Results

This section shows the results obtained after applying the validation strategy. Both the *"Who you should not follow"* proposed method and the *baseline* method were evaluated. In order to visualize how accurate the corresponding outputs were concerning the categorization of users into supporters and opponents, as well as hijackers or no hijackers, Figure 5.6 and Figure 5.7 are provided. By reasoning about the desired results, supporters and opponents are expected to be closer to their corresponding groups of hashtags because they generated or used them. On the other side, they should not be similar to the hashtags of the contrasting class. However, if the users are hijackers, they should tend to be closer in some way to their counterpart's hashtags (in the sense that they employ those hashtags in their tweets). That should not happen if the user is not a hijacker; then, given that he/she did not use a non-corresponding hashtag, the similarity measure should be -1.

Figure 5.6 presents the similarity of the user categories with respect to the supporting hashtags. It is showed that supporters are more similar to supporting hashtags than opponents, having the supporter hijackers with a lower similarity.

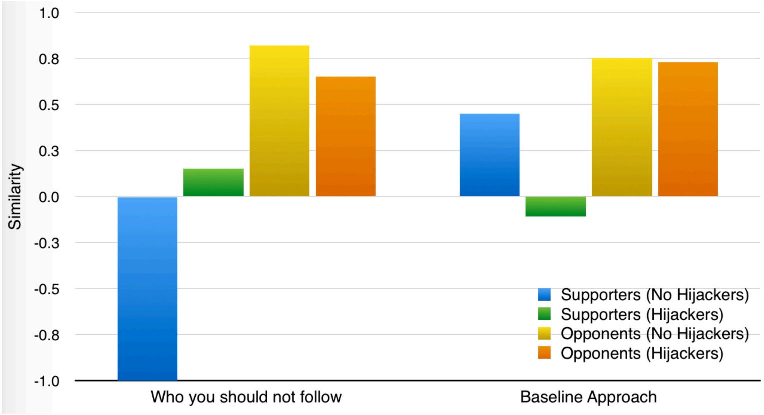


Figure 5.7: Similarity of *Opposing Hashtags* to User Categories, Results of the Proposed Method and the Baseline.

Nevertheless, in the case of opponents no hijackers the baseline method misbehaves because opponents no hijackers are supposed not to use supporting hashtags, then the calculated similarity should be -1. Figure 5.7 shows the measured similarity between opposing hashtags and the different user categories. As expected, opponents are more similar to those hashtags than the supporters. However, the same peculiar results of the baseline approach were found, where supporter no hijackers have a similarity different from -1 with respect to the opposing hashtags.

To close the analysis of results, it is demonstrated that the method proposed in the present research performs the task of classifying supporters and opponents and the respective hijackers in the context of politics in a more accurate way than the baseline method. It should be mentioned that the baseline method is not able to define a class for the user when s/he gets a score for the supporter category equal to the score for the opponent category. Indeed, in this study, 18.7% of users could not be recognized as supporters or opponents.

Besides, the baseline method assumes that a low score means a suspicious behavior, but after an understanding of the Twitter users activity, it may be said that posting a hashtag and a URL (which might sum 2 points in the score calculated by the baseline approach) does not suggest suspicion on its own. Therefore, as the method proposed in this chapter locates the users in a vector space, where the hashtags are represented as well, it is possible to categorize them by measuring distances accordingly.

5.5. Applications of the Proposed Approach

The framework presented in this work can be applied to developing *eDemocracy* projects. More specifically, on so-called voting advice applications (VAAs), which are Web-based tools that provide voting recommendations by positioning the user on a visual landscape of candidates/parties and voters, thereby indicating which candidate/party is the closest to a particular voter's view based on their answers to policy issues questions.

User and candidate profiles are designed based on their answers to questionnaires; thus, these types of profiles are considered static. Terán and Kaskina [100] propose a dynamic approach to VAA that includes social networks such as Twitter in developing profiles. The framework presented in this work can be used to extend the concepts of dynamic profiles by finding groups of interest for each of the political candidates as well as identifying malicious users (hashtag hijackers).

Further implementations can be used in other civic participation projects such as *eCollaboration* and *eCommunity*. In both cases, users interested in on-line collaborative environments and/or communities of interest can get recommendations of other users with similar profiles who are looking to gather with other users in an online community.

One of the applications for the implementation of the proposed framework is in civic participation and discussion platforms that integrates Twitter feeds like the case of *Participa Ingeligente* [101]. The framework presented in this work could be used to enhance the identification of thematic groups and possible hashtag hijackers.

5.6. Discussion

When implementing Twitter-based recommender systems that suggest social entities such as users to follow or add to Twitter lists, topics of interest to discuss, topic communities to be part of, hashtags related to the user's tweets, and user timeline alternatives, the treatment of hijackers should be part of the algorithm's design. Indeed, if the context involves controversial issues, like politics does [102], the presence of hashtag hijacking is unavoidable. For this reason, the content frame in which the hashtags are being used is explored in order to identify the users who tend to hijack hashtags.

Thanks to a corpus of tweets that particularly represents a given sociopolitical series of events, and which was used to train the *word2vec* model, it was possible to have every word (the actual vocabulary of the strikes) be seen in the right context. Then, a method where two aims were accomplished was proposed: hashtags were classified as either supporting or opposition hashtag (in spite of the presence of ambiguous terms) and users' political tendencies were calculated, based on the political tendency score of their tweets (tweet content compared to hashtag usage).

A small percentage of users (17%) who hijacked at least one hashtag in their tweets were found. Nevertheless, the users identified as hijackers did not use the same amount of malicious hashtags. Accordingly, a strategy to rank the extracted hijackers is proposed. For example, if a recommendation for a group of government supporters is going to be formulated, letting them be aware of who uses the same hashtags as them but actually oppose the government may be valuable (the recommendation of *who you should not follow* is addressed at this point).

Part II

Users Modeling based on their Social Graph

Chapters 6 and 7 of this thesis will present a method based on the detection of topic-dependent communities and event category-dependent communities, respectively. Through the detailed explanation of the approach, the way that it has been implemented in Twitter and Meetup, and the reported results of the graph metrics, we are going to see that creators of content and consumers can be linked in ways that facilitate quick and relevant information sharing. Most of the work done in this chapters was published in [103].

Chapter 6

TRENDING TOPICS COMMUNITIES: BRIDGING CONTENT CREATORS AND DISTRIBUTORS

The rise of a trending topic on Twitter or Facebook leads to the temporal emergence of a set of users currently interested in that topic. Given the temporary nature of the links between these users, being able to dynamically identify communities of users related to this trending topic would allow for a rapid spread of information. Indeed, individual users inside a community might receive recommendations of content generated by the other users, or the community as a whole could receive group recommendations, with new content related to that trending topic. In this chapter, we tackle this challenge, by identifying coherent topic-dependent user groups, linking those who generate the content (*creators*) and those who spread this content, *e.g.*, by retweeting it (*distributors*). This is a novel problem on group-to-group interactions in the context of recommender systems. Analysis on real-world Twitter data compares our proposal with a baseline approach that considers the retweeting activity, and validates it with standard metrics.

6.1. Research Problem

Once we belong to an online social network (OSN) we can share content, add people to our network, access interesting information streams created by relevant users, and express our likes and comments about items shared by other users. Personalization is a key feature in OSNs because not all the content generated by our connections may be of our interest, regardless of its quality. Likewise, not all of our connections generate content that we might consider adequate, even if it fits into our topics of interest.

In order to enhance personalization, social recommender systems as part of OSNs are in charge of filtering content streams based on the interest model of each user, the activity of their trusted social connections, and content authority. To do this, one way of finding relevant items to recommend to a user would be to discover their meaningful connections. For instance, the degree of significance could be measured in terms of the impact of the resources the user shares and the links the user has with those inside a topic-dependent community.

When a word, a phrase, or a hashtag is used with a high frequency, it is said to be associated to a *trending topic*. With the rise of a trending topic, a set of users interested in it also emerges. However, multiple points of view might be associated to it (*e.g.*, the `#donaldtrump` hashtag, related to the US president elected in 2016, has been used by people with opposing political views). Being able to manage these users and detect communities associated to a given trending topic is a problem of central interest in social recommender systems. Indeed, having a community of users who are linked and have the same interests would allow a system to generate suggestions at multiple granularities, *i.e.*, (i) for individual users, by providing recommendations of content related to the trending topic and generated by the other users in the community (thus allowing a quick and effective spread of information); or (ii) for the community as a whole, by providing group recommendations with new content related to the trending topic. At the same time, the problem is challenging, since trending topics are characterized by their temporary nature and evolve quickly; therefore, an approach that detects communities in this context should run quickly (*i.e.*, have a fast processing time), in order to dynamically adapt to the evolution of the trending topic (for example, by considering new users interested in it).

In order to tackle the problem of detecting communities related to a trending topic, in this chapter we focus on Twitter, the widely-known microblogging platform. The activity of Twitter is depicted by tweets, retweets, replies, likes and shares, and its structure is defined by *follower* and *followee* unidirectional relationships. A key characteristic of Twitter, and of our approach, in order to enable the desired spread of information, is following and being followed by other users. Follower users are interested in tracking down significant users to follow, whereas the followed (leader) users wish to accumulate a lot of followers. However, to create significant content and be a topic influential user it is necessary to obtain interesting, trendy, and relevant information to generate a tweet. One way of doing this is to form a "collusion" with other content creators or influencers in the domain. As a result, the influential group is able to share and filter key news before they become widely known, and then potentiate its diffusion through the group of users interested in that topic (who may have the role of distributors or consumers of the given topic). Accordingly, we present a method to identify groups of topic-dependent "content creators" (*CCs*) in Twitter. Another key element of our proposal is the identification of their matching spreader groups or topic-dependent "content distributors" (*CDs*). After the identification of these two categories of users, both CCs and CDs are linked by our approach in a unique community, which represents the user base for the different forms of recommendation previously mentioned.

In summary, given this real-world application scenario, our objective is to detect communities of users who (i) are associated to a given trending topic, (ii) are interested in the same content, (iii) are linked among themselves (*i.e.*, they follow each other), and (iv) can be either identified as content creators or content distributors.

Formally, the problem statement is the following: Let H be the set of trending topics at a given time. For each topic $h \in H$, let T_h be the set of tweets that contain h (*i.e.*, those associated to the trending topic), and U_h be the set of users who posted (or retweeted) a tweet that belongs to T_h. The first goal is to identify a set of *content creators* $CCs \subseteq U_h$, who generated tweets that have been retweeted multiple times. The second goal is the identification of a set of *content distributors* $CDs \subseteq U_h$, who retweeted content generated by a CC. The final goal is building a graph G that contains the CCs and CDs as vertices, connected by edges that represent the "following" and "who-retweeted-who" relationships, which will allow us to detect communities that contain both CCs and CDs.

To the best of our knowledge, our work represents the first attempt to detect several communities interested in a given topic, where each community integrates both a content creator group and the corresponding distributor group. The proposed method would improve the interaction and communication among the members of the community, and may be used to generate more personalized recommendations based on the structure of the topic-based community and levels of social influence. To summarize, our contributions are:

- We define a social model that detects topic-dependent content creator and content distributor groups on Twitter;

- The model can be embedded in an individual or group recommender system to suggest social entities;

- We validate our proposal on a real-world dataset extracted from Twitter, by employing standard metrics and by comparing it with a baseline approach that only requires the retweeting activity.

- The contributions of our work were published in Lorena Recalde, David F. Nettleton, Ricardo Baeza-Yates, Ludovico Boratto. "Detection of Trending Topic Communities: Bridging Content Creators and Distributors", in Proceedings of the 28th ACM Conference on Hypertext and Social Media (HT '17), July 04-07, 2017, Prague, Czech Republic.

The remainder of the chapter is organized as follows: Section 6.2 summarizes the context of the present work and the related state of the art; Section 6.3 describes our approach; in Section 6.4 we present the analytical framework built to validate our proposal and the obtained results; finally, in Section 6.5, we discuss some of the relevant findings of the research done.

6.2. Related Work

The Social Web has shown to be one of the richest sources for mining people's interests, personality, and social interactions [58]. Therefore, recommender systems extended the traditional methods like Collaborative [104] and Content-based Filtering [105] to include users' information extracted from their OSNs. In this way, Social Recommender Systems make more personalized suggestions based on an improved user preferences model [106]. Several relevant works related to the present chapter are discussed next.

6.2.1. OSN Analysis to Discover User's Interests

It has been shown that friends are able to make suggestions in a different number of domains and also share some similar interests [67]. Therefore, recommender systems might make suggestions for the target user based on her/his friends' preferences. Thus, *social recommender systems* have emerged with the aim of modeling the user's preferences by using the information s/he and their friends have published in OSNs. For instance, the study done in [107] demonstrated that friends of the target user provided more useful and better recommendations than recommender systems. Ma *et al.* [108] also modeled the preferences of the user in a social recommender system. They took into account that some of the user's friends might have different interests. The premise is that people tend to look for their friends recommendations; hence, this work establishes the difference between trust relationships and social friendships.

The authors represent the diversity of tastes among the user's social connections using matrix factorization to improve the accuracy of the recommendations. In our research we also consider the exploration of users' connections in the Social Web. However, our approach differs from [107] and [108], since the item recommendation for the user may be not only based on his/her direct friends, but also on a community to which the user belongs and which is related to a topic of interest.

6.2.2. Social Entity Recommendation on Twitter

There are two important concerns about information stream personalization (Twitter activity feeds): (*i*) items or news feed filtering of what is to be considered of interest, and (*ii*) relevant content discovery that comes from friends of friends [23]. In [22], the authors present a framework that merges a traditional collaborative ranking approach with Twitter features such as content information and social relations data, so the model can generate better personalized tweet recommendations. In [59], the authors make a proposal to solve the news feed filtering problem in OSNs by presenting a method that automatically reorganizes the feeds and filters out irrelevant posts. The authors in [60] propose a "users to follow" recommender, implemented by using real time data from Twitter. The details about profiling algorithms and recommending strategies used in their recommender system are presented in http://twittomender.ucd.ie. Each user is modeled considering their recent Twitter activity and their social graph.

Other social entities to recommend to Twitter users are hashtags. Users can add some words prefixed by the symbol # to their tweets and they are identified as hashtags. The hashtags give some relevant meaning and structure to the users' posts as a folksonomy. In [109], a method that recommends hashtags is presented. It is based on finding similar user-tweet pairs to the target user-tweet pair, so the hashtags used by the neighbors may be recommended. Compared to the state of the art, our approach may also be used to generate recommendations of news feeds, users to follow, hashtags, and other social entities. However, the novelty of our method is to employ a trending topic of interest to a set of users; consequently, the recommendations that can be generated are topic-dependent and are different for users who are content creators and for those who are distributors.

6.2.3. Social Influence and Grouping

In general, people do not make decisions in a completely rational way; instead they are usually influenced by many factors [67]. Marketing and e-commerce have exploited data in social network sites to propagate knowledge about products faster and collect users' opinions about them. Depending on these connections, consumer groups or communities are then detected. Dholakia *et al.* [110] present a model that structures the role of social influence by the community on its members to define its effect at the moment a user makes a choice, participates in collaboration activities, adopts certain behavior or goes into an engagement process. In the model, they set decision making as a direct function of social influence and as an indirect function of worth judgment.

In [25], the study shows the identification of influential tweeters based on their social and commercial importance. The authors propose a method in which the influential users are classified and ranked by topic of interest, and every topic has a small set of representative words associated with it. In [111], the researchers analyze three measures of influence in Twitter: indegree, retweets, and mentions per user in their dataset, as well as how influence varies across topics. They found that the most influential accounts were authoritative news sources and content trackers.

Some researchers in the field of group recommender systems have seen that social factors, inherent in human behaviour, influence the recommendation and adoption phases. In [112, 113, 114], the authors study social influence inside groups to evaluate how this can be used to improve group recommender systems design. The work in [115] explains the *two-step flow model of influence* [116] where it is said that a small number of people act as influential individuals transmitting information with their own view of mass media to the rest of society. The first step refers to the transmission from the mass media to a group of influential people, and the second step comprises the diffusion of information from the influential group to a bigger audience. Those are the two steps in which a group of leaders may accelerate or prevent an item adoption. From this comes the motivation of our current work to identify influential groups involved in a specific domain of interest in an OSN, where those groups are formed by joining content creators and detecting their corresponding set of distributors. The result may be used to build or improve users' preference models and then formulate social item recommendation. This social model has not been proposed before in the related state of the art.

6.3. Approach

This section provides the details of our approach, named $TreToC$ (which stands for "*Tre*nding *To*pic *C*ommunities"), able to identify content creators and content distributors, as well as detect topic dependent communities related to a trending topic. The approach works in three steps:

1. **Identification of *CCs*.** Analyzing the activity of the users who tweeted about a given trending topic, this step identifies the *content creators*, *i.e.*, those who generate content that is subsequently retweeted by other users.[1]

2. **Identification of *CDs*.** Analyzing the activity of the users who tweeted about a given trending topic, this step identifies the *content distributors*, *i.e.*, those who retweet content generated by the creators.

3. **Detection of Trending Topic Communities.** Given the sets of users detected in the previous two steps, we first generate a graph

[1] At the time this work was done, the comments to a tweet as threads were not part of Twitter functionality nor is there this type of content in the dataset used. However, we think that the method can be extended to include, as content creators, those who generate threads given a tweet as long as they meet the conditions presented in Equation 6.1.

G that connects them, and then apply a community detection algorithm to detect communities associated to the considered trending topic.

What follows is a systematic account of how the tasks performed by our approach have been implemented.

6.3.1. Identification of CCs

Users with a certain number of followers, whose tweets are quickly propagated or retweeted because of their content, and who are experts or somehow represent a specific domain, may be considered creators of significant content.

Given a trending topic $h \in H$, we collect the set of tweets T_h that contain h and consider the set of users U_h associated to these tweets (*i.e.*, those that either tweeted or retweeted content in T_h). Out of all the collected tweets, let T'_h denote the set of tweets that do not represent retweets (*i.e.*, those tweets that contain original content).

Every tweet $t \in T'_h$ is created by users who promote the content amplification over the social network. However, not all the users who generate content can be seen as topic propagators. Indeed, it is essential that the content is considered as interesting by other users, who retweeted a given tweet $t \in T'_h$ at least once. For this reason, we build a set $\hat{T'_h} \subseteq T'_h$, which contains these tweets:

$$\hat{T'_h} = \{t \in T'_h : retweets(t) > 0\}$$

where $retweets()$ is a function that returns the number of times a given tweet was retweeted by other users.

Given the previously defined set, we designate as $CCs \subseteq U_h$ the collection of *content creators*, who favor the content generation. More formally, the set of content creators is defined as follows:

$$CCs = \{u \in U_h : \exists t \in \hat{T'_h} \; s.t. \; author(t) = u\} \tag{6.1}$$

where $author()$ is a function that returns the author of a given tweet.

6.3.2. Identification of CDs

A user who follows another is probably interested in knowing the content s/he posts, but if the user retweets that content as it is, s/he is showing an agreement with it. Moreover, considering the diffusion of a topic, some particular level of interest arises, since many people retweet the emerging tweets. Therefore, the fact that a user *retweets* the tweets of another user is an important source of information to identify the content distributors of a trending topic. Consider that every user $u \in CCs$ posts a tweet $t \in \hat{T_h'}$. Let R_t be the set of tweets that represent a retweet of t:

$$R_t = \{t' \in T_h \setminus \hat{T_h'} : rt(t', t) = true\}$$

where $rt()$ is a function that returns true if a tweet t' is originated by a tweet t (*i.e.*, if it is a retweet of t).

We define as *content distributors* (CDs) the set of users who retweet content in $\hat{T_h'}$ and act as propagators. More specifically, the set is defined as follows:

$$CDs = \{u \in U_h : \exists t' \in \cup_{t \in \hat{T_h'}} R_t \; s.t. \; author(t') = u\}$$

It is worth highlighting that in our approach replies to a tweet are not considered, as they cannot be treated as forms of agreement. It should also be noted that, unlike the *retweeted* content of users, their *favorited* content is not shown in their followers' timelines; thus, favorite activity does not promote the spread of a topic and it is not considered as part of our study.

6.3.3. Detection of Trending Topic Communities

Given the set of users who generated topic-dependent content (CCs) and those who retweeted this content (CDs), the first goal is to find an effective way to link them. Indeed, in order to allow a rapid spread of information, users should follow each other. Moreover, we have to ensure that an explicit connection between a CC and her/his CDs is present.

In order to detect the communities related to a trending topic $h \in H$, it is first necessary to build a graph $G = (V, E)$ that represents the previously mentioned connections. The set V of vertices is represented as the union of the two sets of users identified in the previous two steps:

$$V = CCs \cup CDs$$

In order to build the set E of edges that represent the connections among the users, we consider three types of relationships. The first is the following relationship between two topic-dependent content creators:

$$F_C = \{(u_x, u_y) : follow(u_x, u_y) = true, u_x, u_y \in CCs\}$$

where $follow()$ is a function that returns true if the first user follows the second.

The second type of connection we consider is the following relationship between two topic-dependent content distributors:

$$F_D = \{(u_x, u_y) : follow(u_x, u_y) = true, u_x, u_y \in CDs\}$$

In the third type of connection we link a CC to a CD only if the CD retweeted content generated by the CC. Note that we avoid adding in the graph the following relationships between CCs and CDs because, in this context, this kind of link would be too generic and too weak to relate two users. Indeed, even if a user follows another, it cannot be taken for granted that these two users agree on everything.

Saying that a Twitter following relationship does not explicitly show dependency to a given topic may sound arbitrary. However, if a user retweets another but does not follow him/her, and the following relationship would represent the link between two users, there would be no connection between them (even if, with respect to the trending topic, an important connection between the two users exists). Moreover, our focus is to detect communities in which the consumers (or distributors) get in touch with agreeable content with respect to the considered trending topic.

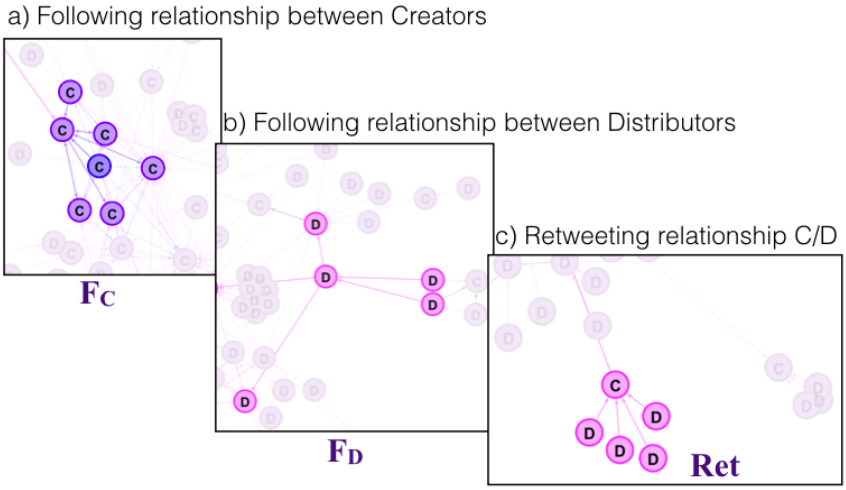

a) Following relationship between Creators
b) Following relationship between Distributors
c) Retweeting relationship C/D
FC
FD
Ret
C
D

Figure 6.1: Proposed Relationships between Users in TreToC Method.

Then, the connection between a CC and a CD is well represented by a retweeting link. More formally, the set can be defined as follows:

$$Ret = \{(u_x, u_y) : \exists(t', t) \in T_h \; s.t. \; rt(t', t) = true \wedge \\ author(t') = u_x \; \wedge \; author(t) = u_y\}$$

To exemplify, Figure 6.1 shows the three kinds of connections between CCs and CDs.

Finally, the set E of edges in the graph is represented as:

$$E = F_C \cup F_D \cup Ret$$

At this point, the Louvain method [117] is applied to detect topic-dependent communities of interest in the graph G. The choice of employing a community detection algorithm was made since it can easily handle networks with millions of nodes in a very short time. This characteristic of the algorithm fits with our need to detect communities that rapidly evolve and are characterized by a temporary nature.

Given the evolution of a trending topic over time (*e.g.*, the appearance of new users that generate new content related to the trending topic), being able to detect communities in a matter of seconds allows the algorithm to work in a real-time scenario like the one we are considering. Another interesting feature of the Louvain method is its capability to generate communities at different granularities (the structure returned by the algorithm is a dendrogram). Therefore, if a trending topic is emerging, our approach would be able to consider communities at higher granularities to make sure that each community contains both content creators and distributors, and if a topic has existed for a longer amount of time and more users are participating in it, communities at lower granularities might be considered.

As previously mentioned, this capability of the Louvain algorithm to rapidly detect communities would allow to capture a snapshot of the evolution of a trending topic (*e.g.*, at fixed time intervals, it would be possible to re-run the algorithm). However, since our proposal was conceived to provide effective recommendations to the users (both individuals and groups) there would be no need to recompute the communities too many times, to avoid "flooding" the users interested in the trending topic with excessive information.

6.4. Analytical Framework

This section presents the analytical framework and gives our results. We first present the analytical strategy and setup (Section 6.4.1), followed by a description of the employed dataset (Section 6.4.2) and metrics (Section 6.4.3). Finally, we present the analytical results (Section 6.4.4).

6.4.1. Analytical Setup and Strategy

The environment for this work is based on the Python language. To build and manipulate the graph, as well as to calculate the metrics presented next, we used the *NetworkX* module.[2] However, the clustering coefficient of nodes for directed graphs is not part of the functions.

[2] https://networkx.github.io

Then, we implemented it following its formal definition. To run the Louvain community detection algorithm and measure the graph modularity we used the *community* module.[3] In order to ensure the repeatability of the analyses, some parameters need to be considered:

- By construction, the graph is *directed* and *unweighted*;
- To define the communities the function employed was *community.best_partition()* where the *resolution* parameter is set to 1. The resolution in modularity is used to adjust the optimization in partitioning (varying the number of communities [118]). If this value is bigger than 1 it leads to the merging of two communities that share one or more edges, independently of the communities' features. We did not alter the resolution to avoid bias. Because of the properties of Louvain, the directed graph needs to be transformed into undirected when calling the function.[4]

The dataset employed in the analyses is the only one existing in the literature containing trending topics on Twitter and the tweets associated to them, which we enriched with the *following* relationships between the users, collected thanks to the Twitter API.

To validate our proposal, five sets of metrics were used for analysis:

1. **Characterization of the trending topics.** Given a trending topic, we analyze the number of content creators and distributors that characterize it. This will allow us to understand the dynamics that characterize the activity on Twitter, even before communities are detected.
2. **Analysis of the disconnected users.** In this case, we analyze the percentage of disconnected users from the graph (which would not

[3] http://perso.crans.org/aynaud/communities/api.html

[4] Note that, even though the communities were detected on the undirected graph, the metrics to evaluate their quality were measured on the original directed graph, as described in Section 6.3.3.

be involved in the community detection[5] and thus would not benefit of the information spreading).

3. **Analysis of the cohesion among the users.** For each community, we evaluate its quality by measuring the cohesion between the users in it, using standard metrics such as modularity, ratio between the number of communities and the number of users, and density.

4. **Analysis of the community structure.** For each community, we analyze its composition, by measuring the ratio of content creators and distributors in it, and their clustering coefficient. This allows us to evaluate the effectiveness of our approach to connect those who generate the content to those who make use of it.

5. **Analysis of the relationships between the users.** On Twitter, there are some kinds of relationships that connect people together. Our assumption was that users in a social network might be connected at a given time because of a common topic of interest. However, these users might be associated to a topic because of previous relationships between them (*e.g.*, friendship). In order to validate that our communities are topic-based and do not appear together because of previous relationships, for each set of trending topics that share at least one user in common we analyze the percentage of users who take part in the intersection by measuring the Jaccard index. For instance, if two graphs (that share users in common) overlap, it would be an indicator that the graphs emerged from existent relationships among the users; otherwise, it would show independence of graphs generation and the corresponding users' *following graphs*.

In order to verify the choices made in our approach to consider the three previously presented types of connections in the graph, we compare our proposal with a baseline approach named *Retweeting-Based Communities* (RBC).

[5] Community detection algorithms work on the largest connected component of a graph.

In the RBC method, the set of edges in the graph connects two users only if one retweeted the other. The *retweeting relationship* is the basic connection between the two kinds of users being analyzed and represent the initial action that makes a topic become a trend. It is worth mentioning that a graph built on the *following* relationship only would represent a community detection performed on the original Twitter graph, and this is unrelated to the trending topic dependency, so we discarded a baseline that considered only this type of connection.

6.4.2. Dataset

The analyses were performed on a dataset specifically built to collect information about trending topics on Twitter, which was presented in [119] and is available online.[6] The dataset contains 1,036 trending topics, which are associated to 567,452 tweets from 348,757 different users. However, in order to form the graph and detect the trending topic communities, the information about the tweets and the users who posted them is not enough. Indeed, we need to have the following relationship between the content creators, and the following relationship between the content distributors (Section 6.3.3).

This was collected by querying the Twitter API, for the first 368 topics (due to the limitations imposed by the API on the number of calls that could be made). The final dataset contains 67,607 tweets, which correspond to the content retweeted at least once and the retweets found during collection, 15,918 unique creators, and 36,890 unique distributors. Of these, 673 were found to be creators of one topic and distributors of another. If a creator retweeted a tweet in the same topic, s/he was considered only as creator, in order to keep the topic graph structure proper. In conclusion, the total number of users in our study was 52,135, having 29.24% of them as creators, 69.46% as distributors, and 1.3% acting as both (in different topics).

[6] http://nlp.uned.es/~damiano/datasets/TT-classification.html

6.4.3. Metrics

The method we propose produces a graph for a trending topic being analyzed. The graph is then divided into communities of interest. For example, if the trending topic is #dataPrivacyDay, there might be one community interested in data privacy laws in Europe, another community that promotes a conference to celebrate the Data Privacy Day, and another community in favor of data disclosure.

Both the graph and its communities can be evaluated by using the following metrics.

Ratio of disconnected users

The *ratio of users disconnected from the graph* measures the fraction of users, either content creators or distributors, who are not present in the graph because of the lack of linkage. Let $\overline{V} \subseteq V$ be the subset of users for which there is no edge $e \in E$ that connects them to the graph G. The ratio is calculated as follows:

$$|\overline{V}|/|V|$$

Cohesion among the users

After executing the community detection, every node in the graph is going to be assigned to a community. The *modularity* is a value that represents the strength of division of a network into communities. High modularity means the connections between the nodes within communities are dense and the connections between nodes in different communities are sparse. The algorithm returns this metric after the community detection process is finished. Readers can refer to [117] for further details.

The *ratio between the number of communities and the number of users* allows us to evaluate the ability of an approach to group the individual users into communities. Indeed, higher values represent a low cohesion among the users (they are not added to the same community), while lower values indicate a smaller number of communities and higher cohesion among the users.

The *density* is the ratio between the number of edges per node to the number of possible edges. The density of a directed graph $G = (V, E)$ can be calculated as:

$$|E|/(|V| \times (|V| - 1))$$

Community structure

Every community is expected to have several content creators in order to have newly generated content that can be spread through the community. The *number of creators per community* quantifies how many creators we can find in a community.

The *number of content distributors per community* measures how many distributors we can find in a community.

The last metric we are going to use, the *community clustering coefficient*, quantifies the extent to which nodes in the graph tend to cluster together. The clustering coefficient for nodes in a directed graph is defined by:

$$C_i = |\{e_{jk} : v_j, v_k \in N_i, e_{jk} \in E\}|/k_i(k_i - 1)$$

where k_i is the number of neighbors of a vertex $v_i \in G = (V, E)$ and N_i is defined as the neighborhood for the vertex:

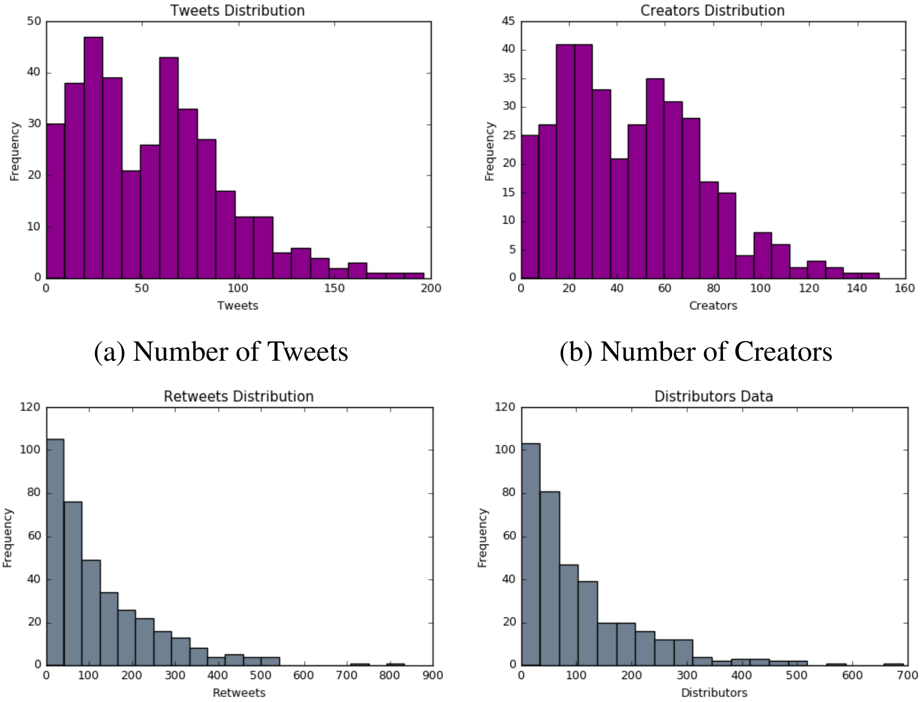
$$N_i = \{v_j : e_{ij} \in E \vee e_{ji} \in E\}$$

6.4.4. Analytical Results

Next, we provide a detailed evaluation of our approach according to what has been described in the Analytical Framework (Section 6.4).

Characterization of the trending topics

In the following, we analyze what characterizes the trending topics in the dataset. Each histogram in Figure 6.2 represents the number of tweets found per trending topic (a), the number of creators who posted



(a) Number of Tweets

(b) Number of Creators

(c) Number of Retweets

(d) Number of Distributors

Figure 6.2: Distribution of the number of tweets, creators, retweets and distributors for the Trending Topics.

those tweets (b), the number of retweets found per trending topic (c), and the number of distributors who made those retweets (d). From the distributions obtained as results, we see that content generation (creation of tweets) and content propagation (retweeting action) behave differently. Indeed, the data related to tweets presents a distribution with two maximum values of approximately 25 y 65 tweets and the average number of tweets per trending topic is 55.51. The tweets distribution is slightly skewed to the right showing that few trending topics have more that 120 tweets (Figure 6.2a). In Figure 6.2b we see that the distribution of creators is very similar to the distribution of tweets. Few trending topics contain more that 90 content creators. In average, the number of creators per trending topic is 46.20. In contrast, with respect to the content propagation (Figures 6.2c and 6.2d), the data for retweets and distributors presents a Power Law distribution. This shows that there are few trending topics that reached a high incidence of retweets/distributors. The median value for the retweets per trending topic is 84.5 and the median number of distributors per trending topic is 69.

Analysis of the disconnected users

When a user has a connection with another, it is going to be considered in a graph as a source or destination node, depending on the relationship. Accordingly, the average number of disconnected users for the trending topics was analyzed. Non-linked users are detected once the trending topic graph is obtained by following the corresponding approach, RBC or $TreToC$, while the rest of the users form the main connected component.

The results show that the trending topic graphs generated with our approach ($TreToC$) cover 85% of the users who take part of the topic. The RBC baseline covers 76.4% of the user base of the dataset. These results demonstrate that our graph construction approach (presented in Section 6.3.3) includes more users in the community detection process. Indeed, if only the retweets (RBC baseline) are considered, more users are left out of the detected communities with respect to our approach, thus reducing the information spreading.

Analysis of the cohesion among the users

In order to analyze the level of cohesion between the users in a community, in Table 6.1 we report the average values of *modularity*, *ratio between the number of communities and the number of users*, and *density*, for our approach $TreToC$ and the baseline RBC.

The corresponding metric values obtained are presented in Figure 6.3. A lower *modularity*, as that obtained in the $TreToC$ method, shows that the communities in the graph maintain a certain level of interaction or

Method	Modularity	Ratio of Communities per Nodes	Density
RBC	0.780	0.278	0.021
TreToC	0.622	0.162	0.027

Table 6.1: Cohesion among the users (average).

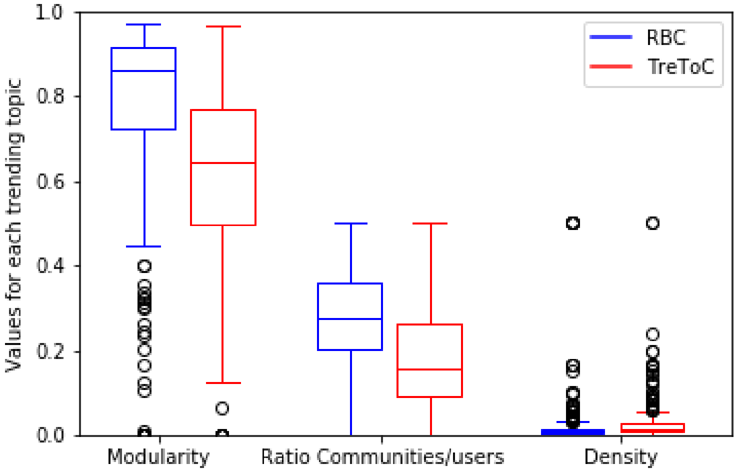


Figure 6.3: Cohesion among the users: Distribution of the metric values.

connection between them. This is not seen in RBC graphs where the distributors behave only as source nodes, causing the modules partitioning to be well defined. In this case, if we would like to adjust the modularity resolution to get fewer communities it would not be possible because the RBC communities are not connected between them. Indeed, it is shown that 75% of the TreToC graphs have a modularity lower than 0.76 while by applying RBC, 75% of the graphs have a modularity bigger than 0.7. Furthermore, the *density* is influenced by this fact, since the

Method	% of Creators	% of Distributors	Clustering Coeff
RBC	4.54	6.76	0.000
TreToC	7.76	9.05	0.077

Table 6.2: Community structure (average).

users in $TreToC$ share two links (*i.e.*, following as well as retweeting) and act as source or destination nodes, resulting in a greater density value compared to density for RBC. Note that a higher modularity does not necessarily mean 'better', it is better just when we want smaller communities (in terms of number of vertices) or non-connected communities (as the RBC baseline produces). Nevertheless, the purpose of our work is to get fewer communities, which are highly associated units composed by a suitable number of content creators and distributors. For example, more linked content creators in a community would cause diversity in future recommendations.

The average number of communities found in an RBC graph is 0.28 per user (or 28 per 100 users), that exceeds the average amount of communities found in a $TreToC$ graph (16.22 communities found for a trending topic with 100 users) which is what our method looks for (*i.e.*, our approach obtains fewer but larger communities).

Analysis of the community structure

Table 6.2 shows a summary of the analysis of the community structure obtained for the set of trending topics in our study. More specifically, this analysis measures the *average percentage of content creators*, the *average percentage of content distributors*, and the *average clustering coefficient* for a given community.

The RBC method relates two content creators only if a suitable number of distributors retweeted both of them; therefore, we are going to be able to find only a few creators in a community (4.54% in average). In the

$TreToC$ method, the following relationship joins content creators making it more likely to find them as close neighbors (in average, 7.76% of creators are found in a $TreToC$ community). Consequently, their individual distributors are also closer. We can observe this in the percentage of content distributors in a community in $TreToC$ method, which is also greater. As a consequence of being able to have more content creators and distributors linked together in a $TreToC$ community, the *clustering coefficient* increases compared to the RBC graph.

The boxplots in Figure 6.4 report the results of the three metrics found over the trending topics and compare the two approaches. From the results, we notice that the $TreToC$ method creates communities where we can find groups with 2.5 content creators and 4 distributors in average. These values are slightly greater than the values for RBC communities. In general terms, the figure shows how the $TreToC$ communities increase their number of creators and distributors. The difference is remarkable, especially in the outliers.

Note that to represent the average clustering coefficient in the same figure, there is another *y axis*. As the clustering coefficients obtained for the RBC communities had a value of zero, they are not plotted in the figure.

Analysis of the relationship between the users

We considered the users who participated in more than one trending topic (3,599 users) and either appear (i) as distributors in a given topic and also as creators in other topics (18.7% of the mentioned 3,599), (ii) only as creators in more than one topic (22%), or (iii) only as distributors in more than one topic (59.3%).

We evaluated to which extent those trending topics that share one or more users are overlapped, in order to find out if our communities are topic-dependent or exist because of previous relationships. Then, we calculated the Jaccard index considering all the users of the set of *possible* overlapped trending topics. We obtained 1,894 different combinations of trending topics that had users in common and the basic statistics show

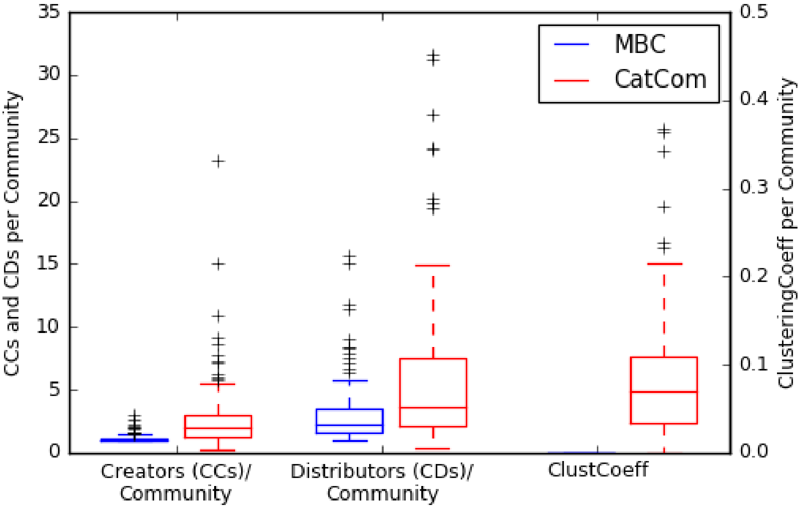


Figure 6.4: Community structure: Distribution of the metric values.

an average Jaccard index of 0.008, having 0.0004 as the minimum value found and 0.65 as the maximum value.[7]

The results validate our approach, whose main focus is to find topic-dependent communities where the users are related to a topic and then linked. The users are gathered together because of the topic and not because of previous relationships between them (indeed, the Jaccard index is very low). As an example, consider the two trending topics '`#dealwithit`' and 'Vernon Gholston', both related to sports and sharing users in common. The hashtag `#dealwithit` was used by fans of the American football team Buckeyes, who posted tweets like 'Go Buckeyes! 93-65 #dealwithit Wisconsin'. On the other hand, the proper name 'Vernon Gholston' belongs to an American football player (who played

[7] A Jaccard index near to 1 would show a total overlap; that is to say, all the users involved in a given trending topic are associated to a second trending topic. This level of association would mean that the two corresponding graphs appeared because the users have a relationship (*i.e.* they follow each other, then they retweet each other).

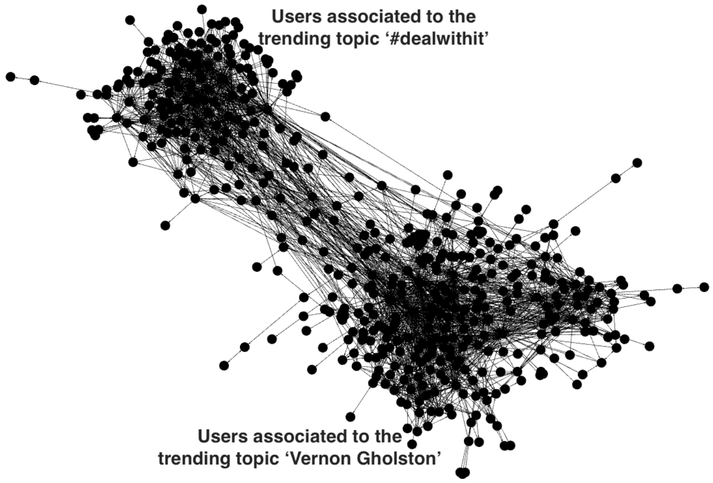

Users associated to the trending topic '#dealwithit'
Users associated to the trending topic 'Vernon Gholston'

Figure 6.5: Two trending topics graph whose structure is based on the TreToC relationships.

in Buckeyes). Indeed, the two trending topics are connected between them, hence the overlap between the users. The graph obtained by taking the CCs and CDs for both topics and relating them according to the proposed method (Section 6.3.3) is shown in Figure 6.5. However, despite the shared users, the graph presents two separated groups of participants that are actually dependent of their respective topic, being the `#dealwithit` group the smallest one. The lack of overlapping shows that both graphs are originated by the emergence of two trending topics and the corresponding topic-dependent users who are not necessarily "friends" in Twitter.

6.5. Discussion

We now summarize and discuss the results obtained in our analysis. When working with trending topics, Section 6.4.4 showed us that while content generation presents a bimodal distribution, content propagation has a “long tail” distribution where few trending topics reached more retweets and distributors than the average. As the analysis of the disconnected users showed (Section 6.4.4), in order to detect communities that are related to a trending topic and involve most of the users, it is necessary to link the users both with the “following” and “who-retweeted-who” relationships. Indeed, the retweeting relationship alone leaves around 24% of the users out of the graph, while the other around 15%.

The analysis of the cohesion among the users (Section 6.4.4) showed that the communities we created are large (the number of communities is very low if compared to the number of users), that the users in a community are well connected (density is high) and that the communities themselves are connected (modularity is not high); this means that the evolution of a trending topic over time would allow a user to be moved from one community to another, to better fit with her/his current interests and the evolution of the trending topic itself.

The third analysis, which studied the structure of the communities (Section 6.4.4) showed us that each community contains both around 7.76% of the content creators and 9.05% of the distributors (this would allow the distributors to get in touch with diverse content, generated by their content creators counterpart); moreover, the clustering coefficient confirmed that the nodes in the communities tend to cluster well together (the values are high), thus enabling the desired spread of information. The last analysis showed that, even though some topics are related and share users in common, they do not overlap, because the users participating in a given topic depend on it (*i.e.*, the communities formed around a trending topic are actually topic-dependent and do not exist because of other types of relationships).

Chapter 7

EVENT CATEGORY COMMUNITIES: RELATIONSHIPS BETWEEN EVENT-BASED SOCIAL NETWORK USERS

In this chapter we use the *Meetup* application to study how users behave in the context of event organizers and attendees, within category communities in which event organizers and group members interact. We analyze how they become linked and how we can increase awareness of other category groups which they can join or events they can attend. Some of the metrics that communicate graphs' properties like nodes cohesion and community structure are explained and reported in order to compare our proposed method, named *CatCom* for Category Communities and the baseline method *MBC* or Membership-Based Communities. In this chapter we map the same background that we saw in our research in Chapter 6 to verify the ability of the approach to be extended to other platforms different than Twitter. We achieved this objective in a satisfactory manner and we can argue that inferring the appropriate connections between users who fulfill a determined role in the social network, could generate richer information that can be used to design recommendation systems.

7.1. Research Problem

Real-world events and their organization mediated by *Event-Based Social Networks* (EBSN) have gained popularity over the past years. For instance, *Meetup* is an EBSN where users plan, organize and publish *offline* events. It facilitates online group formation and the announcement of public or private events. The real-world interactions of people-to-people or people-to-event are captured by the platform in the manner of attendance confirmation, payment of fees (it may be the case for some events) and ratings provided by the attendees.

Among the kinds of interactions of Meetup's entities, we have considered *group membership* where the Meetup group has an organizer and members. While the organizer creates events, the members are notified about them in order to attend. Joining a group is a core action in Meetup that emerges naturally (determined by users interests) and may be explicit as soon as the user register in the platform. By analyzing the available data related to Meetup groups (category, organizer, members), we can make use of it to provide better services.

In this chapter, we analyze the relationship of meetup organizers and members and provide the first known in-depth study of event category communities on Meetup social network. We propose a method with the aim of generalize and extend the research presented in Chapter 6. Then, we demonstrate that by using a graph-based approach we can discover category communities and study their structure. In fact, the relationships of the users in a community help to distribute their attributes (like preference for wide-ranging topics) for others to explore and experiment. Therefore, as the aim of an EBSN is to let the users meet new people and enjoy offline activities, our proposal fits in this challenge.

Meetup recommends three kinds of "items": groups to join, events to attend and topics to label what a new group is intended for. Our proposal may be used to design the mentioned kinds of recommendations for both members and organizers. We used a dataset of 3.9K groups hosted in *Meetup*. Those groups where collected considering nine different cities. The structure of the remainder of the chapter is as follows. We start with the context of our research and related work in Section 7.2. Then, after introducing some characteristics found in terms of MeetUp groups organizers/members and their relation with categories in Section 7.3, we explain our approach in Section 7.4. Section 7.5 presents the analytical framework and results and finally, we discuss some implications in Section 7.6.

7.2. Related Work

We begin by describing three types of concepts that appear in most works related to EBSNs: *1)* Popular Events Discovery, *2)* Recommendation of Events and Groups to Join, and *3)* Graph-based Strategies to Model EBSN Users.

7.2.1. Popular Events Discovery

In [120], the authors address the problem of combining the latent factors of group-organized event popularity to predict how successful the event would be given a category (relative popularity). They study spatial, group, temporal and semantic features, to propose four contextual models. Besides, the authors present a group-based social influence model (social propagation network) specific to the event organizer. A combined framework is proposed and evaluated by using datasets collected taking into account three cities. Pramanik *et al.* [121] propose a method to quantify the success of Meetup groups based on a machine learning model that leverages on particular features. The authors motivate the selection of such features considering those that could measure a group success by representing its ability on (1) organizing popular events, (2) attracting many attendees and (3) maintaining a large growing group. The goal is to provide guidance to the category-dependent group organizers and event hosts in order to form a successful group or to host a successful event.

The scope of the proposals presented comprises the analysis of events by category, as well as ours do. However, they have as goal a popularity prediction task, while we study activity surrounding category-based Meetup groups by analyzing the relationship organizer-members. This kind of data modeled in a graph allow us to have insight into the characteristics of the communities detected.

7.2.2. Recommendation of Events and Groups to Join

The authors in [122] focus on the problem of predicting users social influences on upcoming events in the DoubanEvent platform (EBSN in China). They created a user-event social influence matrix where the aim is to estimate the unobserved values based on the influence of a user on an event. Such influence is calculated considering the number of user's friends influenced to attend the event. The solution that they propose employs event-based and user-based neighborhood methods combined with event and user features into matrix factorization. Similarly, Du *et al.* [123] propose an algorithm by integrating Singular Value Decomposition with Multi-Factor Neighborhood (SVD-MFN) to solve the event attendance prediction task. Their framework fuses the discovered factors (content preference, spatial and temporal context and social influence) through a neighborhood set. The framework validation was done in DoubanEvent dataset.

Different from Meetup, DoubanEvent provides the users or event participants the possibility to follow each other. The connection or social links are explicit; then, the two approaches mentioned before use this platform characteristic to measure social influence and solve the problem of event recommendation. In Meetup, the 'declared' link among users is the group member-organizer connection. Therefore, it is the characteristic that we employ and reinforce.

In terms of learning to rank events for personalized recommendation, in [5], Macedo *et al.* combine content-based signals (event description), collaborative signals (users' RSVPs), social signals (group memberships), and location and temporal signals. The authors propose a hybrid contextual learning approach where the signal values obtained for the target user-event are seen as features to then apply the Coordinate Ascent learning to rank approach.

In [124], a solution for personalized recommendation of event-based groups to users is presented. The proposed method integrates a Latent Factor Model (matrix factorization) with location and social features to identify interactions between users and groups.

As in the previous works, we present an approach that may be used to generate recommendations for users interested in participating in offline activities. However, we are not including contextual data like location; instead, we enhance the connection of like-minded users by linking several category-dependent groups. As a result, we think that our method can be used to recommend both events and meetup groups to users.

7.2.3. Graph-based Strategies to Model EBSN Users

In [125], the authors propose a graph-based unsupervised strategy to recommend a ranked list of related events that a user is likely to be interested in attending. The proposal is based on the identification of available events in pages across the entire Web. In their study, contextual features related to a web-extracted event (*i.e.* extracted field text, surrounding text of the page, taxonomy classification, related queries, etc.) are integrated into a single event-feature bipartite graph. Later, to retrieve the ranked list of events, the authors apply graph propagation methods. Relevance, recall and diversity are considered to validate this research work.

Our analysis differs in that our study focuses on a particular domain which is EBSNs instead of events found in the Web. We extend the options that users have about accessing the same kinds of events by presenting them with new groups (not dissimilar to their interests) to join. In [125], the targeting users are those who are interested in exploring related nearby events. Instead, we target users who have an offline activity and usually scheduled need in mind.

Pham *et al.* [126] propose a graph-based model to solve multiple task recommendation for EBSN users. Specifically, the authors aim at solving recommendation of groups to users, tags to groups, and events to users. They suggest the creation of a directed weighted graph by connecting five types of entities: users, events, groups, tags and venues. This model supports the three recommendation tasks which are seen as a query dependent node proximity problem. The authors uses multivariate Markov chain to solve it. In comparison to our work, we differentiate two kinds of entities in the proposed graph, members and organizers; besides, as the model is simpler and communities do not rely on user-past events links, recommendations of tags, groups, or events can be generated in a cold-start scenario.

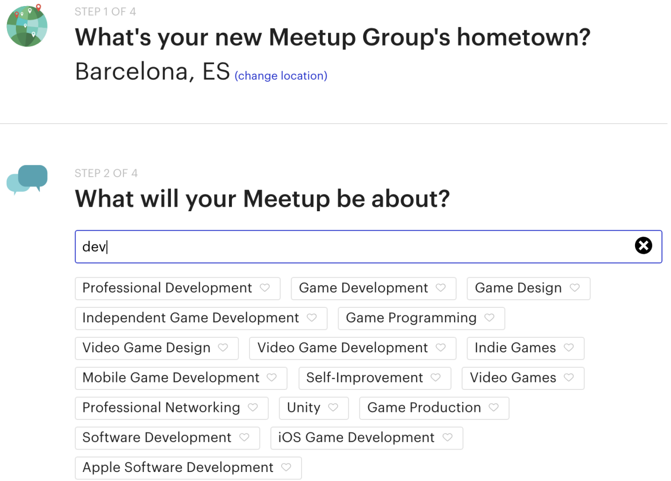
7.3. Meetup Social Network

Meetup was launched in 2002, and in November 2017 WeWork acquired the meeting platform. Since the acquisition, *Meetup* has an average of 37M visits per month.[1] Promoting offline meetings between people is the main goal of *Meetup*. Thus, the platform encourages its users to build real relationships. On the other hand, it supports professional communities and social movements. Correspondingly, in a month 3.4M RSVPs (rpondez s'il vous plat) or responses of "yes" or "no" from the invited people to events are registered. *Meetup* enable users to search and create meetups which are classified in different categories. Regarding the offered functionalities relevant in our research, next we present a brief introduction.

7.3.1. Start a New Meetup

Once the users are registered, they are able to take the role of *organizers* and create meetups. The platform guides them through the necessary steps to start a new meetup, which is conditioned by the creation of a

[1] https://www.similarweb.com/website/meetup.com



STEP 1 OF 4

What's your new Meetup Group's hometown?

Barcelona, ES (change location)

STEP 2 OF 4

What will your Meetup be about?

dev

Professional Development ♡ Game Development ♡ Game Design ♡
Independent Game Development ♡ Game Programming ♡
Video Game Design ♡ Video Game Development ♡ Indie Games ♡
Mobile Game Development ♡ Self-Improvement ♡ Video Games ♡
Professional Networking ♡ Unity ♡ Game Production ♡
Software Development ♡ iOS Game Development ♡
Apple Software Development ♡

Figure 7.1: Creation of a new meetup and its tagging with topics.

group. Having the meetup group launched, the organizer can propose *events* or *meetups*. In the creation process, the meetup group is labeled by the organizer with the appropriate topics (up to 15 topics). Figure 7.1 shows an example of how a group can be linked to certain tags/topics. A name for the meetup and a description of the target people who would like to join must be provided as well. The organizer subscription to start and lead *Meetup* groups has a cost. It depends on the monthly subscription plan that can be 'basic' or 'unlimited'.

Meetup operates in more than 180 countries and depending on the city, there are from tens to thousands of meetup groups. For example, as of May, 2018, in Quito, Ecuador, there are 63 groups. Nonetheless, in New York we find 15632 meetup groups. The most popular group[2] in Quito, 'Lean Quito: Development of innovative products', has a total of 1540 members, while the biggest group in New York, 'NY Tech Meetup', has 58671 members. It is worth mentioning that 'membership' in Meetup makes reference to the subscription of registered users in a group.

[2] We refer to the term *popularity* to describe meetup groups based on number of members or participants the group has.

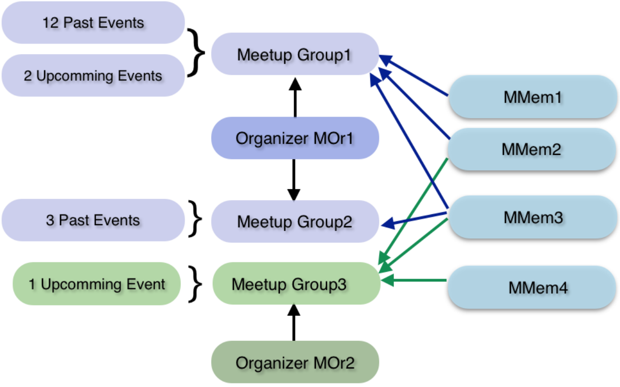


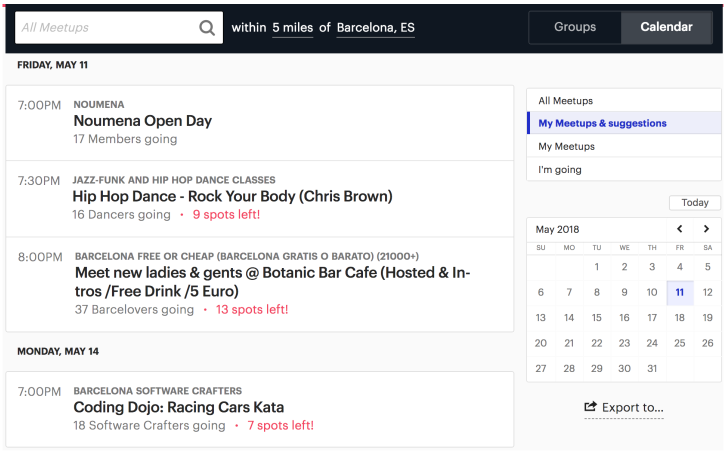
Figure 7.2: Meetup Group Organizers and Group Members.

To understand better the dynamics between Meetup organizers and members, refer to Figure 7.2. In the diagram, we exemplify the roles of Meetup users where those who are the 'organizers (*MOr*)' are able to propose and make public social events trough a given group, and the 'members (*MMem*)' may join a group and access its scheduled events. Indeed, if we consider the Group1, that has been created by *MOr1*, we can say that it has three members, *MMem1*, *MMem2*, and *MMem3* as well as 12 past events and 2 upcoming.[3]

7.3.2. Find and Subscribe in a Meetup

The option of 'Find a Meetup' in the platform provides two alternatives for its users. The first one is to find meetups by searching groups in a specific city. The retrieved list of groups may be sort by 'Most active', 'Newest', 'Recommended', 'Most members', and 'Closest' groups. The second alternative to find meetups is to see the 'calendar' and explore the scheduled events. Figure 7.3 shows an example of the use of 'Calendar' to find meetups.

[3] As the analysis of events is beyond the scope or our proposal, Figure 7.2 does not show, for example, the RSVPs' members *MMem* for group events.



FRIDAY, MAY 11

7:00PM NOUMENA
Noumena Open Day
17 Members going

7:30PM JAZZ-FUNK AND HIP HOP DANCE CLASSES
Hip Hop Dance - Rock Your Body (Chris Brown)
16 Dancers going · 9 spots left!

8:00PM BARCELONA FREE OR CHEAP (BARCELONA GRATIS O BARATO) (21000+)
Meet new ladies & gents @ Botanic Bar Cafe (Hosted & Intros /Free Drink /5 Euro)
37 Barcelovers going · 13 spots left!

MONDAY, MAY 14

7:00PM BARCELONA SOFTWARE CRAFTERS
Coding Dojo: Racing Cars Kata
18 Software Crafters going · 7 spots left!

May 2018

SU	MO	TU	WE	TH	FR	SA
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

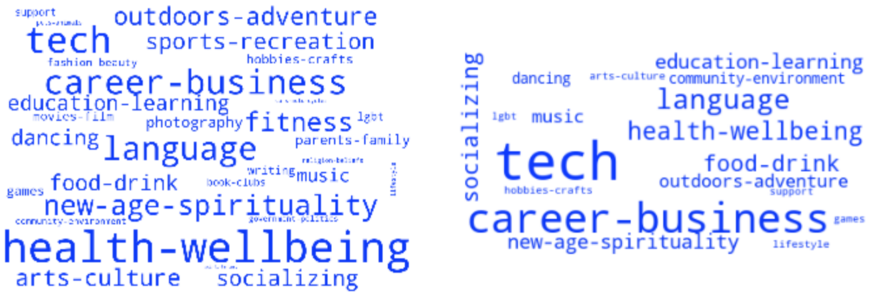
Figure 7.3: Finding a meetup by searching the scheduled events.

7.3.3. Categories for a Meetup

When the organizer has created the meetup group, it is going to be possible to add a main ‘category’ for it. The category will show the purpose of the group and the kind of events it is going to cover. Besides, the category name contains more generally the topics already chosen while creating the group. Indeed, whereas Meetup incorporates hundreds of topics (to be chosen as ‘topics of interest’ by the users or as tags for groups by the organizers), there are only 33 categories. Among the social entities found in Meetup (members, venues, messages, etc.) only the groups can have a category. However, the events proposed in the group inherit its category.

It needs to be said that a topic may be associated to more than one category. It depends on how the organizer of a group chooses the topics and then the main category. For instance, we may find groups with the topic ‘Data Science’ of which some may belong to the *Tech* category and others to the *Education and Learning* category.

In Figures 7.4a and 7.4b, we can see the level of category popularity in two different cities. In the images, the size of the name of the category is determined by the number of groups classified in it.



(a) Event Categories in Barcelona

(b) Event Categories in Santiago

Figure 7.4: Visual Representation of the Popularity of Event Categories in two locations.

For example, considering a total of 1198 groups in Barcelona and 111 groups in Santiago and in reference to the ‘music’ category, the percentage of groups belonging to it is 2.42% and 2.70% respectively. On the other hand, the categories ‘health-wellbeing’ and ‘tech’ prove to be the most popular.

7.3.4. Comparison with Twitter

It has been mentioned that the main goal of this chapter is to validate the general application of the approach presented in Chapter 6. Accordingly, before presenting the approach description in next section, we detail the ‘mapping’ of the involved concepts in Table 7.1 for a better understanding.

In this chapter we focus on the analysis of Meetup Event Categories and the users associated with these categories which are Organizers and Members. Moreover, while in Chapter 6 we worked with ‘tweets’ related to a ‘trending topic’, in this chapter we work with ‘groups’ related to a ‘category’.

Concept	Twitter (Ch6)	Meetup (Ch7)
Graph	Topic-Based	Category-Based
Communities Detected	Trending Topic Communities	Event Category Communities
Content Analyzed	Tweets	Meetup Groups
Content Generators	Tweet Creators or CCs	Groups Organizers or MOrs
Content Propagators	Tweet Distributors or CDs	Groups Members or MMems
Name of Approach	TreToC	CatCom

Table 7.1: Comparison of the Concepts employed in Chapters 6 and 7.

7.4. Approach

Following prior Meetup concepts regarding our research, we provide the details of the proposed approach named *CatCom*, which stands for "*Cat*egory *Com*munities". The approach is achieved by analyzing a *specific event category* (in a location) and following three stages:

1. **Identification of MOrs.** By analyzing the metadata of a Meetup group, it is possible to get the information of the *group's organizer* (*MOr*), *i.e.*, who created the Meetup group, gave it a name and categorized it.

2. **Identification of MMems.** Having the 'group_id' of a given group, we are able to obtain the information of the users who have subscribed to it; *i.e.*, *group's members*. The *MMems* will be notified about groups' upcoming events.

3. **Detection of Category Communities.** Given the collection of users obtained in the previous two steps, who are related to a particular event/group category, we first generate a graph G that connects them. Then apply a community detection algorithm to detect communities associated to the considered event category.

The implementation of the previous tasks is as follows:

7.4.1. Identification of Meetup Organizers

In Event-Based Social Networks, EBSNs, there are users who create groups aiming at promoting social events and gathering like-minded people to participate in offline activities. We have call them Meetup Organizers or *MOrs*.

Concerning the popularity or success of a group/event, the assumptions about the relevance and influence of the organizers' job were raised earlier in Section 7.2. Then, we think that their analysis may be significant when designing EBSNs recommenders.

Given a particular location or city, we collect the set of meetup groups G that have been created and that are public. Then, addressing our analysis to a specific category $c \in C$, we identify the groups that belong to c; *i.e.*, G_c. More formally,

$$G_c = \{g \in G : category(g) = c\}$$

where $category()$ is a function that returns the kind of events category supported by the group g.

There is a set of users U_c associated to G_c. Therefore, given the previously defined set, we designate as $MOrs \subseteq U_c$ the collection of *meetup organizers*. In summary, $MOrs$ are the users who produce the content that is consumed in EBSNs which is 'events'. Formally, the set of organizers is defined as follows:

$$MOrs = \{u \in U_c : \exists g \in G_c \; s.t. \; organizer(g) = u\}$$

where $organizer()$ is a function that returns the organizer of a given group.

7.4.2. Identification of Meetup Members

Most users who are registered in an EBSN are interested in social activities that facilitate meet people while participating in those real world events. To do so, the EBSN offers them the possibility to join groups whose category advertise the kind of gatherings. We have call these users Meetup Members or *MMems*. If the group category is of the user's interest, s/he may subscribe to the group and later receive the invitations to the group's meetups. Generally, members of groups are the users who consume the platform's information and are exposed to 'groups to join' and 'events to attend' recommendations.

Considering the success of a group, we may observe the number of its members as a factor that affects the group's impact. Then, the group's category c becomes relevant in the platform since many people keep subscribing to it and consuming the proposed events. Assume that every user $u \in MOrs$ has created a group $g \in G_c$. We define as consumers or *group members* ($MMems$) the set of users who subscribe to g. More specifically, the set is defined as follows:

$$MMems = \{u \in U_c : \exists g \in G_c \; s.t. \; member(g) = u\}$$

where $member()$ is a function that returns true if u is member of g.

7.4.3. Detection of Event Category Communities

Given the collection of users who create meetup groups (*MOrs*) and those who participate of the group's events (*MMems*), the first goal is to find an adequate way to link them. Indeed, in order to allow the success of a group and its stable permanence in the EBSN, users who are not members yet in the group might be aware of its events; thus, we have to connect them. Even though there is no explicit connection between users in Meetup, we can infer the links by studying its entities conditions.

In order to detect the communities related to a Meetup category $c \in C$, it is first necessary to build a *directed* graph $G = (V, E)$ that represents the previously mentioned connections.

The set V of vertices is represented as the union of the two sets of users identified in the previous two steps:

$$V = MOrs \cup MMems$$

In order to build the set E of edges that represent the connections among the users, we consider three types of relationships. The first is the 'founders' relationship (or F) between two category-dependent meetup organizers. We believe that two organizers who share their interest of promoting events in a given category may have members in common (subscribed in organizers' groups) as well.

$$F = \{(u_x, u_y) : share\_members(u_x, u_y) = true, u_x, u_y \in MOrs\}$$

where $share\_members()$ is a function that returns true if the group of the first organizer has at least one member who has also joined the second organizer's group.

The second type of connection we elicit is the 'participants' relationship (or P) between two category-dependent meetup members. As before, we propose to link two users who hold the role of members if they are subscribed to the same groups (two or more groups):

$$P = \{(u_x, u_y) : share\_groups(u_x, u_y) = true, u_x, u_y \in MMems\}$$

where $share\_groups()$ is a function that returns true if u_x and u_y are found as members of at least the same two groups.

In the third type of connection we link an *MOr* to an *MMem* only if the *MMem* has joined to a group created by the *MOr*. Note that to be in agreement with our scope (connections between users), the attendance behavior of *MMems* to the past events organized by *MOrs* is not represented in the category-based graph. We argue that if users join a specific group, the action alone shows the user interest in the group's events and group category. Moreover, our focus is to detect communities in which the users get in touch with agreeable content with respect to the considered category. Then, the connection between an *MOr* and an *MMem* is well represented by a 'membership' (or M) link. More formally, the set can be defined as follows:

$$\begin{aligned} M = \{(u_x, u_y) : \exists g \in G_c \ s.t. \\ member(g) = true \ \wedge organizer(g) = u_y\} \end{aligned} \quad (7.1)$$

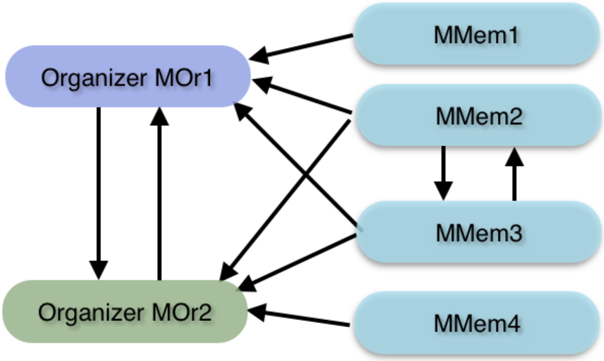


Figure 7.5: Example of the Structure of CatCom Graph Implementation.

Finally, the set E of edges in the graph is represented as:

$$E = F \cup P \cup M$$

To have a visual scheme of our framework, let us refer to Figure 7.2. Suppose that groups 1, 2 and 3 belong to the same category c. Then, after the application of the *CatCom* approach, users connections would be established as Figure 7.5 describes it. In the figure, the relationships F, P and M defines the structure for the graph G_c.

Once having implemented the graph G, we apply the Louvain method [117] to detect category-dependent communities of interest. The reasons that motivate the use of Louvain are detailed in Chapter 6, Section 6.3.3 as it was employed to detect the $TreToC$ communities.

7.5. Analytical Framework

This section presents the analytical framework and obtained results. We first present the analytical strategy and setup (Section 7.5.1), followed by a description of the employed dataset (Section 7.5.2) and metrics (Section 7.5.3). Finally, we present the analytical results (Section 7.5.5).

7.5.1. Analytical Setup and Strategy

We prepared the environment of our work in Python 3.6. To build and analyze the category-based graph we work with the modules *NetworkX*[4] and *community*.[5]

The following considerations were made:

- The graph is defined as *directed* and *unweighted*;
- To define the communities the function employed was *community.best_partition()* where the *resolution* parameter is set to 1 and the original directed graph needs to be transformed into 'undirected' to be employed as argument.[6]

To validate our proposal, four sets of analyses were performed:

1. **Characterization of the Group Categories.** Given a category, we analyze the number of groups, organizers and members that characterize it. This will give us an insight about the participation of users in the platform and its relation to the different group/event categories.
2. **Analysis of the cohesion among the users.** For each community, we evaluate its quality by measuring the cohesion between the users in it, using standard metrics such as modularity, ratio between the number of communities and the number of users, assortativity, transitivity and density.
3. **Analysis of the community structure.** For each community, we analyze its composition, by measuring the ratio of organizers and members in it, and their clustering coefficient. This allows us to evaluate the effectiveness of our approach to connect those who generate content to be consumed to those who make use of it.

[4] https://networkx.github.io

[5] http://perso.crans.org/aynaud/communities/api.html

[6] Note that, even though the communities were detected on the undirected graph, the metrics to evaluate their quality were measured on the original directed graph, as described in Section 7.4.3.

4. **Analysis of the exposure of a graph's node to diverse topics.** It was said before that on Meetup users can add topics of interest to their profiles while registering. Through our method, users are being connected depending on their common interest in a category. However, assuming that each user has other topics of interest reported in their profiles, personalization challenges may be overcome. For instance, we think that recommendations based on those other topics that are being implicitly embedded in a users' community could introduce access to diverse events or give alternatives to cold-start users. Therefore, we analyze the ratio of topics to which a community's user would be exposed.

To validate the proposed stages of our approach, we compare the $CatCom$ method with a baseline approach named Membership-Based Communities or *MBC*. In *MBC*, the set of edges in the graph connects two users only if one (the $MMem$) has joined the group created by the other (the MOr). Then, the connection between an *MOr* and an *MMem* is well represented by a 'membership' (or M) link. Refer to the Equation 7.1. Considering the lack of links such as 'following' or 'friendship' in Meetup, the natural connection between the users that is based on the groups categories drifts in *MBC*. To have a visual scheme of the baseline *MBC*, let us refer to Figure 7.2. Suppose that groups 1, 2 and 3 belong to the same category c. Then, after the application of *MBC*, users connections in the produced graph would be established as Figure 7.6 describes it. In the figure, the relationship M defines the structure for the graph G_c.

7.5.2. Meetup Dataset

Meetup provides an API for developers to extract publicly available data stored in the platform. The entities we can get information about are users, groups, events, venues, photos, profiles, rsvp, categories, messages and cities.[7] Also, Meetup has prepared a 'ready to use' web site where

[7] Datasets and code are published in `https://github.com/lore10/meetup_project` to facilitate reproducibility of our research.

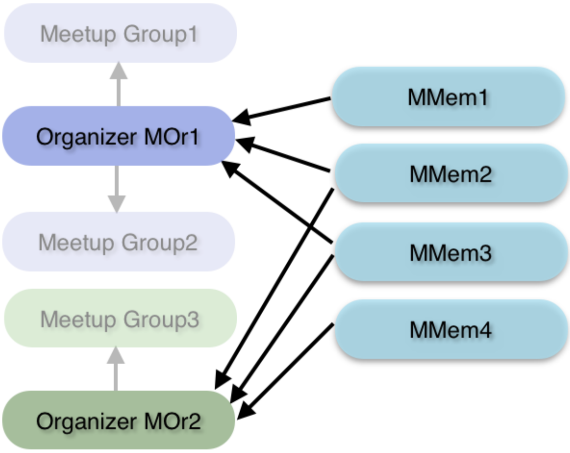


Figure 7.6: Example of the Structure of an MBC Graph.

developers can test and verify queries to the API through real time calls in the console.[8] By using this interface we are able to see the description of the API methods, the input parameters (required and optional) and the kind of metadata that would be returned.[9]

The recommendations proposed by Meetup for its users are ‘location-dependent’. Indeed, groups or events are presented after the user has confirmed his/her *city*. Moreover, literature review has shown that this factor is important when doing research on EBSNs. Therefore, we oriented our study to analyze category-related communities in nine cities: Barcelona, Madrid, Buenos Aires, La Plata, Bogota, Medellin, Mexico City, Lima and Santiago.

We collected the groups created in the nine cities in the year from April 2017 to March 2018. From the 33 existing categories on Meetup, 32 were found after aggregating the categories of the cities’ groups. The only category which did not include any groups was “Paranormal”.

[8] Meetup API: `https://www.meetup.com/meetup_api/`

[9] API console example to find topics: `https://secure.meetup.com/meetup_api/console/?path=/find/topics`

To collect and preprocess the data we performed the following steps:

- Extraction of groups given a city. Method *GetGroups(city, country)*;
- Selection of groups created in the period April 2017-March 2018 (*created* field included in the group's metadata);
- Classification of selected groups according to their category (*category* field included in the group's metadata);
- For the chosen groups, find the organizer's id (*organizer* field included in the group's metadata);
- For the chosen groups, find the members' id. Method *GetMembers(group_id)*

In total, the dataset contains 3928 public groups, 2585 unique organizers (*MOrs*) and 186372 unique members (*MMems*). Note that there is a small quantity of users who match both roles; in other words, the intersection of organizers and members gives a total of 1468 users.

In conclusion, the total number of users in our study was 187489, having 0.6% of them 'only' as organizers, 98.6% only as members, and 0.8% acting as both.

7.5.3. Metrics

The method we propose produces a graph for a Meetup category being analyzed. The graph is then divided into category-related communities. Both the graph and its communities can be evaluated by using the following metrics.

Cohesion among users

After executing the community detection algorithm, every node in the graph is going to be assigned to a community.

Later, we can calculate next metrics.

Modularity. This is a value that represents the strength of division of a network into communities. High modularity means the connections between the nodes within communities are dense and the connections between nodes in different communities are sparse [117].

Ratio between the number of communities and the number of users. This allows us to evaluate the ability of our approach to group the individual users into communities. Indeed, higher values represent a low cohesion among the users (they are not added to the same community), while lower values indicate a smaller number of communities and higher cohesion among the users.

Degree Assortativity. An assortative behavior is presented when nodes with similar degree tend to connect to each other (Equation 21 in [127]). In general, in social networks, nodes tend to create links with other nodes with similar degree values. On the other hand, *disassortativity* shows high degree nodes attached to low degree nodes. The more disassortative the network more low degree nodes are being connected to a high degree node; then, cohesive components may be found around few cores.

Transitivity. Metric that implies that, if i is connected through an edge to j, and j is connected to h, then i is connected to h as well [128]. Transitivity depends on the number of triads or subgraphs formed by three nodes in the network; then, it shows cohesion. In real networks, it is rare to have high transitivity since it implies that each component is a clique, that is, each pair of reachable nodes in the graph would be connected by an edge.

Density. It is the ratio between the number of edges per node to the number of possible edges.[10]

Community structure

Number of organizers per community. Every community is expected to have more than one organizer in order to have more groups and therefore, more event calls that can be spread through the community. The

[10] This is equivalent to the clustering coefficient of a 1 hop neighbourhood in a graph.

number of organizers per community quantifies how many founders of groups we can find in a community. Indeed, if in the community structure we find more than one organizer, more groups are going to be found; then, they and their events can be suggested for those with the role of *members* in the corresponding community.

Number of members per community. This measures how many users interested in attending events or joining groups we can find in a community.

Community clustering coefficient. This quantifies the extent to which nodes in the graph tend to cluster together. It is said that this metric is a structural feature which characterizes small-world networks [129].

Exposure to diverse topics

The last metric we are going to use, is the ratio of topics that a user in a community is indirectly exposed to. It is expected that more users in the community would bring on more different topics. To calculate this value we considered the number of nodes (or users) in the community divided by the number of unique event topics registered in their profiles (after aggregating all the users' topics).

7.5.4. Analytical Results

In the following subsections, we provide a detailed evaluation of our proposal.

Characterization of the Group Categories

In the following, we analyze what characterizes the Meetup categories in the cities' datasets. Figure 7.7 represents the number of groups found per category. The category names are organized according to their number of groups. The graphic also shows the gathering behavior or meeting dynamics of users in the cities. For example, the category 'health-wellbeing' seems to be one of the favorites in Barcelona and Madrid. However, if we visit the rest of the cities we are going to find very few groups that fall in

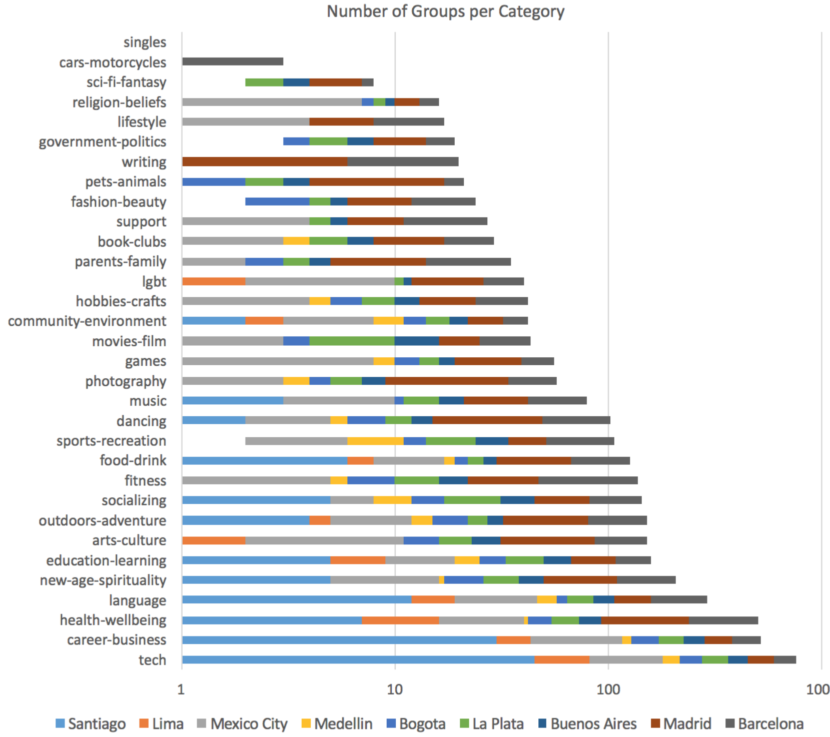


Figure 7.7: Number of Groups per Category in the Cities (log scale).

this category. We can observe that in these cities, the use of the platform is more professional-oriented than leisure-oriented. Indeed, 'tech' and 'career-business' gather the largest number of groups, particularly for the cities other than Barcelona and Madrid. This characteristic is illustrated in Figures 7.8 and 7.9.[11]

There is evidence that the most popular categories to create meetup groups are 'tech' and 'career-business', especially in the Latin American cities (refer to Figure 7.8). In the same way, the largest number of

[11] The values obtained in Figures 7.8 and 7.9 were obtained by dividing the total number of organizers found in the corresponding category by the total number of unique organizers in the city. Note that an organizer may have created groups in different categories.

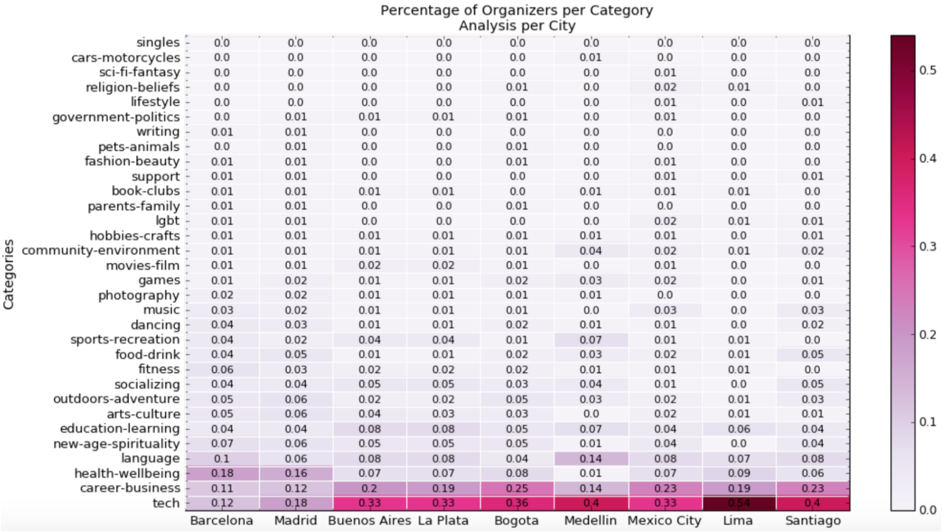


Figure 7.8: Percentage of Organizers per Category.

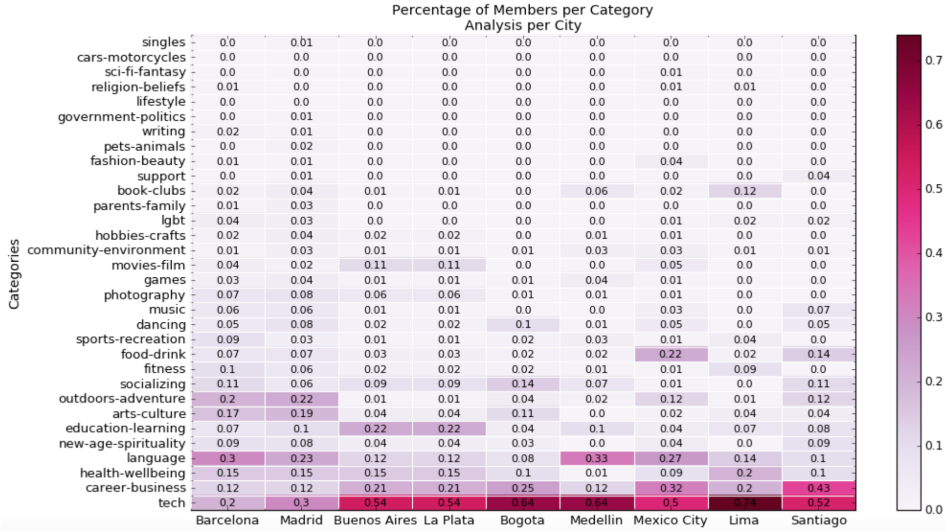


Figure 7.9: Percentage of Members per Category.

Method	Modularity	Ratio of Communities/Nodes	Assortativity	Transitivity	Density
MBC	0.611	0.010	-0.134	0.032	0.003
CatCom	0.493	0.009	-0.293	0.565	0.006

Table 7.2: Cohesion among the users for Meetup Barcelona (average).

subscribed members crowd together in these two categories (Figure 7.9). As it was commented before, 'health-wellbeing' is a relevant category in terms of group creation; *i.e.*, for the organizers in Barcelona and Madrid. Nevertheless, users with the role of member concentrate the most in 'language' groups for these two cities. It is worth observing that in Lima, three quarters of the members in the city have joined groups related to 'tech' compared with only half of the organizers who have groups in this category. In summary, we can see that the number of members in the categories increases with the number of organizers and the heatmaps (Figures 7.8 and 7.9) presented make this trend clearer.

Analysis of the cohesion among the users

In order to analyze the level of cohesion between the users in a community, in Table 7.2 we report the results for Barcelona, that is the city with the largest number of organizers and members. Table 7.2 presents the average values of *modularity*, *ratio between the number of communities and the number of users*, *assortativity*, *transitivity*, and *density*, for our approach $CatCom$ and the baseline MBC.

The corresponding metric values obtained for the nine cities in our research are presented in Figure 7.10 and Figure 7.11. The distributions reflect what is described in Table 7.2. A lower *modularity*, as that obtained in the $CatCom$ method, shows that the communities in the graph maintain certain level of interaction or connection between them. This is

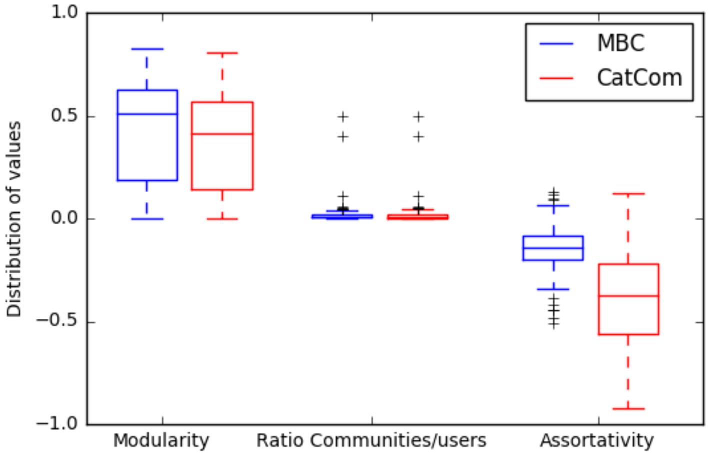


Figure 7.10: Cohesion among users: Distribution of the metric values (1).

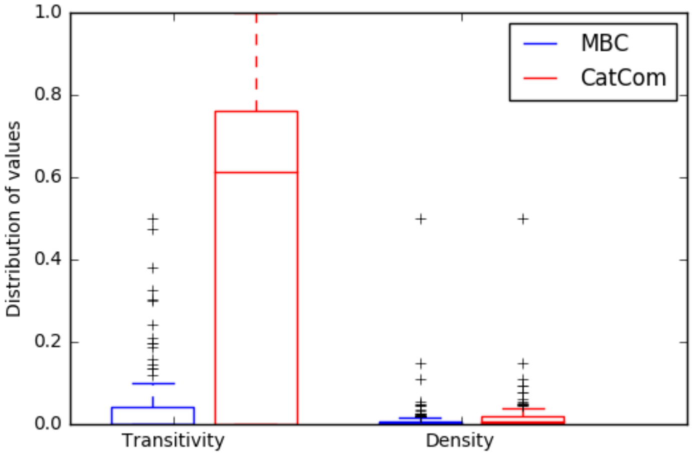


Figure 7.11: Cohesion among users: Distribution of the metric values (2).

not seen in MBC graph where the members behave only as source nodes, causing the modules partitioning to be well defined.

In this case, if we would like to adjust the resolution to get fewer communities it would not be possible because the MBC communities are not connected between them. Note that a higher modularity does not necessarily mean 'better', it is better just when we want smaller communities (in terms of number of nodes) or non-connected communities (as the MBC baseline produces). Nevertheless, the purpose of our work is to get fewer communities, which are highly associated units composed by a suitable number of organizers and members. For example, more organizers in a community would cause diversity in future recommendations.

The average number of communities found in MBC graphs is of 10 communities for every 1000 users. It exceeds in 1 unit the average amount of communities found in the $CatCom$ graphs where we have 9 communities per 1000 users. Then, our approach obtains larger communities, specially if the number of users interested in the category is of thousands.

The negative values for *assortativity* shows a more "celebrity"-driven nature; *i.e.*, there are a few extremely popular organizers (and their groups) on Meetup to whom many low degree users are connected [130]. For our method, this disassortative behavior increments (lower value for assortativity) showing that less popular organizers are probably being linked to the ones with more members. The trend of high degree nodes attached to low degree nodes can strengthen those less visible organizers by suggesting their groups to the members linked to the "celebrities".

An example that shows this behavior is presented in Figure 7.12. This graph corresponds to the city of Lima and presents the users who have shown interest (as organizers or members) in the 'language' category. The links were established by the CatCom method and 7 communities were found. We can see that the nodes with high indegree are connected to nodes with low indegree. The central node in purple color is the one with the highest degree, with a value of 703 (699 for indegree). It represents the organizer whose name is *GMAT TOEFL English* and manages three groups in the 'language' category in Lima. The vast majority of the users linked to it have a degree of 1; thus, the "celebrity"-driven nature.

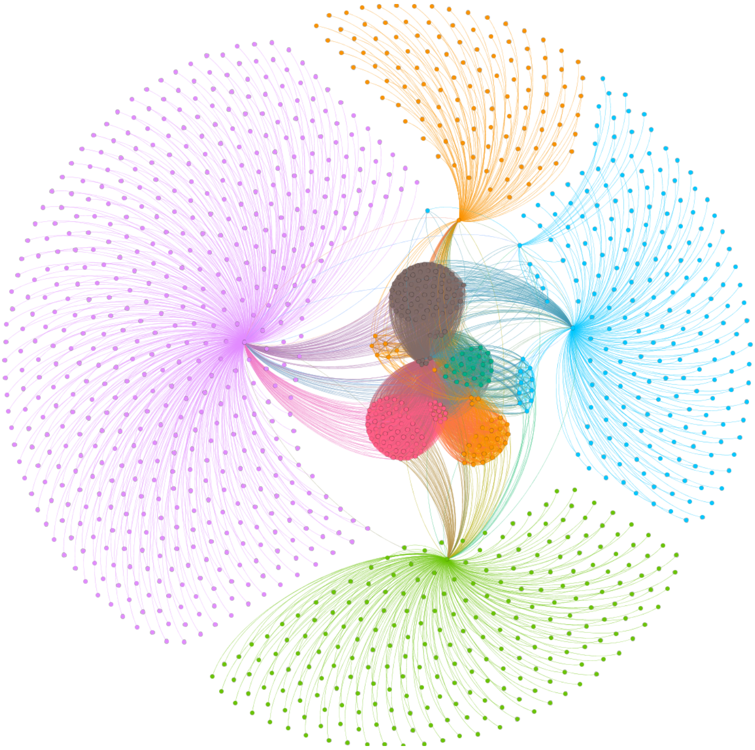


Figure 7.12: Example of category graph that shows a “celebrity”-driven nature.

The transitivity for the graphs built with $CatCom$ is in average 0.61. This means that, on average, the chance that three (or more) members that share a common organizer subscribe to another organizer’s group is almost two-thirds. In such case, these three members would have a connection between them and create a triad. On the other hand, triads are not expected in a graph generated by MBC due to its different structure.

The *density* is influenced by the level of cohesion of users, and since the users in $CatCom$ share three types of links (*i.e.*, ‘founders relationship, ‘participants relationship, as well as membership) they act as source or destination nodes, resulting in a greater density when compared to MBC.

Method	Num of Creators/Comm	Num of Distributors/Comm	Clustering Coefficient
MBC	20.613	2185	0.014
CatCom	43.684	4987	0.144

Table 7.3: Community structure in Meetup Barcelona (average).

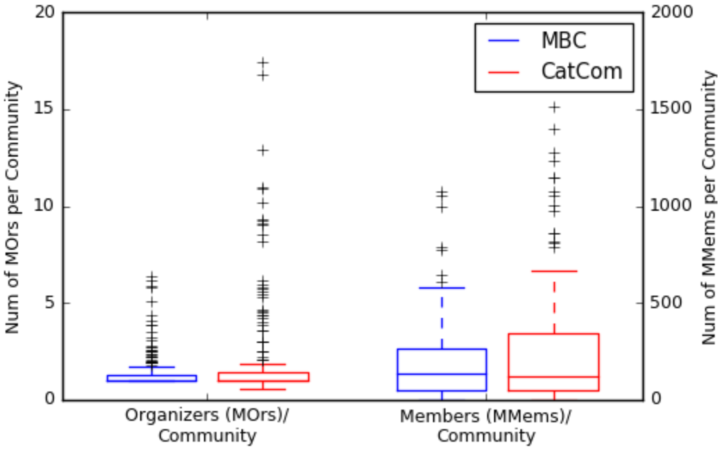


Figure 7.13: Community structure: Number of Organizers and Members per community.

Analysis of the community structure

Table 7.3 shows, in terms of general results, a summary of the analysis of the community structure in Barcelona. More specifically, this analysis measures the *average number of organizers per community*, the *average number of members per community*, and the *average clustering coefficient*. A graphical representation of the statistics is presented in Figures 7.13 and 7.14.

The MBC method relates less than seven organizers in one community; *i.e.*, just in cases where two organizers have members in common,

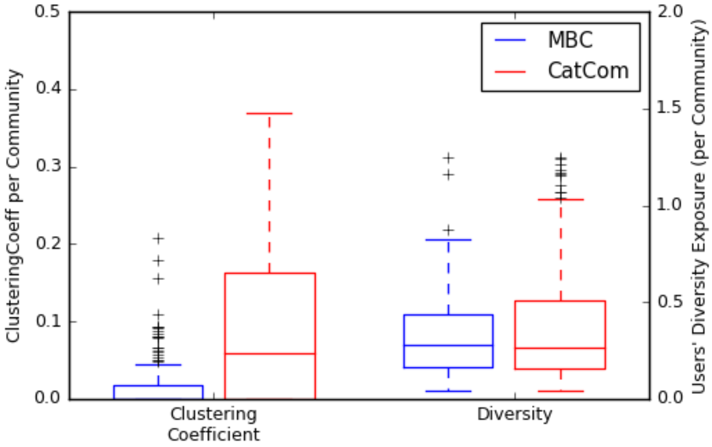


Figure 7.14: Community structure: Distribution of the metric values.

the community detection algorithm is going to be able to put them together in a community.[12] In the $CatCom$ method, the inferred relationship F joins organizers to each other, making it more likely to find them as close neighbors. Consequently, their individual members come together too. Considering the members per community, despite that in $CatCom$ method the number of members in the communities have a slight increase, the number of members in average does not increase radically compared to MBC.[13] In general, being able to have more users linked together in a $CatCom$ community, the *clustering coefficient* increases as well in comparison to the MBC graph (Figure 7.14). Note that in the Membership-Based Communities there is only one kind of connection that relates members with organizers. Then, in the MBC communities

[12] Something that needs to be noted is that to facilitate a the representation, the values for ‘organizers per community’ and ‘members per community’ have their corresponding *y axis*.

[13] A further deep study by city would reveal conclusive motives considering that in Barcelona (Table 7.3), the number of members per community doubles by implementing $CatCom$.

the vast majority of communities have a clustering coefficient less than 0.1. On the other hand, in $CatCom$, 50% of the communities have a clustering coefficient between 0.07 and 0.38.

To deal with readability, in Figure 7.14, we also report the results of the degree of access to diverse topics in a given community. This finding is explained next.

Analysis of the exposure of users to diverse topics

Let us suppose that the category c is associated to 100 topics.[14] The bigger the communities found for c, the bigger the possibility to find all the possible 100 topics among the interests of the community's users. Considering Meetup Barcelona data as a general example, we find that in a $CatCom$ community in average, 449 users would get in touch with content related to a given topic. On the contrary, in a MBC community, only 319 users would access the same information. Figure 7.14 presents the values for this metric obtained by aggregating the cities' results. The median value for the two approaches do not change, showing that commonly, a user's community has a exposure to a topic of 0.25. However, in $CatCom$, in 25% of the communities the ratio of exposure of a user to a topic increases slightly.

7.5.5. Approach Performance: Twitter vs Meetup

In this section we compare the performance of the proposed approach which takes the name of *TreToC* when applied on Twitter Trending Topic graphs and the name of *CatCom* when applied on Meetup Event Category graphs. The approach allows finding the corresponding communities in which generators and propagators of content coexist. The presence of propagators (whether distributors or consumers of content) exceeds the number of creators in both environments, and this gap may grow depending on the topic/category. Indeed, if the topic being tweeted is more

[14] For example, the category 'health-wellbeing' covers Meetup topics like ['yoga', 'yoga sutras, 'life transform', 'nutrition', ...].

trendy, the number of distributors increase exponentially. In Meetup, if the category is more popular, more members who have joined the category group are going to be found.

The graph-oriented metrics we employed to analyze the graphs targeted characteristics such as 'cohesion among the users' and 'community structure'. The approach results were the same in both cases; for instance, among the main metrics we may mention the modularity, which was lower compared to the baselines (RBC and MBC) and demonstrated that the communities were highly associated units that nevertheless maintained connections with each other. The density of the two kinds of graphs is in average 0.003, showing that the presence of edges is minimal respect to the *possible* number of edges. However, the density for the baseline graphs is lower than 0.001. Concerning the number of creators and distributors found in a TreToC community, we may say that there is more increase effect than in CatCom. Similarly, the clustering coefficient in TreToC has an average of 0.077 while for CatCom it is 0.05. We think that this difference might be related to the graphs structure. For example, the average number of users in a TreToC graph is 153.5 but in CatCom it is 933. Sections 6.5 and 7.6 provide more detail and a discussion of the corresponding results.

7.6. Discussion

We now summarize the findings obtained in our analysis. Firstly, when working with Meetup categories, Section 7.5.4 showed us that depending on the dynamics of a city a category may be more or less attractive for the users. Even though a positive correlation between the percentage of organizers and the percentage of members in a category is observed, there might be unexpected cases like in Mexico, for the category 'food-drink'. Indeed, the results show that 2% of organizers in the city have created groups that are preferred for the 22% of the Mexican members. Regarding this example, there are nine groups for 'food-drink' in Mexico City (seven organizers). The group with the largest number of members (3496) has the name of *Language Bar* (or Bar de Idiomas in Spanish) and promotes events for language interchange that take place in that bar. Secondly, our analysis of cohesion among users in category-based graphs (Section 7.5.4) draws on prior insights about the need of different kinds of connections between organizers and between members. We adopt relationships like 'founders or F relationship and 'participants' or P relationship to extend the 'membership' relation considered in the baseline MBC. Then, we could evaluate the proposed method $CatCom$ and compare it with MBC. The inclusion of those links improved the cohesion of nodes in the category-based graphs which was shown by the reports of metrics such as modularity, ratio between the number of communities and the number of users, assortativity, transitivity and density.

The third analysis, which studied the structure of the communities (Section 7.5.4) showed us that each community contains in average 2.125 MOrs and 293.468 MMems per community (facing 1.39 MOrs and 192.241 MMems of MBC communities). This would allow the users interested in attending events to get in touch with different organizers who pertain to the same category but schedule events in a variety of topics. Moreover, the clustering coefficient results confirmed that the users in the communities tend to cluster well together (the values are high compared with the baseline). The last analysis showed that, even though some topics of interest are related and share the same category, if the users are in some way aware of them, they can introduce a certain degree of diversity in terms of the groups they join and the events where they participate. The results summarized above were obtained by aggregating the calculations made per city in order to see the metrics distribution. These prior conclusions are thus generalized. However, some Meetup categories have few groups, so it may be necessary to verify if the proposed method is as effective as on popular categories (those with a large number of groups) as on the least favorite.

Chapter 8, which is the following, presents the conclusions and future directions for our work.

Chapter 8

CONCLUSIONS

In this thesis, we have tackled problem of modeling users' preferences in Online Social Networks and oriented our studies to challenges like personalization, recommendation and group detection.

In Part I (comprising Chapters 3 to 5), through the analysis of microblogging user-generated content, we were able to propose some approaches which make it possible to efficiently model the preferences of users for different kinds of content. We worked on methods whose core implementation depends on neural language modeling or 'word embeddings' and target a specific field of application that is E-Government. For instance, we proposed the study of digital citizens and their level of engagement in politics by modeling their Twitter posts. Also, given a scenario of chaotic political situation in a democracy, we open the possibility of monitoring the political tendency of users and their exposure to the misuse of political-related hashtags. Broadly speaking, the unsupervised or semi-supervised approaches proposed in the first part of the thesis combine techniques such as text preprocessing, probabilistic clustering, and linear operations on vectors.

In Part II of the thesis (comprising Chapters 6 and 7), we presented an approach based on graph theory and related metrics. The method aims at detecting topic/category -based communities where users who generate content are linked to the users who have the role of content consumers;

thus, the network structure facilitates information propagation. The problem to be addressed was inferring relevant connections among them in such a way that we can identify cohesive communities that group a suitable number of both creators and consumers. Problems of this nature occur in Social Networks because they feed on their users' content and interactions. The solution proposed was evaluated in Twitter and in Meetup datasets showing the effectiveness and generalization of our approach.

8.1. Main Results

In detail, the main conclusions based on the results obtained through the chapters of this thesis are:

- Chapter 3: When the users are modeled by employing a method based on word embeddings, we can learn their features in more manageable dimensions (*i.e.* 300). The detail captured makes it possible to establish relationships and structure from such data. From the model learned using as input a corpus of preprocessed short text, we worked on methods to obtain a vector representation of tweets and users (by aggregating their tweet models). In our particular case, we obtained an optimal user model that combines word embeddings and Mixture of Gaussians, which performed better than strategies that integrates text modeling with *TF-IDF* and supervised learning. We validated our approach by verifying that like-minded people had, in fact, very similar models.

- Chapter 4: It is possible to calculate the degree of interest that a user has for a specific topic by analyzing the content s/he shares. Moreover, we determine that there is some level of consistency about the users' tweets and other kinds of interactions. For example, if we consider users highly engaged with politics, we see that their friends and the lists where they subscribe share certain quantity of political-related content, as well. The reliability of strategies based on word embeddings seen in Chapter 3 motivated us to compare the performance of ‘word2vec’ and ‘GloVe’ in Chapter 4. After validations, we employed ‘word2vec’ to create not only the users model, but also we defined a (corpus-dependent) political vector space representation.

- Chapter 5: The presence of hashtag misuse is unavoidable in Twitter especially when there are groups of users with opposing views/positions. However, hashtag hijackers may be detected and recommended as Twitter accounts ‘not to follow’. In the context of politics, the identification of the users’ political affiliation, the hashtags that they commonly use to support this affinity, and the users who hijack those hashtags (with a malicious behavior or to express sarcasm) is of interest in areas like E-Government and Group Recommender Systems.

  Regarding their use, hashtags may be labeled after the political groups that promote them (supervised approach). For instance, if two political tendencies are prevalent in a given location, the corresponding hashtags can be classified as government ‘supporting’ hashtags and government ‘opposing’ hashtags. We show that combining Twitter content (words, hashtags, mentions and emoticons) modeled through word embeddings with labeled data, it is feasible to measure, for example, how similar a ‘supporting’ hashtag is with respect to the rest of the tweet. Thus, dissimilar calculations disclose hashtag hijackers.

- Chapter 6: To facilitate users’ interactions and the spread of new information, people who create content require to be linked with people who propagate it, and they should be part of the same topic community. Users who create content generate value and bring in new ideas into the network as long as they are surrounded by content distributors. Moreover, content diffusion depends on its trendiness and quality which can be potentiated if ‘creators’ are exposed to other creators’ posts.

  We showed that topic-oriented graphs that implement different kinds of relationships between the users like ‘retweeting’ and ‘following’ may hide cohesive communities. So, this kind of structure may be used to generate community-based recommendations. In this chapter, we worked with Twitter data and validated the effectiveness of our approach at identifying the proper links between the users who participate in the evolution of a trending topic and then to detect suitable communities, which contain both creators and distributors.

- Chapter 7: Given that in Meetup there are no explicit connections between the users, Group Membership is the most relevant signal to extract organizer-member interactions. Therefore, by generalizing the method proposed in Chapter 6, we have proposed a category-based framework to bring together event organizers and members. We have presented useful insights about the correlation between the presence of organizers and the number of members depending on the location and the popularity of a category. To target the *event* and *group to join* recommendations we suggest a community-oriented modeling of the users which could introduce a certain diversity among the users interested in a given event category.

In summary, we have presented different approaches to represent users regarding their preferred topics by making use of their published content and the connections that they have with others in a social graph. We saw that what users share in their profiles leads to discover their topics of interest in the short and medium term. Text models based on word embeddings have to be able to incorporate the users’ context; *i.e.*, language, commonly used abbreviations, location-dependent expressions, etc. Therefore, the training corpus has to be carefully defined. Once the users are represented by their models, they can be targeted with recommendations, or grouped with similar ones and seen as units of information propagation. The proposed approaches create bases to combine strategies like neural language modeling, probabilistic clustering, graphs and community detection to discover hidden users’ interests and their relation with others.

8.2. Future Work

The work of the thesis has suggested some new possible future directions which are detailed next.

- Part I: As it was said before, the most important step is the definition of the training dataset. Then, for future work, we consider updating the vocabulary obtained with *word2vec* algorithms due to new topics/hashtags that appear over time. Indeed, as the nature of user content-generated is hardly constant and its amount increases day after day, users models should be adaptive in some extent [131].

  We think that our method can be used in recommender systems to find new content and subscription lists that match the users profiles. We propose for further research to label the groups of users and to apply a validation of *word2vec*-based models not only in the application field of politics but also in 'researcher communities', 'sportmen', and so on. Also, we plan to evaluate our approach with other probabilistic topic models like LDA and test its performance at topic assignment for short texts.

- Part II: Users sessions and contextual factors could be useful to observe the consumption behavior of items and evaluate which factors influence their adoption. Establishing 'weighted' graphs where not only users are linked but also topics or hashtags, categories and places, among others would open new problems and therefore new ways to solve topic community-based recommendations.

  The *cold start* problem is evident for "first time" target users given that we cannot obtain a profile based on their posts or social ties.[1] Considering our approach, these kind of users would be disconnected from the graph. It would be necessary to propose strategies

[1] The term cold-start is used in recommender systems to identify the problem caused by the initial lack of user's interests information when he has just registered in the system, so there are no rates or historical feedback known to build the user preferences model.

to measure communities *popularity* to try to mitigate this problem by generating initial recommendations.

To conclude, we have focused exclusively on the processing of user generated posts and their social links with others who have shown explicit interest in a topic/category. For many user actions, it would be useful to enhance the knowledge already extracted. For example, sentiment analysis (and the use of specific dictionaries) can be studied in combination with political alignment and hashtag misuse. Moreover, in topic-based graphs the identified communities could show ‘sentiment’ patterns as well. To verify the general applicability of our apporaches and their limitations, we plan to work with datasets that represent other domains than politics and which may be obtained from diverse OSNs.

Bibliography

[1] D. Jannach, M. Zanker, A. Felfernig, and G. Friedrich, *Recommender Systems: An Introduction*, 1st ed. New York, NY, USA: Cambridge University Press, 2010.

[2] M. Cha, A. Mislove, B. Adams, and K. P. Gummadi, "Characterizing Social Cascades in Flickr," in *Proceedings of the First Workshop on Online Social Networks*, ser. WOSN '08. New York, NY, USA: ACM, 2008, pp. 13–18.

[3] E. H. Chi, "The Social Web: Research and Opportunities," *Computer*, vol. 41, no. 9, pp. 88–91, 2008.

[4] R. E. Kraut, P. Resnick, S. Kiesler, Y. Ren, Y. Chen, M. Burke, N. Kittur, J. Riedl, and J. Konstan, *Building Successful Online Communities: Evidence-Based Social Design*. The MIT Press, 2012.

[5] A. Q. Macedo, L. B. Marinho, and R. L. Santos, "Context-Aware Event Recommendation in Event-Based Social Networks," in *Proceedings of the 9th ACM Conference on Recommender Systems*, ser. RecSys '15. New York, NY, USA: ACM, 2015, pp. 123–130.

[6] F. Ricci, L. Rokach, and B. Shapira, "Introduction to Recommender Systems Handbook," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer US, 2011, pp. 1–35.

[7] R. Burke, "Hybrid Web Recommender Systems," in *The Adaptive Web*, ser. Lecture Notes in Computer Science, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds. Springer Berlin Heidelberg, 2007, vol. 4321, pp. 377–408.

[8] A. J. P. Jeckmans, M. Beye, Z. Erkin, P. Hartel, R. L. Lagendijk, and Q. Tang, *Privacy in Recommender Systems*. London: Springer London, 2013, pp. 263–281.

[9] D. Kelly and J. Teevan, "Implicit Feedback for Inferring User Preference: A Bibliography," *SIGIR Forum*, vol. 37, no. 2, pp. 18–28, Sep. 2003.

[10] M. McPherson, L. Smith-Lovin, and J. M. Cook, "Birds of a Feather: Homophily in Social Networks," *Annual Review of Sociology*, vol. 27, no. 1, pp. 415–444, 2001.

[11] D. F. Nettleton, "Data Mining of Social Networks Represented as Graphs," *Computer Science Review*, vol. 7, pp. 1 – 34, 2013.

[12] H. Tajfel, *Social Identity and Intergroup Relations*. New York, USA: Cambridge University Press, 2010.

[13] M. O'Connor, D. Cosley, J. A. Konstan, and J. Riedl, *PolyLens: A Recommender System for Groups of Users*. Dordrecht: Springer Netherlands, 2001, pp. 199–218.

[14] A. Jameson and B. Smyth, "The Adaptive Web," P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds. Berlin, Heidelberg: Springer-Verlag, 2007, ch. Recommendation to Groups, pp. 596–627.

[15] L. Boratto, S. Carta, and G. Fenu, "Discovery and Representation of the Preferences of Automatically Detected Groups: Exploiting the Link between Group Modeling and Clustering," *Future Generation Computer Systems*, vol. 64, pp. 165 – 174, 2016.

[16] L. Recalde and R. Baeza-Yates, “What Kind of Content Are You Prone to Tweet? Multi-topic Preference Model for Tweeters,” in *Workshop on Social Aspects in Personalization and Search*, ser. SOAPS’18, 2018.

[17] L. Recalde and A. Kaskina, “Who is Suitable to Be Followed Back when You Are a Twitter Interested in Politics?” in *Proceedings of the 18th Annual International Conference on Digital Government Research*, ser. dg.o ’17. New York, NY, USA: ACM, 2017, pp. 94–99.

[18] L. Recalde, J. Mendieta, L. Boratto, L. Teran, C. Vaca, and G. Baquerizo, “Who You Should Not Follow: Extracting Word Embeddings from Tweets to Identify Groups of Interest and Hijackers in Demonstrations,” *IEEE Transactions on Emerging Topics in Computing*, vol. PP, no. 99, pp. 1–1, 2017.

[19] A. P. Dempster, N. M. Laird, and D. B. Rubin, “Maximum Likelihood from Incomplete Data via the EM Algorithm,” *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 1–38, 1977.

[20] S. Grabner-Kräuter, “Web 2.0 Social Networks: The Role of Trust,” *Journal of Business Ethics*, vol. 90, no. 4, pp. 505–522, Dec 2009.

[21] I. Guy, *Social Recommender Systems*. Boston: Springer US, 2015, pp. 511–543.

[22] K. Chen, T. Chen, G. Zheng, O. Jin, E. Yao, and Y. Yu, “Collaborative Personalized Tweet Recommendation,” in *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR ’12. New York, NY, USA: ACM, 2012, pp. 661–670.

[23] J. Chen, R. Nairn, L. Nelson, M. Bernstein, and E. Chi, “Short and Tweet: Experiments on Recommending Content from Information

Streams," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '10. New York, NY, USA: ACM, 2010, pp. 1185–1194.

[24] G. Salton and C. Buckley, "Term-weighting Approaches in Automatic Text Retrieval," *Inf. Process. Manage.*, vol. 24, no. 5, pp. 513–523, Aug. 1988.

[25] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "TwitterRank: Finding Topic-sensitive Influential Twitterers," in *Proceedings of the Third ACM International Conference on Web Search and Data Mining*, ser. WSDM '10. New York, NY, USA: ACM, 2010, pp. 261–270.

[26] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent Dirichlet Allocation," *J. Mach. Learn. Res.*, vol. 3, pp. 993–1022, Mar. 2003.

[27] M. Steyvers and T. Griffiths, *Latent Semantic Analysis: A Road to Meaning*. Laurence Erlbaum, 2007, ch. Probabilistic topic models.

[28] L. S. L. Lai and E. Turban, "Groups Formation and Operations in the Web 2.0 Environment and Social Networks," *Group Decision and Negotiation*, vol. 17, no. 5, pp. 387–402, Sep 2008.

[29] J. Surowiecki, *The Wisdom of Crowds*. Anchor, 2005.

[30] J. Masthoff, *Group Recommender Systems: Combining Individual Models*. Boston, MA: Springer US, 2011, pp. 677–702.

[31] L. Boratto and S. Carta, "State-of-the-Art in Group Recommendation and New Approaches for Automatic Identification of Groups," in *Information Retrieval and Mining in Distributed Environments*. Springer Berlin, 2011, vol. 324, pp. 1–20.

[32] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas, "Short Text Classification in Twitter to Improve Information Filtering," in *Proceedings of the 33rd International ACM*

SIGIR Conference on Research and Development in Information Retrieval, ser. SIGIR ’10. New York, NY, USA: ACM, 2010, pp. 841–842.

[33] F. Godin, V. Slavkovikj, W. De Neve, B. Schrauwen, and R. Van de Walle, “Using Topic Models for Twitter Hashtag Recommendation,” in *Proceedings of the 22Nd International Conference on World Wide Web*, ser. WWW ’13 Companion. New York, NY, USA: ACM, 2013, pp. 593–596.

[34] D. Parra, C. Trattner, D. Gmez, M. Hurtado, X. Wen, and Y.-R. Lin, “Twitter in Academic Events: A Study of Temporal Usage, Communication, Sentimental and Topical Patterns in 16 Computer Science Conferences,” *Computer Communications*, vol. 73, pp. 301 – 314, 2016, online Social Networks.

[35] Z. Iman, S. Sanner, M. R. Bouadjenek, and L. Xie, “A Longitudinal Study of Topic Classification on Twitter,” in *Proceedings of the Eleventh International AAAI Conference on Web and Social Media*, ser. ICWSM ’17, 2017, pp. 552–555.

[36] S.-H. Yang, A. Kolcz, A. Schlaikjer, and P. Gupta, “Large-scale High-precision Topic Modeling on Twitter,” in *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD ’14. New York, NY, USA: ACM, 2014, pp. 1907–1916.

[37] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality,” in *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, C. J. C. Burges, L. Bottou, Z. Ghahramani, and K. Q. Weinberger, Eds., 2013, pp. 3111–3119.

[38] R. L. Thorndike, "Who Belongs in the Family," *Psychometrika*, pp. 267–276, 1953.

[39] D. Arthur and S. Vassilvitskii, "K-means++: The Advantages of Careful Seeding," in *Proceedings of the 18 ACM-SIAM Symposium on Discrete Algorithms*, ser. SODA '07, Philadelphia, PA, USA, 2007, pp. 1027–1035.

[40] J. Macqueen, "Some Methods for Classification and Analysis of Multivariate Observations," in *In 5-th Berkeley Symposium on Mathematical Statistics and Probability*, 1967, pp. 281–297.

[41] Y. Low, J. Gonzalez, A. Kyrola, D. Bickson, C. Guestrin, and J. M. Hellerstein, "GraphLab: A New Framework for Parallel Machine Learning," *CoRR*, vol. abs/1006.4990, 2010.

[42] R. A. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval.* Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1999.

[43] Y. Freund and R. E. Schapire, "Experiments with a New Boosting Algorithm," in *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning*, ser. ICML'96. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1996, pp. 148–156.

[44] Y. Yang, "A Study of Thresholding Strategies for Text Categorization," in *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '01. New York, NY, USA: ACM, 2001, pp. 137–145.

[45] A. Herv and W. L. J., "Principal Component Analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459.

[46] K. Mossberger, C. Tolbert, and R. McNeal, *Digital Citizenship: The Internet, Society, and Participation.* Massachusetts, USA: MIT Press, 2007.

[47] C. Nunes Silva, *Citizen E-Participation in Urban Governance: Crowdsourcing and Collaborative Creativity.* Lisbon, Portugal: IGI Global, 2013.

[48] A. Haro-de Rosario, A. Sáez-Martín, and M. Caba-Pérez, "Using Social Media to Enhance Citizen Engagement with Local Government: Twitter or Facebook?" *New Media & Society*, pp. 1 – 21, 2016.

[49] J. Carlisle and R. Patton, "Is Social Media Changing How We Understand Political Engagement? An Analysis of Facebook and the 2008 Presidential Election," *Political Research Quarterly*, vol. 66, no. 4, pp. 883–895, 2013.

[50] O. Larsson and M. Hallvard, "Representation or Participation?" *Javnost - The Public*, vol. 20, no. 1, pp. 71–88, 2013.

[51] C. Vaccari, A. Valeriani, P. Barberá, R. Bonneau, J. Jost, J. Nagler, and J. Tucker, "Political Expression and Action on Social Media: Exploring the Relationship Between Lower- and Higher-Threshold Political Activities Among Twitter Users in Italy," *Journal of Computer- Mediated Communication*, vol. 20, no. 2, pp. 221–239, 2015.

[52] S. Rill, D. Reinel, J. Scheidt, and R. V. Zicari, "Politwi: Early Detection of Emerging Political Topics on Twitter and the Impact on Concept-Level Sentiment Analysis," *Knowledge-Based Systems*, vol. 69, pp. 24–33, 2014.

[53] J. Ausserhofer and A. Maireder, "National Politics on Twitter: Structures and Topics of a Networked Public Sphere," *Information, Communication Society*, pp. 291–314, 2013.

[54] L. Hemphill, J. Otterbacher, and M. Shapiro, “What’s Congress doing on Twitter?” in *Computer supported cooperative work.* ACM, 2013, pp. 877–886.

[55] E. Martínez-Cámara, M. Martín-Valdivia, L. Ureña López, and A. Montejo-Ráez, “Sentiment Analysis in Twitter,” *Natural Language Engineering*, vol. 20, no. 1, pp. 1–28, Jan 2014.

[56] M. M. Mostafa, “More than Words: Social Networks’ Text Mining for Consumer Brand Sentiments,” *Expert Systems with Applications*, vol. 40, no. 10, pp. 4241 – 4251, 2013.

[57] E. Sang and J. Bos, “Predicting the 2011 Dutch Senate Election Results with Twitter,” in *Proceedings of the Workshop on Semantic Analysis in Social Media.* Stroudsburg, PA, USA: Association for Computational Linguistics, 2012, pp. 53–60.

[58] X. Zhou, Y. Xu, Y. Li, A. Josang, and C. Cox, “The State-of-the-Art in Personalized Recommender Systems for Social Networking,” *Artificial Intelligence Review*, vol. 37, no. 2, pp. 119–132, 2012.

[59] S. Berkovsky and J. Freyne, “Personalised Network Activity Feeds: Finding Needles in the Haystacks,” in *Mining, Modeling, and Recommending ’Things’ in Social Media*, ser. Lecture Notes in Computer Science, M. Atzmueller, A. Chin, C. Scholz, and C. Trattner, Eds. Springer International Publishing, 2015, vol. 8940, pp. 21–34.

[60] J. Hannon, M. Bennett, and B. Smyth, “Recommending Twitter Users to Follow Using Content and Collaborative Filtering Approaches,” in *Proceedings of the Fourth ACM Conference on Recommender Systems*, ser. RecSys ’10. New York, NY, USA: ACM, 2010, pp. 199–206.

[61] Y. Liu, X. Chen, S. Li, and L. Wang, *A User Adaptive Model for Followee Recommendation on Twitter*. Cham: Springer, 2016, pp. 425–436.

[62] D. Kim, Y. Jo, I.-C. Moon, and O. Alice, "Analysis of Twitter Lists as a potential source for discovering latent characteristics of users," *ACM CHI Workshop on Microblogging*, 06 2010.

[63] J. Pennington, R. Socher, and C. D. Manning, "GloVe: Global Vectors for Word Representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543.

[64] P. Berkhin, *A Survey of Clustering Data Mining Techniques*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 25–71.

[65] T. Mikolov, W.-t. Yih, and G. Zweig, "Linguistic Regularities in Continuous Space Word Representations," in *HLT-NAACL*, 2013, pp. 746–751.

[66] O. Levy and Y. Goldberg, "Linguistic Regularities in Sparse and Explicit Word Representations," in *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*. Association for Computational Linguistics, 2014, pp. 171–180.

[67] P. Bonhard and M. A. Sasse, "'Knowing Me, Knowing You' – Using Profiles and Social Networking to Improve Recommender Systems," *BT Technology Journal*, vol. 24, no. 3, pp. 84–98, Jul. 2006.

[68] F. Comunello and G. Anzera, "Will the Revolution be Tweeted? A Conceptual Framework for Understanding the Social Media and the Arab Spring," *Islam and Christian–Muslim Relations*, vol. 23, no. 4, pp. 453–470, 2012.

[69] H. Gil de Zúñiga, A. Veenstra, E. Vraga, and D.-v. Shah, "Digital Democracy: Reimagining Path-Ways to Political Participation," *Journal of Information Technology & Poli-tics*, vol. 7, no. 1, pp. 36–51, 2010.

[70] A. Segerberg and W. L. Bennett, “Social Media and the Organization of Collective Action: Using Twitter to Explore the Ecologies of Two Climate Change Protests,” *The Communication Re-view*, vol. 14, no. 3, pp. 197–215, 2011.

[71] P. N. Howard, A. Duffy, D. Freelon, M. M. Hussain, W. Mari, and M. Maziad, “Opening Closed Regimes: What was the Role of Social Media During the Arab Spring?” *Available at SSRN 2595096*, 2011.

[72] A. Di Florio, N. V. Verde, A. Villani, D. Vitali, and L. V. Mancini, “Bypassing Censorship: A Proven Tool Against the Recent Internet Censorship in Turkey,” in *Software Reliability Engineering Workshops (ISSREW), 2014 IEEE International Symposium on.* IEEE, 2014, pp. 389–394.

[73] K.-w. Fu, C.-h. Chan, and M. Chau, “Assessing Censorship on Microblogs in China: Discriminatory Keyword Analysis and the Real-Name Registration Policy,” *IEEE Internet Computing*, vol. 17, no. 3, pp. 42–50, 2013.

[74] G. Lotan, E. Graeff, M. Ananny, D. Gaffney, I. Pearce *et al.*, “The Arab Spring— The Revolutions Were Tweeted: Information Flows During the 2011 Tunisian and Egyptian Revolutions,” *International journal of communication*, vol. 5, p. 31, 2011.

[75] M. Conover, J. Ratkiewicz, M. R. Francisco, B. Gonçalves, F. Menczer, and A. Flammini, “Political Polarization on Twitter,” in *ICWSM.* The AAAI Press, 2011.

[76] J. Ausserhofer and A. Maireder, “National politics on Twitter,” *Information, Communication & Society*, vol. 16, no. 3, pp. 291–314, 2013.

[77] S. Passini, “The Facebook and Twitter Revolutions: Active Participation in the 21st Century,” *Human Affairs*, vol. 22, no. 3, pp. 301–312, 2012.

[78] B. Enjolras, K. Steen-Johnsen, and D. Wollebæk, "Social Media and Mobilization to Offline Demonstrations: Transcending Participatory Divides?" *New Media & Society*, vol. 15, no. 6, pp. 890–908, 2013.

[79] K. Varnali and V. Gorgulu, "A social influence perspective on expressive political participation in Twitter: The case of #OccupyGezi," *Information, Communication & Society*, vol. 18, no. 1, pp. 1–16, 2015.

[80] H. G. Ramírez and R. M. G. García, "Communication Networks of the 15M Movement on Twitter."

[81] R. Recuero, G. Zago, M. T. Bastos, and R. Araújo, "Hashtags Functions in the Protests Across Brazil," *SAGE Open*, vol. 5, no. 2, 2015.

[82] K. Thomas, F. Li, C. Grier, and V. Paxson, "Consequences of Connectivity: Characterizing Account Hijacking on Twitter," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 489–500.

[83] E. Bursztein, B. Benko, D. Margolis, T. Pietraszek, A. Archer, A. Aquino, A. Pitsillidis, and S. Savage, "Handcrafted Fraud and Extortion: Manual Account Hijacking in the Wild," in *Proceedings of the 2014 Conference on Internet Measurement Conference*. ACM, 2014, pp. 347–358.

[84] R. Shay, I. Ion, R. W. Reeder, and S. Consolvo, "My religious aunt asked why I was Trying to sell her viagra: Experiences with Account Hijacking," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2014, pp. 2657–2666.

[85] C. VanDam and P.-N. Tan, "Detecting Hashtag Hijacking from Twitter," in *Proceedings of the 8th ACM Conference on Web Science*. ACM, 2016, pp. 370–371.

[86] N. Jain, P. Agarwal, and J. Pruthi, "HashJacker - Detection and Analysis of Hashtag Hijacking on Twitter," *International Journal of Computer Applications*, vol. 114, no. 19, 2015.

[87] S. J. Jackson and B. Foucault Welles, "Hijacking #myNYPD: Social Media Dissent and Networked Counterpublics," *Journal of Communication*, vol. 65, no. 6, pp. 932–952, 2015.

[88] J. Sanderson, K. Barnes, C. Williamson, and E. T. Kian, "'How could anyone have predicted that #AskJameis would go horribly wrong?' Public Relations, Social Media, and Hashtag Hijacking," *Public Relations Review*, vol. 42, no. 1, pp. 31–37, 2016.

[89] A. T. Hadgu, K. Garimella, and I. Weber, "Political Hashtag Hijacking in the US," in *Proceedings of the 22nd International Conference on World Wide Web*. ACM, 2013, pp. 55–56.

[90] J. Masthoff, "Group Recommender Systems: Aggregation, Satisfaction and Group Attributes," in *Recommender Systems Handbook*, 2015, pp. 743–776.

[91] A. Hotho, A. Nürnberger, and G. Paass, "A Brief Survey of Text Mining," *LDV Forum*, vol. 20, no. 1, pp. 19–62, 2005.

[92] M. Porter. (2001) Snowball: A Language for Stemming Algorithms. Accessed Sept. 26, 2016.

[93] S. Pachano, "Ecuador: Cuando la Inestabilidad se vuelve Estable," *Íconos-Revista de Ciencias Sociales*, no. 23, pp. 39–46, 2013.

[94] C. De la Torre, "Protesta y Democracia en Ecuador: la caída de Lucio Gutiérrez," *Luchas contrahegemónicas y cambios políticos recientes de América Latina, Buenos Aires: CLACSO*, 2008.

[95] P. O. Peralta, "¿Por qué protestan en Ecuador?: Rafael Correa y el Fracasado Aumento del Impuesto a las Herencias," *Nueva sociedad*, no. 258, pp. 121–130, 2015.

[96] M. L. Maya, N. I. Carrera, P. Calveiro, and C. L. de Ciencias Sociales, *Luchas Contrahegemónicas y Cambios Políticos Recientes de América Latina*. Consejo Latinoamericano de Ciencias Sociales-CLACSO, 2008.

[97] J. Del Salto C., "La Comunicación Política 2.0 en la Campaña Presidencial de 2013 en Ecuador. Un Análisis del Uso de la Red Social Twitter," 2014.

[98] R. P. Vanegas Molina, "La Influencia de Rafael Correa en Twitter con Relación a las Convocatorias a Movilizaciones del 12, 13 y 14 de Agosto del 2015," 2016.

[99] M. B. Valdez Apolo, "Twitter como Instrumento de Comunicación y Gestión de Gobierno: El Caso del Presidente Rafael Correa," 2016.

[100] L. Terán and A. Kaskina, "Enhancing Voting Advice Applications with Dynamic Profiles," in *Proceedings of the 9th International Conference on Theory and Practice of Electronic Governance*. ACM, 2016, pp. 254–257.

[101] L. Terán, A. Balda, F. Mendez, I. Puyosa, I. Rivera, G. Baquerizo, D. Pastor, A. Illingworth, C. Vaca, J. Mendieta, and L. Recalde. Participa Ingeligente. Plataforma de Discusión y Participación Ciudadana. Accessed Jan. 26, 2017.

[102] A.-M. Popescu and M. Pennacchiotti, "Detecting Controversial Events from Twitter," in *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, ser. CIKM '10. New York, NY, USA: ACM, 2010, pp. 1873–1876.

[103] L. Recalde, D. F. Nettleton, R. Baeza-Yates, and L. Boratto, "Detection of trending topic communities: Bridging content creators and distributors," in *Proceedings of the 28th ACM Conference on Hypertext and Social Media*, ser. HT '17, 2017, pp. 205–213.

[104] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative Filtering Recommender Systems," in *The Adaptive Web*, ser. Lecture Notes in Computer Science, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds. Springer Berlin Heidelberg, 2007, vol. 4321, pp. 291–324.

[105] P. Lops, M. de Gemmis, and G. Semeraro, "Content-based Recommender Systems: State of the Art and Trends," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer US, 2011, pp. 73–105.

[106] I. Guy, N. Zwerdling, D. Carmel, I. Ronen, E. Uziel, S. Yogev, and S. Ofek-Koifman, "Personalized Recommendation of Social Software Items Based on Social Relations," in *Proceedings of the Third ACM Conference on Recommender Systems*, ser. RecSys '09. New York, NY, USA: ACM, 2009, pp. 53–60.

[107] R. R. Sinha and K. Swearingen, "Comparing Recommendations Made by Online Systems and Friends," in *DELOS Workshop: Personalisation and Recommender Systems in Digital Libraries*, 2001.

[108] H. Ma, D. Zhou, C. Liu, M. Lyu, and I. King, "Recommender Systems with Social Regularization," in *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, ser. WSDM '11, New York, USA, 2011, pp. 287–296.

[109] S. M. Kywe, T. A. Hoang, E. P. Lim, and F. Zhu, "On Recommending Hashtags in Twitter Networks," in *Proceedings of the 4th International Conference on Social Informatics*, ser. SocInfo'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 337–350.

[110] U. M. Dholakia, R. P. Bagozzi, and L. K. Pearo, "A Social Influence Model of Consumer Participation in Network- and Small-Group-Based Virtual Communities," *International Journal of Research in Marketing*, vol. 21, no. 3, pp. 241 – 263, 2004.

[111] M. Cha, H. Haddadi, F. Benevenuto, and K. Gummadi, "Measuring User Influence in Twitter: The Million Follower Fallacy," in *4th International AAAI Conference on Weblogs and Social Media (ICWSM)*, 2010.

[112] Y. Chen, L. Cheng, and C. Chuang, "A Group Recommendation System with Consideration of Interactions Among Group Members," *Expert Systems with Applications*, vol. 34, no. 3, pp. 2082 – 2090, 2008.

[113] L. Quijano-Sanchez, J. Recio-Garcia, B. Diaz-Agudo, and G. Jimenez-Diaz, "Social Factors in Group Recommender Systems," *ACM Transactions on Intelligent Systems and Technology*, vol. 4, no. 1, pp. 8:1–8:30, Feb. 2013.

[114] M. Ye, X. Liu, and W. Lee, "Exploring Social Influence for Recommendation: A Generative Model Approach," in *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '12. New York, NY, USA: ACM, 2012, pp. 671–680.

[115] D. J. Watts and P. S. Dodds, "Influentials, Networks, and Public Opinion Formation," *Journal of Consumer Research*, vol. 34, no. 4, pp. 441–458, 2007.

[116] E. Katz and P. F. Lazarsfeld, *Personal Influence, the Part Played by People in the Flow of Mass Communications*. Chicago, USA: The Free Press, 1955.

[117] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast Unfolding of Communities in Large Networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[118] N. M. Arqué and D. F. Nettleton, "Analysis of Online Social Networks Represented As Graphs - Extraction of an Approximation

of Community Structure Using Sampling," in *Proceedings of the 9th International Conference on Modeling Decisions for Artificial Intelligence*, ser. MDAI'12. Berlin, Heidelberg: Springer-Verlag, 2012, pp. 149–160.

[119] A. Zubiaga, D. Spina, R. Martínez-Unanue, and V. Fresno, "Real-Time Classification of Twitter Trends," *JASIST*, vol. 66, no. 3, pp. 462–473, 2015.

[120] S. Zhang and Q. Lv, "Event Organization 101: Understanding Latent Factors of Event Popularity," in *Proceedings of the Eleventh International AAAI Conference on Web and Social Media*, ser. ICWSM '17, 2017, pp. 716–719.

[121] S. Pramanik, M. Gundapuneni, S. Pathak, and B. Mitra, "Predicting Group Success in Meetup," 2016.

[122] X. Li, X. Cheng, S. Su, S. Li, and J. Yang, "A Hybrid Collaborative Filtering Model for Social Influence Prediction in Event-Based Social Networks," *Neurocomputing*, vol. 230, pp. 197 – 209, 2017.

[123] R. Du, Z. Yu, T. Mei, Z. Wang, Z. Wang, and B. Guo, "Predicting Activity Attendance in Event-Based Social Networks: Content, Context and Social Influence," in *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, ser. UbiComp '14. New York, NY, USA: ACM, 2014, pp. 425–434.

[124] W. Zhang, J. Wang, and W. Feng, "Combining Latent Factor Model with Location Features for Event-Based Group Recommendation," in *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '13. New York, NY, USA: ACM, 2013, pp. 910–918.

[125] C. Li, M. Bendersky, V. Garg, and S. Ravi, "Related Event Discovery," in *Proceedings of the Tenth ACM International Conference on*

Web Search and Data Mining, ser. WSDM '17. New York, NY, USA: ACM, 2017, pp. 355–364.

[126] T. A. N. Pham, X. Li, G. Cong, and Z. Zhang, "A General Graph-Based Model for Recommendation in Event-Based Social Networks," in *2015 IEEE 31st International Conference on Data Engineering*, April 2015, pp. 567–578.

[127] M. E. J. Newman, "Mixing Patterns in Networks," *Phys. Rev. E*, vol. 67, no. 2, p. 026126, Feb. 2003.

[128] S. Wasserman and K. Faust, *Social Network Analysis: Methods and Applications*. Cambridge university press, 1994, vol. 8.

[129] Z. Ertem, A. Veremyev, and S. Butenko, "Detecting Large Cohesive Subgroups with High Clustering Coefficients in Social Networks," *Social Networks*, vol. 46, pp. 1 – 10, 2016.

[130] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee, "Measurement and Analysis of Online Social Networks," in *Proceedings of the 7th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '07. New York, NY, USA: ACM, 2007, pp. 29–42.

[131] P. Brusilovsky and E. Millán, *User Models for Adaptive Hypermedia and Adaptive Educational Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 3–53.