Fake News Detection on Twitter Using Propagation Structures

Document status and date:
Published: 01/10/2020

DOI:
10.1007/978-3-030-61841-4_10

Document Version:
Publisher's PDF, also known as Version of record

Document license:
Taverne

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

Download date: 27 Apr. 2024

Fake News Detection on Twitter Using Propagation Structures

Marion Meyers(✉), Gerhard Weiss, and Gerasimos Spanakis

Department of Data Science and Knowledge Engineering, Maastricht University,
Maastricht, The Netherlands
marion.meyers@hotmail.com,
{gerhard.weiss,jerry.spanakis}@maastrichtuniversity.nl

Abstract. The growth of social media has revolutionized the way people access information. Although platforms like Facebook and Twitter allow for a quicker, wider and less restricted access to information, they also consist of a breeding ground for the dissemination of fake news. Most of the existing literature on fake news detection on social media proposes user-based or content-based approaches. However, recent research revealed that real and fake news also propagate significantly differently on Twitter. Nonetheless, only a few articles so far have explored the use of propagation features in their detection. Additionally, most of them have based their analysis on a narrow tweet retrieval methodology that only considers tweets to be propagating a news piece if they explicitly contain an URL link to an online news article. By basing our analysis on a broader tweet retrieval methodology that also allows tweets without an URL link to be considered as propagating a news piece, we contribute to fill this research gap and further confirm the potential of using propagation features to detect fake news on Twitter. We firstly show that real news are significantly bigger in size, are spread by users with more followers and less followings, and are actively spread on Twitter for a longer period of time than fake news. Secondly, we achieve an 87% accuracy using a Random Forest Classifier solely trained on propagation features. Lastly, we design a Geometric Deep Learning approach to the problem by building a graph neural network that directly learns on the propagation graphs and achieve an accuracy of 73.3%.

Keywords: Fake news · Twitter · Propagation

1 Introduction

The way people access information and news has radically shifted since the rise of social networks. From being platforms centered around creating and maintaining better social connections, applications such as Facebook and Twitter have become news providers for many of their users [3]. Twitter, with its 326 million monthly active users, has become more than just a social platform but has reinvented how citizens interact with each other and access information about the

world [11, 19]. As those platforms constitute a place where any opinion can be expressed and shared, they are also highly exposed to the dissemination of fake information. While traditional media sources such as newspapers and the television have a one-to-many structure, information on social media is shared on a many-to-many fashion hence making the monitoring of the information being diffused a much more complicated task.

The term *fake news* has been the subject of much controversy in the past years. Many definitions exist but none is universally accepted. It often encompasses notions such as manipulation, disinformation (information purposefully misleading), misinformation (information that is verifiably fake) and rumors [14]. In order to remain consistent throughout this article, the terms *fake news*, *fake information* and *fake fact* will be used interchangeably and their definitions will be restricted to claims that are verifiably false. Similarly, *real news*, *real information* and *real fact* will refer to claims that are verifiably true.

Fake news are referred to by many institutions and governments as one of the most dangerous threats to our current society [12], for example because of their influence on elections' results [6, 9, 10, 15, 16, 23]. As the power and dangers of fake news are increasingly acknowledged, many groups are taking actions against their diffusion, but a systematic way to detect them on social media is still lacking. Most approaches to fake news detection make use of user and content-based features. However, a recent study showed that fake and real news have significantly different propagation patterns [28]. This suggests that propagation features could be successfully used as a basis for classification. Additionally, compared to content-based features, propagation characteristics present the key advantage of being language independent. However, only a few studies so far have leveraged these features for the fake news detection task. Additionally, they have only done so on URL-restricted data sets, defined throughout this research as data sets created by a tweet retrieval methodology where a tweet is only considered to be propagating a news piece if it explicitly contains an URL link to an online news article. In contrast, we define a non-URL-restricted data set as one created by a tweet retrieval methodology that also allows tweets without an URL link to be considered as propagating a news piece.

Building on the apparent potential of propagation features to detect fake news on Twitter, and considering the narrow definition of news used in most of the research so far, this paper contributes to filling this research gap by answering the following research question: given a news graph G, defined here as a set of tweets and retweets that have been associated to a specific news item using a non URL-restricted retrieval methodology, how significant are propagation features at classifying G as a real or a fake piece of information?

This paper answers this question in 2 ways. On one hand, it does so by further investigating the significant differences in the propagation of real and fake information on a non URL-restricted Twitter data set. On the other, it evaluates the performance of 2 different types of classifiers that solely leverage propagation information: a Random Forest Classifier trained on manually extracted features

from the propagation graphs, and a Geometric Deep Learning approach directly applied on the full graphs representation. Our code is available via GitHub[1]

2   Related Work

Approaches to fake news detection typically make use of 3 types of information: user-based, news-based and propagation-based [26].

First, user-based approaches have shown promising classification results. Indeed, features extracted from user profiles such as their amount of followers and followings, their time since creation as well as their activity rate have shown to differ between real and fake information [2,22]. Additionally, user-based approaches to fake news detection have been further supported by the evidence that fake accounts play a great role in the dissemination of fake information on social media [6,20,23,24]. Hence, detecting fake accounts on social media is a valuable proxy for attempting to detect fake news [4].

Second, some approaches discriminate real and fake information on social media based on the content of the message being spread. This entails the topic being discussed in the post but also the type of words used, the sentiment portrayed and the 'non-linguistic' information such as the number of question marks or exclamation points employed. [2] for example showed that tweets displaying a stronger sentiment, containing many question marks or smiling emoticons were more likely to be related to non-credible news.

Third, propagation-based approaches classify real and fake information based on their respective diffusion patterns on social media. They are built on a theoretical framework of news diffusion on social media to which a considerable amount of research has been dedicated [21,27,30,31]. Propagation models generally represent tweets (or users) as nodes of a graph and social connections (follower, following) or influence paths (retweet, mention, comments, etc.) as edges. Throughout this article, those graphs will interchangeably be referred to as propagation graphs, propagation models, propagation structures or propagation networks. While user-based and content-based approaches have been the main focus in the existing literature, considerably less research has been dedicated to applying propagation features to the fake news detection task. However, some articles have successfully proved that fake and real news present significantly different propagation patterns on Twitter. [28] discovered that real news take about 6 times as long as fake news to reach 1500 users, consistently reach less users in total and were less retweeted. Additionally, [13] proved that fake news have a more fluctuated temporal diffusion. Then, a few attempts to make use of propagation features to detect fake news on Twitter have been developed. [2] combined different types of features (message-based, user-based, topic-based and propagation-based) and demonstrated that network features such as the number of tweets in the graph and the average node degree played a key role in their classifier's performance. Furthermore, [13] showed that the temporal features extracted from the propagation graphs allowed their classifier to achieve

---

[1] https://github.com/MarionMeyers/fake_news_detection_propagation.

better results than the baseline performance. Together, those articles suggest that propagation structures seem like promising features for classifying real and fake information on Twitter.

However, a more novel approach to graph classification that aims to optimize the use of propagation features has recently been applied to the problem. [5] makes use of the recent advances in Geometric Deep Learning to classify news directly on their Twitter propagation graphs and achieves state-of-the-art classification results (92.7% AUC_ROC). The field of 'Geometric Deep Learning' refers to methods that adapt deep learning approaches to higher dimensional data such as graphs and manifolds. Indeed, most machine learning approaches only work on Euclidian data, ie. 2-dimensional lists of features. When applied on graphs, this means reducing and discarding parts of the information through the manual choice of the 2D features to extract. Geometric Deep Learning approaches counter this limitation by designing neural networks able to learn directly from the 3D representation of the input: Graph Neural Networks. This entails the creation of layers able to cope with a varying input size since the training graphs have a different number of nodes and edges: Graph Convolutional Layers [29]. The success of this approach once again supports the relevance of using propagation features to classify real and fake news [17].

Lastly, both [28] and [5] gather news on Twitter by collecting URL links relating to a news article from fact-checking websites such as Snopes.com or Politifact.com[2,3]. Those websites collect news and score them on a veracity scale based on extensive investigation by independent journalists. Next, they either gather all tweets containing these URL links together with their corresponding retweets [5], or gather all reply tweets containing those URL links together with the original tweet and its associated retweets [28]. Both approaches lead to the creation of a data set where each array of tweets relating to a certain news item is labelled real or fake depending on the veracity of the article they are sharing. As previously defined, their approaches both present a URL-restricted tweet retrieval methodology.

3 Dataset

3.1 Dataset Collection

In our research we make use of the FakeNewsNet data set created in response to a clear lack of existing fake news data sets [25]. Their approach to data collection is to gather news articles from fact-checking organizations (Politifact and Gossipcop) together with their truth label assigned by independent journalists. From those labelled news articles, the headline is extracted and separated into a set of keywords. Then, those keywords are concatenated into a query for the Twitter API. For each news article, labelled real or fake, different kinds of information are then accessed:

---

[2] www.politifact.com.
[3] www.snopes.com.

– news content: the body of the article, images, publish date
– tweets: the list of tweets containing the article headline keywords
– retweets: the list of retweets of all tweets previously retrieved
– user information: the profile information (user id, creation date, 200 most recent published tweets, list of followers and friends) of all users that have posted a tweet or retweet related to the news article.

Not only does this data set provide us with the necessary information to create the propagation graphs detailed in the following section, but it also uses a non URL-restricted tweet retrieval methodology. Indeed, instead of collecting tweets that explicitly contain the URL link to the news piece, it gathers all tweets that contain the keywords associated with the article's headline.

The data set downloaded contains 347 fake news graphs and 310 real ones for a total of 518,684 tweets and 686,245 retweets.

Due to retrieval rate limitations imposed by Twitter, some parts of the data set require a very long time to be collected and were hence not included in this research. This includes both followers and followings information. Additionally, this limitation also led us to restrict the data set only to the Politifact website.

3.2   Propagation Graphs Creation

The propagation graphs, derived from the set of tweets and retweets correspond-ing to a labelled piece of information, are defined as follows:

– Let V be the set of nodes of the graph. A node can be of two types:
  1. A tweet node: the node stores the tweet and its associated user. A tweet belongs to a news graph if it contains the keywords extracted from the headline of the news article.
  2. A retweet node: the node stores the retweet and its associated user. All retweets of a tweet node are present in the graph.
– Let E be the set of edges of the graph. Edges are drawn between a tweet and its retweets. Edges contain a time weight that corresponds to the time difference between the tweet and retweet publish times.

Then G = (V,E) is the news graph. G is then a composition of non-connected sub-graphs where each sub-graph comprises a tweet and its associated retweets. It is important to note that Twitter is designed in such a way that a retweet of a retweet will point back to the original tweet. Hence, the depth of the graph is never more than 1.
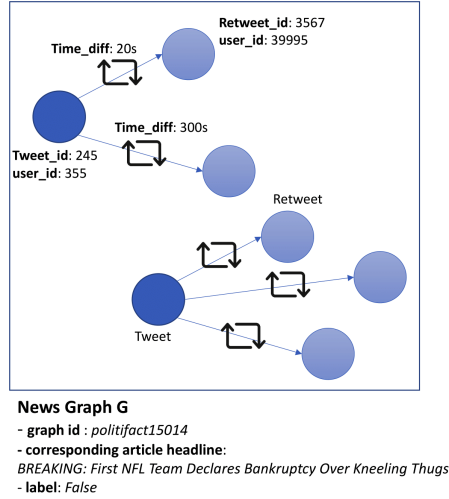
Fig. 1. Example of a propagation graph

4   Our Approach

Our research consists of two main steps:

1. Manually extract features from the propagation graphs in order to further investigate the possible significant differences between how real and fake information propagate on Twitter.
2. Build 2 classifiers trained on the propagation graphs (1) a classifier trained on the manually extracted features (2) a Geometric Deep Learning approach trained on the propagation graphs themselves.

4.1   Manual Extraction of Propagation Features

Table 1 presents all features extracted from the propagation graphs. Once extracted from all graphs, we perform a t-test statistical analysis on the means of the features in the real news and fake news graphs with a 0.05 significant level. Additionally, we perform an outlier analysis for several features in order to gain a better understanding of our data. Lastly, we look more in depth at the propagation of the tweets and retweets over time and analyze the temporal characteristics of their spread.

Table 1. Features Extracted From Each News Graph.

Scope	Feature	Description
User/Social Context Features	Avg number of followers	For each user that has either posted a tweet or a retweet in the graph, his amount of followers is retrieved. Those counts are then averaged over all users involved in the news graph
	Avg number of following	For each user that has either posted a tweet or a retweet in the graph, his amount of following (friends) is retrieved. Those counts are then averaged over all users involved in the news graph
Network Features	Retweet Percentage	This is measured through the following equation: $\frac{number\ of\ retweets}{number\ of\ tweets\ +\ number\ of\ retweets}$
	Average Time Diff	This measures the average time between a tweet and a corresponding retweet. Since each edge of the graph has a time weight on it, it is computed by making the average of all the edge weights of the graph
	Number of tweets	
	Number of retweets	
	Time_first_last or News lifetime	This measure is obtained by computing the time difference between the first and last recorded publish dates of tweets (or retweets) in the graph
	Average favorite count	For each node, its number of favourites is retrieved. Those counts are then averaged over all nodes in the graph
	AvgRetCount	For each tweet, its number of retweets is retrieved. Those counts are then averaged over all tweets in the graph
	UsersTouched 10 h	Starting from the first post recorded in the graph, all posts that happened in the first 10 h of the diffusion are retrieved. From those posts, the amount of unique users involved in the spread is then calculated
	PercPosts1hour	This feature is calculated by the following equation: $\frac{number\ of\ tweets\ and\ retweets\ in\ the\ first\ hour}{total\ number\ of\ tweets\ and\ retweets\ in\ the\ graph}$

4.2   Classification Approaches

Approach to the Classification on Manually Extracted Features. Our approach to the creation and the analysis of a classifier trained on manually extracted features from the graphs can be separated into 2 steps (1) Compare and select the best type of classifier for the problem (2) Analyze the importance of the different features in the classification.

Compare and Select the Best Type of Classifier

Different classifiers were trained using a 10-fold cross validation method. Namely, the algorithms tried are: Random Forest, Decision Tree, Linear Discriminant Analysis, Bayes Neural Network, Logistic Regression, K-Nearest Neighbors, Quadratic Discriminant Analysis and Support Vector Machine. As the data set

is slightly unbalanced, it is important to evaluate if this significantly impacts the classification performance. Hence, the performance of all classifiers is not only recorded on the full data set but also on 5 different under-sampled balanced versions of the data set. Their results are then compared and the algorithm yielding the highest accuracy will be chosen for further analysis.

Analyze the Importance of Different Features

To evaluate the importance given to each feature by the classifier, we record its performance over all possible subsets of features. Given that there are 11 features in total, the power set hence contains 2048 unique subsets (including the empty set). For each set size, we then record which feature (or combination of features) lead to the highest performance score. We do this for the best accuracy, best f1 score and best AUC_ROC. This approach not only allows us to understand what set size typically reaches the highest performance, but also which features play key roles in the classification.
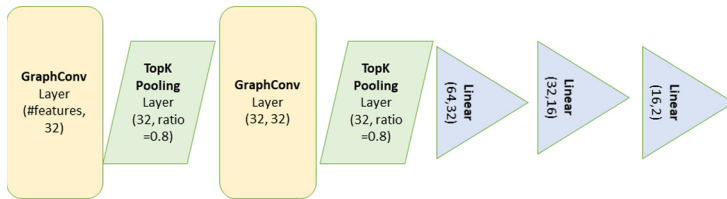

Fig. 2. GDL network architecture.

Geometric Deep Learning Approach. While most existing graph neural networks have been developed for the node classification task, the problem tackled here is that of graph classification. However, [1] and [8] have adapted current successes from the node to the graph classification task. They make use of the graph convolutional layer described in [18] as this layer was shown to be applicable to social networks and molecule graphs classification. In out research, this layer is combined with a specific pooling layer developed in [8], the topk pooling layer, that reduces the size of the graph at each iteration by choosing the top k best nodes and dropping the remaining ones. The choice of nodes to drop or keep is based on their inner features.

The neural network architecture used in this research is described more in details in Fig. 2.

The data fed into the network has to be specifically structured for the task. Indeed, not only are the graph connections themselves used for learning, but relevant features can also be encoded in both the nodes and the edges. Hence, nodes will be characterized by the following information:

– Number of followers of the user
– Number of friends (following) of the user
– Number of favorites of the tweet/retweet

- Number of retweets of the tweet (0 if the node is a retweet)
- Node type (either a tweet or a retweet)

Edges are characterized by the time difference between the tweet and its associated retweet. It is to be noted that all features inserted in the nodes are features that are also available to the classifier trained on manually extracted features in order for the future performance comparison to be applicable. In order to build our architecture, we have been using the recently released Pytorch Geometric library that had already implemented the different layers we are utilizing [7]. The network is using a 10-fold cross validation method on a balanced version of the data set (using under-sampling).

5   Experimental Results

5.1   Manually Extracted Propagation Features Analysis

After extracting the propagation features detailed in Table 1, their distribution for both real and fake news are analyzed. Table 2 presents the means and standard deviations of all features as well as the results of the student t-tests performed. When using a 0.05 significance level, the outcome of the analysis shows that 8 out of the 11 features are significantly different. Furthermore, the boxplot distributions of all 8 significant features are displayed in Appendix A.

By combining the t-test results with the significant features distribution presented in Appendix A, different conclusions can be drawn on the data set and the differences in propagation between real and fake information on Twitter.

Real News Are 'bigger' Than Fake News. Real news have an average of 1212 tweets and 1796 retweets while fake news have on average 411 tweets and 372 retweets. From the statistical analysis displayed in Table 2, it is observed that the means of both features are significantly different. By further analyzing

Table 2. Features Summary.

	mean real	mean fake	std real	std fake	t	pValue	signif
followerAvg	34607.0280	8835.2657	73084.3660	14107.1257	6.1079	0.0000	Y
followingAvg	3386.4674	4535.2654	3998.6284	3201.5401	−4.0336	0.0001	Y
retweetPerc	0.4132	0.3730	0.2262	0.2214	2.2969	0.0219	Y
avgTimeDiff (in seconds)	372966.1338	320420.5629	1956157.2011	1451788.9476	0.3872	0.6988	N
numTweets	1212.3710	411.6686	2824.1935	1600.1888	4.4005	0.0000	Y
numRetweets	1796.6161	372.6052	4927.2753	1969.8602	4.7600	0.0000	Y
avgFav	.1861	1.3384	5.9917	4.5917	2.0175	0.0441	Y
avgRetCount	3.1288	3.1925	8.6245	15.7255	−0.0653	0.9480	N
news lifetime (in seconds)	115662880.1871	27737159.8963	97964001.7070	45342932.5034	14.4778	0.0000	Y
usersTouched10hours	71.6710	57.7666	192.7123	150.3321	1.0225	0.3070	N
percPosts1hour	0.1528	0.0720	0.2521	0.1269	5.0940	0.0000	Y

the 4 highest outliers in the number of tweets (2 real and 2 fake), a limitation to the data collection protocol used in this research was discovered. Indeed, they all have an extremely large number of tweets because the list of keywords used to extract the relevant twitter information is very broad and leads to the retrieval of many posts that do not correspond to the original news. For example, a query that lead to the retrieval of 24,338 tweets is 'One in Four – Congressman Joe Pitts'. Initially referring to an article written by the Congressman Joe Pitts on addiction rate in Pennsylvia, the broad query led to the retrieval of many unrelated tweets such as "In Chinese universities, students sleep four to a dorm room. I would not have survived it. One was difficult enough...".

Real News Stay Longer 'in the Loop'. The news lifetime was shown to be significantly different for real and fake graphs (see Table 2). Real news stay on average 4.16 times longer on Twitter than fake ones (1338 vs 321 days). By looking at the boxplots in Appendix A, it is interesting to note that while the lifetime of fake news presents a certain amount of outliers, the real news lifetime is more spread but doesn't show any outlier. A deeper look at the fake news outliers proves once again that very broad queries lead to the retrieval of many more tweets than intended. For example, the query 'Sid Miller', initially referring to a fake image of the politician spread on Twitter in 2016, encompassed a tweet dating from 2011 that used the same keywords, thereby yielding an abnormally large lifetime for the fake news. We also note the possibility of recurrent fake news that lead to an abnormally long lifetime. This is the case for a fake news that emerged both in 2012 and 2017 involving Barack Obama's face being printed on one-dollar bills.

Two hypothesis can then be formulated to try to explain why real news show a longer lifetime on Twitter. First, real news could present queries that are more likely to be used at different points in time hence augmenting their probability of showing a larger news lifetime average. In comparison, fake news would show a more novel and rare set of keywords that are less likely to be re-used in other news items. Second, the lifetime of fake news could be shorter due to the fact that once they are proven to be misleading, their spread is more likely to be halted.

Users Spreading Real News Tend to Have More Followers but to Follow Less Accounts. On average, users involved in the propagation of real information have 34,607 followers while fake news propagators only have 4,535. The statistical results in Table 2 confirm that those means are significantly different. A quick look at the real news outliers in follower counts shows that they seem to be shared by trustworthy accounts such as the NY Times (43,254,008 followers) or the Huffington Post (11,477,200 followers). On the contrary, real news propagators follow on average less accounts than fake news propagators do. While accounts linked to spreading real news follow on average 3386.47 other accounts, fake news propagators follow on average 4535.27 accounts. Once again, this difference has been statistically proven to be significant.
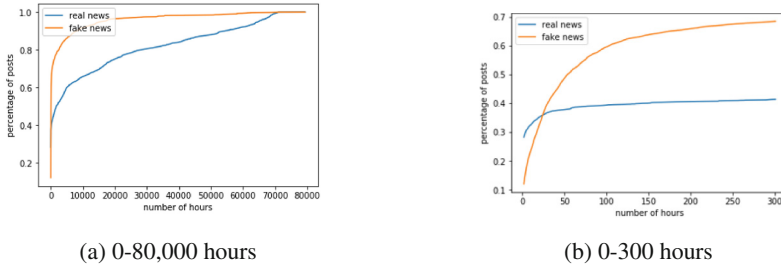
(a) 0-80,000 hours

(b) 0-300 hours

Fig. 3. Average Percentage of Posts over Time

Temporal Spread Analysis. Figure 3 presents an average of the percentage of tweets and retweets posted over time for fake and real news. Firstly, we see on the first graph of Fig. 3 that fake news reach 100% of their posts earlier than real news (30,000 vs 70,000 h). This corresponds to our previous finding that the lifetime of real news is bigger than that of fake news.

Secondly, the shapes of the two curves are very different. The fake news curve shows a strong increase in the its beginning before increasing in a more moderate manner and remaining relatively stable from about 15,000 h on. The real news curve also shows an steep increase at the beginning but quickly evolves into a more moderate increase over time, to only reach its 100% at about 70,000 h. In order to better visualize and compare the early increases of the two curves, the second graph of Fig. 3 presents the same curves on a shorter amount of time. We observe that although real news already reach 30% of their posts in the first hour of spread, fake news quickly overtake and reach 70% of their posts after 300 h. By then, the real news have only reached 40% of their posts.

This analysis allows us to visually represent our previous finding that fake news have a shorter lifetime than real news. Indeed, we see that while real news have a slower increase over time and thereby a larger lifetime, fake news reach the end of their spread faster, hence have a shorter lifetime. It is also important to note that our previous finding about the news size are likely to have impacted the results of this temporal analysis. Indeed, as real news are significantly bigger in size, they are more likely to take a longer time to be spread.

5.2   Classification Results

Classifier on Manually Extracted Features
Compare And Select The Best Type Of Classifier Appendix B presents the scores of all classifiers attempted. Firstly, we only observe a small difference in the accuracy of the classifiers when applied on the full data sets or on the balanced versions. Looking at the Random Forest Classifier, its accuracy on the balanced data sets oscillates between 83.5% and 86.5%, and obtains an accuracy of 85% on the full data set. We then conclude that the slightly unbalanced characteristic of the data set does not have a concrete influence on the classification performance.

The Random Forest Classifier ranked the highest in all scores and was hence selected as classification algorithm for the rest of the analysis.

	bestAcc
1 : {'time_first_last'}	0.675641
2 : {'time_first_last', 'followerAvg'}	0.814499
3 : {'time_first_last', 'followingAvg', 'followerAvg'}	0.829580
4 : {'avgFav', 'time_first_last', 'followingAvg', 'followerAvg'}	0.859953
5 : {'followingAvg', 'followerAvg', 'percTweets1hour', 'avgFav', 'time_first_last'}	0.859977
6 : {'followingAvg', 'followerAvg', 'percTweets1hour', 'avgFav', 'time_first_last', 'retweetPercentage'}	0.872075
7 : {'followingAvg', 'followerAvg', 'avgRetCount', 'avgFav', 'time_first_last', 'retweetPercentage', 'numTweets'}	0.866154
8 : {'followingAvg', 'followerAvg', 'avgFav', 'usersTouched10hours', 'time_first_last', 'numTweets', 'retweetPercentage', 'numRetweets'}	0.873730
9 : {'followingAvg', 'followerAvg', 'avgRetCount', 'percTweets1hour', 'avgFav', 'usersTouched10hours', 'time_first_last', 'numTweets', 'retweetPercentage'}	0.856853
10 : {'followingAvg', 'followerAvg', 'avgRetCount', 'avgFav', 'avgTimeDiff', 'percTweets1hour', 'usersTouched10hours', 'time_first_last', 'numTweets', 'numRetweets'}	0.852331
11 : {'followingAvg', 'followerAvg', 'avgRetCount', 'avgFav', 'avgTimeDiff', 'percTweets1hour', 'usersTouched10hours', 'time_first_last', 'numTweets', 'retweetPercentage', 'numRetweets'}	0.852331

Fig. 4. Set of features reaching the highest accuracy per subset size

Analyze The Importance Of Different Features Firstly, we observe in Fig. 4 that the random forest classifier reaches its highest accuracy on the full data set when using a set of 8 features. While it reaches 85% accuracy using the 11 features, the performance goes up to 87% when using the following set of features: *followingAvg, followerAvg, avgFav, usersTouched10hours, news lifetime, numTweets, retweetPercentage, numRetweets*.

Secondly, Table 3 presents a summary of the number of occurrences of each feature in all the best subsets presented in Fig. 4. We observe that the news lifetime is present in all of them, followed by the average number of followers present in 10 out of 11 subsets. We also see in Fig. 4 that these two features combined already accurately classify 81.44% of all graphs. This leads us to conclude that they are both of major importance in the classification. Additionally, both the following average and the average number of favourites seem to be important as they are present in respectively 8 and 9 of all best subsets.

Thirdly, we note without surprise that the 3 features that were proven to be non-significant (*avgTimeDiff, usersTouched10hours* and*avgRetCount*) don't contribute much to the classification performance.

Lastly, we observe that the number of tweets and retweets are only present in 4 of the best subsets. Although the features were both shown to be significant, the median of both features were very similar between the real and fake sets of graphs, which might explain why the random forest classifier did not give them a strong importance.

5.3 Geometric Deep Learning

Before training the algorithm, the pre-processing step of normalizing the features is performed. Then, the neural network is trained using a 10-fold cross validation method. A mini-batch size of 1 and a learning rate of 0.001 were found to be yielding the best results. When trained for 400 epochs, the neural network

Table 3. Number of occurrences of each feature in all best subsets

	Number of Occurrences
newsLifetime	11
followerAvg	10
followingAvg	9
avgFav	8
retweetPerc	5
numTweets	4
numRetweets	4
avgRetCount	4
usersTouched10hours	4
percPosts1hour	3
avgTimeDiff	1

achieved the results displayed in Fig. 5. On average over the 10 folds, the accuracy recorded on the last epoch is 73.29%, with a standard deviation of 0.0746 which proves the robustness of the model.

Our Geometric Deep Learning approach has only been tried on one neural network architecture, which leads us to conclude that a gdl-based detection of fake news seems like a promising approach given the satisfactory results presented above. However, a systematic comparison of gdl models is needed in order optimize the model for this specific task instead of utilizing a model proven to be successful in other classification tasks.

	mean	standard dev
accuracy	0.7329	0.0746
precision	0.6846	0.1102
recall	0.8755	0.1081
f1 score	0.7606	0.0821
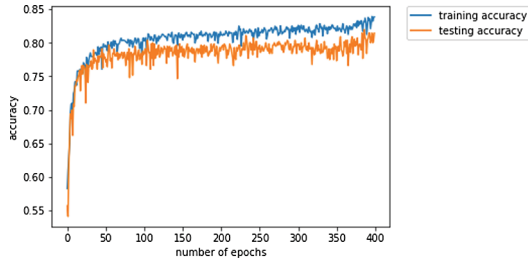
Fig. 5. Geometric Deep Learning approach scores over 400 epochs

6   Discussion

The experiments performed in this paper led us to gain insights on how fake and real news propagate on Twitter. It is then interesting to compare our findings with those achieved by previous research. Firstly, [28] has found that fake news

propagate wider, faster and deeper than real news. More specifically, they discovered that real news take about 6 times as long as fake news to reach 1500 users, consistently reach less users in total and were less retweeted. However, our conclusions somehow contradict their findings since we have observed that real news present more tweets and retweets. However, both the average retweet count and the users touched in the first 10 h feature are not significant in our results hence preventing us from fully arguing against their finding. It is however important to note that while our results have been discovered on an entire news graphs composed of non-connected sub-graphs, their conclusions are drawn from individual retweet cascades. This methodological contrast might contribute to the evident disaccord between our results. Secondly, both [2] and [28] support our finding that real news are spread by users with more followers than those spreading fake information. However, our results about the number of followings is opposite to theirs. While both their analysis show that real news propagators follow more people, our research shows that fake news propagators actually have more followings. Lastly, to the best of our knowledge, no previous work seems to make use of 'lifetime' as classification feature thereby preventing us from making any comparison.

The last section of the experiments entailed the application and evaluation of a Geometric Deep Learning approach to the problem, which achieved an accuracy of 73.3%. The only other application of Geometric Deep Learning to fake news detection had achieved an AUC_ROC of 92.7% on their URL-wise classification but their network had the advantage of containing social connections and influence paths [5].

Before summarizing the final conclusions of our research paper, it is necessary to underline its major limitations. First of all, although using a non URL-restricted news definition distinguishes our research from most of the existing literature on fake news classification, it brings up the issue of using a definition that is very broad. As explained in Sect. 5, using the keywords from the articles headlines leads in some cases to the retrieval of many tweets that are unrelated to the original news piece. This also causes some graphs to cover periods of time that seem unrealistic. This limitation is hard to circumvent when dealing with fake news detection research. One the one hand, our choice of data is restricted by the very limited availability of Twitter labelled news data sets. On the other hand, none of these data sets agree on a precise methodology to retrieve tweets that correspond to a news piece. Although the majority has been following the URL-restricted approach defined earlier, this methodology also has major limitations. Second of all, all news analysed come from a single source of information, Politifact, that mainly includes American political news. This hence prevents us from generalizing our findings to other news topics.

7   Conclusion

This paper demonstrated the potential of using propagation features to discriminate real from fake news on Twitter by analyzing a non URL-restricted data

set. More specifically, it firstly discovers the following significant differences in the propagation of the real and fake news: real news graphs are bigger in size, are spread by users with more followers and less followings, and stay longer on Twitter than fake news. Secondly, it achieves a 87% detection accuracy using a Random Forest Classifier solely trained on propagation features, hence further confirming the latter assumption. Lastly, by developing a graph neural network trained directly on the 3D representation of the propagation graphs, it achieves an accuracy of 73.3%. Overall, the significant differences discovered as well as the good performances achieved by the 2 algorithms trained on propagation information lead us to conclude that propagation features are a relevant and important asset to the fake news detection task on Twitter.

Further research should firstly be dedicated to the evaluation of our classification approaches on the early detection of fake news instead of at the end of their diffusion. Secondly, further efforts should go into refining our data set in order to counter the negative impact of our broad definition of news on the reliability of our results. In order to do that, a time limit on the retrieval of the tweets could be set, or the analysis could be performed on the tweet cascades (the set of one tweet and its corresponding retweets) instead of on the entire news graph. Thirdly, it would be interesting to apply our approach to other news topics than political news in order to evaluate if the same conclusions on the propagation patterns can be drawn. Lastly, the GDL experiments were only performed on one type of convolutional and pooling layers, while many more have been shown to be successful in various applications. Further research should hence be dedicated to trying different versions of this neural network and hopefully improve the classification performance by finding the optimal combination of convolutional and pooling layers.

Appendix

Appendix A: Significant Features Distribution
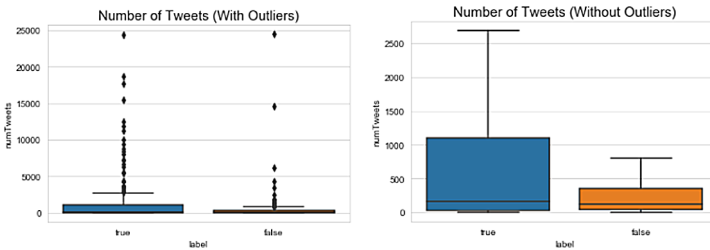
Fig. 6. Number of Tweets Distribution.
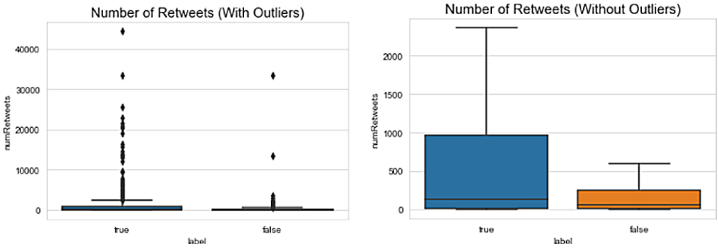
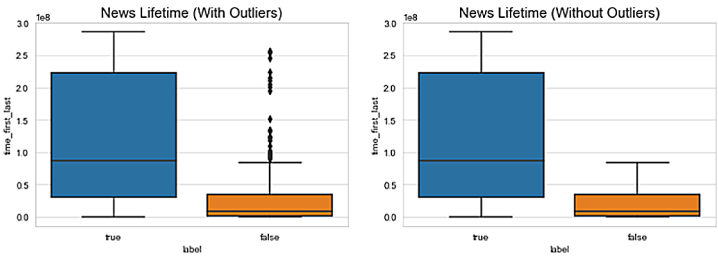**Fig. 7.** Number of Retweets Distribution.

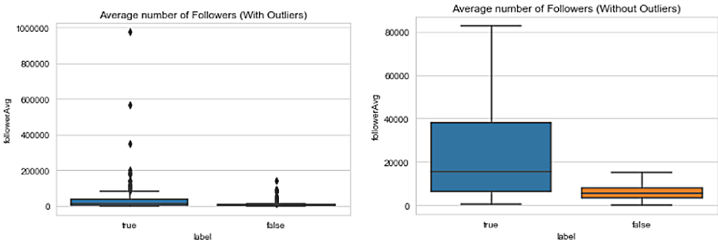**Fig. 8.** News Lifetime (time_first_last) Distribution.

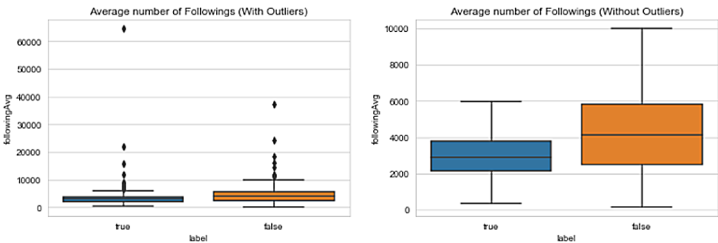**Fig. 9.** Average Number of Followers Distribution.

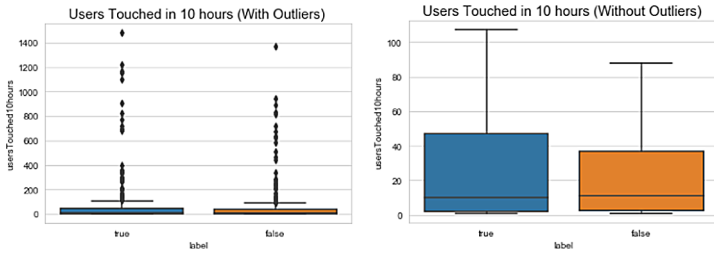**Fig. 10.** Average Number of Followings Distribution.

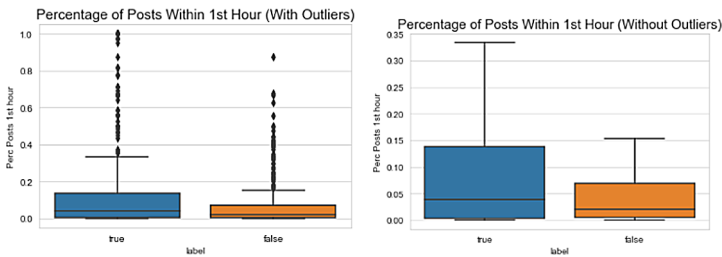**Fig. 11.** Number of Users Touched Within the First 10 h Distribution.

Fig. 12. Percentage of Posts In The First Hour Distribution.

Appendix B: Classifiers Scores Comparison

	Model	Fitting time	Scoring time	Accuracy	Precision	Recall	F1_score	AUC_ROC
5	Random Forest	0.019847	0.009275	0.850000	0.855226	0.850000	0.849355	0.917950
3	Linear Discriminant Analysis	0.002991	0.004322	0.799767	0.813922	0.799767	0.797219	0.891538
1	Decision Tree	0.005983	0.005187	0.782258	0.785503	0.782258	0.781449	0.782258
4	Quadratic Discriminant Analysis	0.001796	0.004787	0.762903	0.786729	0.762903	0.757126	0.852549
7	Bayes	0.001394	0.004387	0.751613	0.791883	0.751613	0.741174	0.856400
6	K-Nearest Neighbors	0.001183	0.014667	0.738710	0.742815	0.738710	0.737370	0.788450
0	Logistic Regression	0.004387	0.004888	0.635484	0.733380	0.635484	0.574515	0.866389
2	Support Vector Machine	0.177230	0.022935	0.503226	0.350820	0.503226	0.340308	0.509677

Fig. 13. Classifier Scores: under-sampled balanced data set 1

	Model	Fitting time	Scoring time	Accuracy	Precision	Recall	F1_score	AUC_ROC
5	Random Forest	0.018745	0.008777	0.864516	0.870510	0.864516	0.863998	0.909209
3	Linear Discriminant Analysis	0.002882	0.003988	0.808077	0.823229	0.808077	0.805685	0.887988
1	Decision Tree	0.005080	0.003392	0.790323	0.795953	0.790323	0.789402	0.790323
4	Quadratic Discriminant Analysis	0.001297	0.003989	0.764516	0.783938	0.764516	0.759388	0.853174
7	Bayes	0.001295	0.003889	0.754839	0.793074	0.754839	0.745782	0.853486
6	K-Nearest Neighbors	0.001292	0.015763	0.750000	0.751943	0.750000	0.749607	0.798075
0	Logistic Regression	0.004004	0.003981	0.685484	0.751533	0.685484	0.651006	0.862019
2	Support Vector Machine	0.150106	0.018250	0.506452	0.451639	0.506452	0.347283	0.512903

Fig. 14. Classifier Scores: under-sampled balanced data set 2

	Model	Fitting time	Scoring time	Accuracy	Precision	Recall	F1_score	AUC_ROC
5	Random Forest	0.018440	0.008182	0.851613	0.854290	0.851613	0.851336	0.909105
3	Linear Discriminant Analysis	0.002771	0.004321	0.806489	0.820166	0.806489	0.804613	0.890859
1	Decision Tree	0.005983	0.003491	0.775806	0.777676	0.775806	0.775449	0.775806
4	Quadratic Discriminant Analysis	0.001396	0.004987	0.762903	0.779588	0.762903	0.759104	0.845578
7	Bayes	0.001306	0.004177	0.751613	0.787460	0.751613	0.743435	0.847242
6	K-Nearest Neighbors	0.001200	0.016457	0.730645	0.732476	0.730645	0.730079	0.783351
0	Logistic Regression	0.003804	0.003679	0.722581	0.785927	0.722581	0.700066	0.869823
2	Support Vector Machine	0.184508	0.020855	0.508065	0.402091	0.508065	0.350243	0.509677

Fig. 15. Classifier Scores: under-sampled balanced data set 3

	Model	Fitting time	Scoring time	Accuracy	Precision	Recall	F1_score	AUC_ROC
5	Random Forest	0.019647	0.009072	0.835484	0.842741	0.835484	0.834513	0.904214
3	Linear Discriminant Analysis	0.002881	0.004211	0.800233	0.813652	0.800233	0.797959	0.886793
1	Decision Tree	0.004981	0.003592	0.793548	0.797135	0.793548	0.792968	0.793548
4	Quadratic Discriminant Analysis	0.001696	0.003989	0.774194	0.792638	0.774194	0.770666	0.845369
7	Bayes	0.001495	0.005094	0.758065	0.793220	0.758065	0.750942	0.853382
6	K-Nearest Neighbors	0.001379	0.016767	0.741935	0.744589	0.741935	0.740953	0.787929
0	Logistic Regression	0.003390	0.003092	0.674194	0.747914	0.674194	0.636437	0.856712
2	Support Vector Machine	0.157588	0.017446	0.506452	0.451639	0.506452	0.347283	0.512903

Fig. 16. Classifier Scores: under-sampled balanced data set 4

	Model	Fitting time	Scoring time	Accuracy	Precision	Recall	F1_score	AUC_ROC
5	Random Forest	0.016362	0.008080	0.854839	0.860123	0.854839	0.854257	0.917430
3	Linear Discriminant Analysis	0.002658	0.003324	0.805182	0.822330	0.805182	0.802539	0.890440
1	Decision Tree	0.005385	0.004289	0.796774	0.800022	0.796774	0.796213	0.796774
4	Quadratic Discriminant Analysis	0.001799	0.005084	0.761290	0.777300	0.761290	0.757422	0.857232
7	Bayes	0.001595	0.003989	0.756452	0.790125	0.756452	0.748277	0.853278
6	K-Nearest Neighbors	0.001502	0.015652	0.720968	0.724078	0.720968	0.719957	0.784183
0	Logistic Regression	0.003094	0.003595	0.645161	0.711975	0.645161	0.602609	0.859001
2	Support Vector Machine	0.157389	0.018448	0.503226	0.350820	0.503226	0.340308	0.509677

Fig. 17. Classifier Scores: under-sampled balanced data set 5

	Model	Fitting time	Scoring time	Accuracy	Precision	Recall	F1_score	AUC_ROC
5	Random Forest	0.021441	0.009176	0.850793	0.854662	0.852197	0.850523	0.921442
3	Linear Discriminant Analysis	0.002992	0.003657	0.809689	0.826785	0.803407	0.806346	0.884708
1	Decision Tree	0.006286	0.003791	0.785408	0.789165	0.786788	0.784979	0.786788
4	Quadratic Discriminant Analysis	0.001795	0.005387	0.777669	0.793786	0.771159	0.772636	0.851762
7	Bayes	0.001610	0.004280	0.765571	0.802432	0.755701	0.755153	0.851277
6	K-Nearest Neighbors	0.001399	0.020044	0.736503	0.739172	0.734965	0.735711	0.792261
0	Logistic Regression	0.003697	0.004182	0.730326	0.778263	0.737258	0.712641	0.867978
2	Support Vector Machine	0.197067	0.026137	0.534219	0.465700	0.506452	0.378318	0.512903

Fig. 18. Classifier Scores: full data set

Appendix C: Feature Importance Analysis

Fig. 19. Best Subsets Analysis Full Data Set

References

1. Cangea, C., Veličković, P., Jovanović, N., Kipf, T., Liò, P.: Towards sparse hierarchical graph classifiers. arXiv preprint arXiv:1811.01287 (2018)
2. Castillo, C., Mendoza, M., Poblete, B.: Information credibility on twitter. In: Proceedings of the 20th International Conference on World Wide Web, pp. 675–684. ACM (2011)
3. Center, P.R.: News use across social media platforms 2018 (2018). https://www.journalism.org/2018/09/10/news-use-across-social-media-platforms-2018/. Accessed 03 June 2019
4. Davis, C.A., Varol, O., Ferrara, E., Flammini, A., Menczer, F.: Botornot: a system to evaluate social bots. In: Proceedings of the 25th International Conference Companion on World Wide Web, pp. 273–274. International World Wide Web Conferences Steering Committee (2016)
5. Federico, M., Fabrizio, F., Davide, E., Damon, M.: Fake news detection on social media using geometric deep learning. arXiv preprint arXiv:1902.06673 (2019)
6. Ferrara, E.: Disinformation and social bot operations in the run up to the 2017 french presidential election (2017)
7. Fey, M., Lenssen, J.E.: Fast graph representation learning with PyTorch Geometric. In: ICLR Workshop on Representation Learning on Graphs and Manifolds (2019)
8. Gao, H., Ji, S.: Graph u-net (2019). https://openreview.net/forum?id=HJePRoAct7
9. Gorodnichenko, Y., Pham, T., Talavera, O.: Social media, sentiment and public opinions: Evidence from# brexit and# uselection. Technical report, National Bureau of Economic Research (2018)
10. Guardian, T.: Bolsonaro business backers accused of illegal whatsapp fake news campaign (2018). https://www.theguardian.com/world/2018/oct/18/brazil-jair-bolsonaro-whatsapp-fake-news-campaign. Accessed 03 Aug 2019
11. Iqbal, M.: Twitter revenue and usage statistics (2018). http://www.businessofapps.com/data/twitter-statistics/. Accessed 03 June 2019
12. Kalsnes, B.: Fake news, May 2019. https://oxfordre.com/communication/view/10.1093/acrefore/9780190228613.001.0001/acrefore-9780190228613-e-809
13. Kwon, S., Cha, M., Jung, K., Chen, W., Wang, Y.: Prominent features of rumor propagation in online social media. In: 2013 IEEE 13th International Conference on Data Mining, pp. 1103–1108. IEEE (2013)
14. Lazer, D.M., et al.: The science of fake news. Science **359**(6380), 1094–1096 (2018)
15. Leonhardt, D., Thompson, S.A.: Trump's lies (2017). https://www.nytimes.com/interactive/2017/06/23/opinion/trumps-lies.html, archived from the original on 23 June 2017
16. Marwick, A., Lewis, R.: Media Manipulation and Disinformation Online. Data & Society Research Institute, New York (2017)
17. Monti, F., Boscaini, D., Masci, J., Rodola, E., Svoboda, J., Bronstein, M.M.: Geometric deep learning on graphs and manifolds using mixture model CNNS. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5115–5124 (2017)
18. Morris, C., et al.: Weisfeiler and leman go neural: Higher-order graph neural networks. arXiv preprint arXiv:1810.02244 (2018)
19. Nielsen, R.K.: News media, search engines and social networking sites as varieties of online gatekeepers. In: Rethinking Journalism Again, pp. 93–108. Routledge (2016)

20. Review, M.T.: First evidence that social bots play a major role in spreading fake news (2017). https://www.technologyreview.com/s/608561/first-evidence-that-social-bots-play-a-major-role-in-spreading-fake-news/. Accessed 03 June 2019
21. Sadikov, E., Martinez, M.M.M.: Information propagation on twitter. CS322 project report (2009)
22. Shao, C., Ciampaglia, G.L., Flammini, A., Menczer, F.: Hoaxy: a platform for tracking online misinformation. In: Proceedings of the 25th International Conference Companion on World Wide Web, pp. 745–750. International World Wide Web Conferences Steering Committee (2016)
23. Shao, C., Ciampaglia, G.L., Varol, O., Flammini, A., Menczer, F.: The spread of fake news by social bots. arXiv preprint arXiv:1707.07592 pp. 96–104 (2017)
24. Shao, C., Ciampaglia, G.L., Varol, O., Yang, K.C., Flammini, A., Menczer, F.: The spread of low-credibility content by social bots. Nat. Commun. **9**(1), 4787 (2018)
25. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., Liu, H.: Fakenewsnet: A data repository with news content, social context and dynamic information for studying fake news on social media. arXiv preprint arXiv:1809.01286 (2018)
26. Shu, K., Sliva, A., Wang, S., Tang, J., Liu, H.: Fake news detection on social media: a data mining perspective. ACM SIGKDD Explorations Newsletter **19**(1), 22–36 (2017)
27. Tambuscio, M., Ruffo, G., Flammini, A., Menczer, F.: Fact-checking effect on viral hoaxes: a model of misinformation spread in social networks. In: Proceedings of the 24th International Conference on World Wide Web, pp. 977–982. ACM (2015)
28. Vosoughi, S., Roy, D., Aral, S.: The spread of true and false news online. Science **359**(6380), 1146–1151 (2018)
29. Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., Yu, P.S.: A comprehensive survey on graph neural networks. arXiv preprint arXiv:1901.00596 (2019)
30. Xiong, F., Liu, Y.: Opinion formation on social media: an empirical approach. Chaos: An Interdisciplinary J. Nonlinear Sci. **24**(1), 013130 (2014)
31. Xiong, F., Liu, Y., Zhang, Z.J., Zhu, J., Zhang, Y.: An information diffusion model based on retweeting mechanism for online social media. Phys. Lett. A **376**(30–31), 2103–2108 (2012)