J SYST SCI SYST ENG
Vol. 30, No. 1, February 2021, pp. 85–104
DOI: https://doi.org/10.1007/s11518-020-5470-4
ISSN: 1004-3756 (paper), 1861-9576 (online)
CN 11-2983/N

# "Towards Re-Inventing Psychohistory": Predicting the Popularity of Tomorrow's News from Yesterday's Twitter and News Feeds

**Jiachen Sun,[a, b] Peter Gloor[a]**

[a]MIT Center for Collective Intelligence, 245 First Street, 02142, Cambridge, MA, USA
pgloor@mit.edu (✉)
[b]School of Electronics and Information Technology, Sun Yat-Sen University, Guangzhou, 510275, China
sunjch6@mail2.sysu.edu.cn

**Abstract.** Rapid advances in machine learning combined with wide availability of online social media have created considerable research activity in predicting what might be the news of tomorrow based on an analysis of the past. In this work, we present a deep learning forecasting framework which is capable to predict tomorrow's news topics on Twitter and news feeds based on yesterday's content and topic-interaction features. The proposed framework starts by generating topics from words using word embeddings and K-means clustering. Then temporal topic-networks are constructed where two topics are linked if the same user has worked on both topics. Structural and dynamic metrics calculated from networks along with content features and past activity, are used as input of a long short-term memory (LSTM) model, which predicts the number of mentions of a specific topic on the subsequent day. Utilizing dependencies among topics, our experiments on two Twitter datasets and the HuffPost news dataset demonstrate that selecting a topic's historical local neighbors in the topic-network as extra features greatly improves the prediction accuracy and outperforms existing baselines.

**Keywords:** Topic's popularity, trend forecasting, social media

## 1. Introduction

*"Since emotions are few and reasons many, the behavior of a crowd can be more easily predicted than the behavior of one person can. And that, in turn, means that if laws are to be developed that enable the current of history to be predicted, then one must deal with large populations, the larger the better. That might itself be the First Law of Psychohistory, the key to the study of Humanics. Yet."*

- Isaac Asimov, Robots and Empire

Inspired by "psychohistory" invented by the fictitious mathematician Hari Seldon in the science fiction stories of Isaac Asimov, many researchers have tried to predict parts of the future using social media sources, although until now mostly limited to areas where solid time-series of the past are available, such as book sales (Gruhl et al. 2005), stock prices (Choudhury et al. 2008, Zhang et al. 2011), election outcomes (Ebrahimi et al. 2017), and Oscar awards (Krauss et al. 2010). Today, thanks to the advent of artificial intelligence and the boom in social networking, we finally have reached the point where we can tackle the general concept of psychohistory foreseen by Asimov over 60 years ago. In support of this trend, the notion of crowd behavior prediction on social media, such as what kind of topic people are going to mention, and how many shares or page-views an online article or a video will receive, has been widely studied. Obviously, the capability to forecast online content popularity is of great significance to both theory and practice. On the one hand, predicting tomorrow's

news presents important opportunities to capture the evolution of a global audience's attention around the massive and fast-growing contents and topics available on social media and thus helps researchers better understand how people utilize the Internet in terms of collective human behavior. On the other hand, news organizations, politicians, and marketing executives of companies would all like to know what matters to their audiences tomorrow, to target their messages in the best possible way to suit their needs. For example, knowing what kind of news topics will bring the most traffic on social media in the near future enables media participants to develop efficient marketing strategies to either maximize the impact of their online articles or downgrade unpopular content in advance.

Given the high level of interest in online content popularity prediction, a number of researches (Szabo and Huberman 2010, Ahmed et al. 2013, Tatar et al. 2011, Weng et al. 2014, Abbar et al. 2018) have been proposed by measuring a topic or article's popularity as user participative activity such as user comments, votes, or the size of the crowd who mention it. However, most of the proposed techniques merely rely on early popularity measurements and consider the prediction of a single topic or article's popularity as a separate isolated task by neglecting the underlying connection between the predicted topic or article and other news items. Intuitively, the popularity of a topic or article is often correlated with or even dependent on other relevant topics/articles. For example, nobody would talk about #MakeAmericaGreatAgain, or the wall to Mexico without Donald Trump. Therefore, it would clearly be useful if we can incorporate the relevance of news topics in the predictive model to improve prediction performance.

In this work, we develop a unified topic-popularity forecasting framework which can be applied to both social networking platforms and news feeds. In particular, we study the complex relationships among topics through the structure of the underlying topic-interaction networks. The proposed scheme for topic prediction involves word embedding, unsupervised clustering, network structure analysis and deep learning for time series forecasting. Firstly, frequent words extracted from original texts (e.g. tweets, news articles) are represented as high dimensional vectors using Word2Vec embeddings, which are further clustered into a number of meaningful topics by the K-means algorithm complemented with manually labelling. These well-defined topics are then denoted as nodes in the network where links between two topics are created if the same user has worked on both of them, such that a series of temporal topic-networks can be created. After that we use structure and dynamic properties of the network combined with information about content and past activity to train a long short-term memory (LSTM) (Hochreiter and Schmidhuber 1997) model, which predicts the number of mentions of a specific topic on the subsequent day. Furthermore, we propose a simple but powerful approach to select relevant topics as additional features using the network's historical information about the topic's local neighbors. We show that using these topics as additional features can significantly improve the prediction results and outperforms the existing approaches including the linear-correlation feature selection method (Abbar et al. 2018). Extensive experiments on two Twitter datasets and the HuffPost news feeds dataset demonstrate the effectiveness and efficiency of the proposed scheme.

The remainder of this manuscript is organized as follows. In Section 2, we provide an

overview of related work. Section 3 introduces the datasets used in this work. The method of topic generation is given in Section 4. Section 5 presents the topic-network creation as well as visualization results from three datasets. The measurement of popularity and features are provided in Section 6, along with the proposed topic selection method. Section 7 introduces the prediction task's results. Finally, Section 8 contains conclusions and further applications and extensions.

## 2. Related Work

In this section, we summarize existing researches which are closely related to the study presented in this paper, grouped by the different features used in the predictive models.

**Single past time series.** When predicting how active the crowd behavior in the future will be, a typical approach is to look at the *single* past time series of such activity and employ regression-based techniques. For instance, Szabo and Huberman (2010) found a strong correlation between the logarithmically transformed popularities at early and later times. They showed that the total number of views of YouTube videos and votes of Digg stories can be predicted by measuring the observed views/votes at an earlier time. In Kim et al. (2011), authors predicted the popularity (total number of visits) of an article in a Web-blog with more than 70% accuracy, by only exploiting visits in the first 30 minutes. Pinto et al. (2013) proposed multivariate linear regression models to predict the future popularity (views) of a given YouTube video by using previous daily popularity as input. An agnostic feature space was proposed in Ahmed et al. (2013) to capture the patterns of user behavior overtime, which is able to categorize and predict the growth in popularity of content on YouTube, Digg and Vimeo datasets.

**Exogenous features.** Recently there have been efforts towards building sophisticated predictive models incorporating various *exogenous* features beyond the past activity such as user interaction, social network structure and other characteristics of social media content. For instance, Tatar et al. (2011 2012) predicted the number of comments to news articles on a French news platform, based on earlier measurements such as publication date, category, section and number of comments. Bandari et al. (2012) used the category of a news article together with information about the communication source, language subjectivity, and named entities to predict the popularity of news articles before they go online. Regarding network impact, Ruan et al. (2012) showed that combining past activity with social network features (e.g. the amount of topic-related information a user has received from his/her friends) can improve the prediction performance of a topic's volume on Twitter. The effect of a user-interaction network was also validated in Weng et al. (2014) where features from basic network topology and community structure turned out to be the most effective predictors. Besides, emotion features conveyed in news headlines also demonstrated predictive power for news popularity (Gupta et al. 2019).

**Topic-relevance features.** Among existing work that we reviewed, relatively little has been done studying the relationship between topics to predict news popularity. The most related existing work we can find is Abbar et al. (2018), which proposed to utilize the similarity of articles to enhance the predictability of an article's popularity before its date of publication. The basic idea behind this work is to select a set of articles in recent publications as additional features whose volumes have strong cross correlation (measured by Pearson coefficient) with the predicted article. Using a collection of articles, the authors showed that the use of this

dependency can improve the accuracy of topic or article popularity prediction. However, in this work the article similarity is measured linearly, which is insufficient to capture the complex and dynamic interactions among different articles or topics. Our method differs from this approach in that we map the relationship between topics into a network representation.

## 3. Dataset

In this section, we describe three datasets (two from Twitter and one from news feeds) that will be used in the remainder of this study.

### 3.1 Twitter Tribes

We collect data from Twitter, one of most popular online social network platforms on which messages, often known as "tweets", are posted and interacted with by registered users. In particular, we focus on analyzing Twitter's virtual-tribe, defined as a group of heterogeneous users who shares common opinions, emotions and vision of life (Cova and Cova 2002). Recently, based upon latest developments in AI, a novel tool called *Tribefinder* (Gloor et al. 2020) was developed to automatically identify virtual tribes on Twitter. Its basic idea is that members of the same virtual tribe use similar language. Specifically, *TribeFinder* first creates tribes by automatically searching a large sample of Twitter users according to initial given keywords expressing concepts, ideas and beliefs associated with a particular tribe. Then *TribeFinder* analyzes the language of these users and identifies the textual patterns that characterize each tribe through deep learning. After training, for arbitrary individuals, *TribeFinder* will predict a tribe for each of their tweets and sum up the result to compute a tribe allocation for that user. In this work, we adopt the tribal category of alternative realities defined in an earlier work (Gloor et al. 2019) and consider two specific tribes within it: *Treehuggers* and *Fatherlanders*. *Treehuggers* is a group of environmentalists who strive to protect nature from phenomena like global warming (Gloor et al. 2019). They believe in the limits of growth and in the protection of nature and challenge some elements of technological progress such as gene manipulation. *Fatherlanders* are extreme patriots whose vision is a recreation of the national states from the 1900s. They have unshakable faith in God and their fatherland. They cling to the good old times, hold the idea of the family in high regard and have little time for foreigners (Gloor et al. 2019). These two tribes are representative for two of the major topics of our time because *Fatherlanders* stand for ultra-nationalism represented by the likes of Erdogan, Bolsonaro, Trump, or Johnson, while *Treehuggers* stand for another other major issue of our time, global warming, represented by Greta Thunberg and her allies.

To identify users belonging to either one of two tribes on Twitter, we use a trained *TribeFinder* system built in Condor (Gloor 2017), which is a powerful social media analysis tool. First, *TribeFinder* automatically fetches *Treehuggers* and *Fatherlanders* separately on Twitter according to pretrained patterns. For each tribe, we select the most active 2000 Twitter accounts in terms of the total number of tweets for subsequent analysis. We then collect their published tweets (including posted and retweeted ones) from the end of 2018 to early 2020 through *TribeFinder*. Each tweet consists of the full text, date and the id of the user who posts or retweets the tweet. We clean the tweet's text by removing the "@" mention (another user that the tweet mentioned), URL links and all the non-letter characters including numbers. Then a tweet's sentences are tokenized with all the letters set to lowercase. After data cleaning, we obtain two Twitter datasets, *Treehuggers* and *Fatherlanders* with

**Table 1** Descriptive Information of Three Datasets Studied in This Work

| Dataset | Sample period | # of tweets | # of unique words | # of high frequency words | # of high frequency words included in pre-trained *Glove* model |
|---|---|---|---|---|---|
| *Treehuggers* (Twitter) | 2018.10.01-2020.01.30 | 232162 | 111975 | 6131 | 5950 |
| *Fatherlanders* (Twitter) | 2018.12.01-2020.01.30 | 783467 | 103964 | 11131 | 10626 |
| *HuffPost news* | 2012.01.28-2018.05.26 | 148933 | 75241 | 7351 | 7334 |

about 200K and 700K tweets, respectively.

### 3.2 HuffPost News Feeds

The news feeds dataset used in this study is provided by Misra (2018), which contains a variety of news from 2012 to 2018 obtained from HuffPost. For each news article, we consider the headline, a short description of the article, date and the person who authored the article. Since the short description can often provide useful information about the article, we concatenate the headline of the article with its description and view it as an integrated text of an article. The original dataset also contains the category each article belongs to (e.g. politics, wellness, entertainment). We do not need this information in our proposed scheme since topics are identified according to words rather than articles, as explained later. Like the Twitter data, we employ a similar data cleaning process for the articles' texts, resulting in a *HuffPost news* dataset with about 150K articles, which will be used for further analysis.

## 4. Topic Generation

The generation of topics includes three steps. First, we use word embeddings to assign each word to a multi-dimensional vector. Second, K-means clustering is applied to these vectors to uncover topics in our datasets. Finally, meaningful topics are named which is done by manually labelling.

### 4.1 Word Embedding

Word embeddings, as a feature learning technique in NLP tasks, has gained prominent attention in the literature. Among these techniques, *GloVe* (Global Vectors for Word Representation, Pinto et al. (2013)) is one of the most frequently used learning algorithms for obtaining vector representations for words. In this work, we adopt the 200-dimension version of the pre-trained *Glove* model publicly available as our word representation method. For each dataset, we collect a set of all unique words within it. We then remove stop words as well as words which are not commonly used. Specifically, words are considered to be low frequency if they appear less than 50 times in the whole dataset and thus will be deleted. After that, we search the remaining *high frequency* words in the pre-trained *Glove* model for their vector representations. If a word is absent from the model, we simply discard it. Table 1 presents the descriptive statistics for the three datasets.

### 4.2 Clustering

After obtaining the vector representation of words, we extract the underlying topics by using K-means (Arthur and Vassilvitskii 2006) clustering, which is an intuitive and well-suited method integrated with our framework. Firstly, K-means is a hard-clustering method (Arthur and Vassilvitskii 2006) whereby the words are clustered into $k$ disjoint clusters such that one word only belongs to one cluster. This

is convenient for our algorithm to assign a unique underlying topic for each cluster later. Secondly, K-means can use an arbitrary type of word vector as input feature and thus can be combined with the pre-trained *Glove* word embeddings. Thirdly, its implementation is simple with low computational complexity. By contrast, other soft-clustering methods such as Latent Dirichlet allocation (Blei et al. 2003) may be infeasible in our scenario since they either cluster documents rather than words or a single sample can belong to multiple clusters. We find an appropriate number of topics $k^*$ by testing various $k$ and evaluate the clustering performance in term of the average silhouette coefficient (Rousseeuw 1987, Amorim and Hennig 2015). Generally speaking, the higher an average silhouette coefficient is, the more effective and reasonable the clustering result is. By testing different $k$ from 20 to 150, we find that, for all three datasets, a higher silhouette coefficient falls in a similar $k$'s range between 60 and 70 (see Figure 1 for the dependency of the silhouette coefficient on $k$). Note that a subtle difference of $k^*$ will not significantly affect our subsequent analysis as the proposed method will further check the clustering results to ensure the topic's substance as described later. Therefore, for the convenience of the analysis, we set the number of clusters $k^*$ to 70 for all three datasets.

### 4.3 Manually Labelling

Since there is no objective measure of accuracy available to check the results from the unsupervised algorithm, we conducted careful manual verifications to interpret the results of $k^*$ clusters. Although some clusters look like spillovers from other unassigned words that are thrown together by the K-means algorithm, we do find that more than half of the clusters contain meaningful homogeneous words and thus can be assigned clear topics.

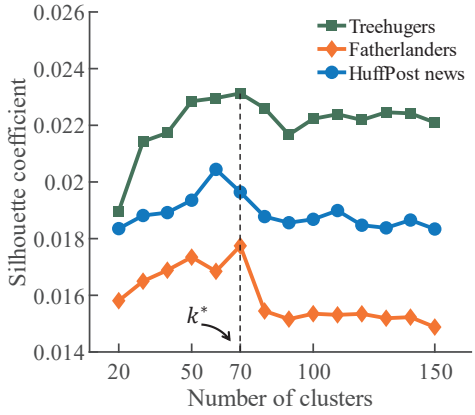


**Figure 1** Relationship between the Number of Clusters and the Clustering Performance in Terms of Silhouette Coefficient

Consequently, we identify the number of topics $N = 38$ for the *Treehuggers* and *Fatherlanders* datasets and $N = 39$ for the *HuffPost news* datasets. Table 2 presents the identified topics of the *Treehuggers* along with the top-5 words by frequency within each of them (See Appendix for the generated topics of the *Fatherlanders* datasets and the *HuffPost news* datasets). Since the topics are generated automatically based on the textual patterns in each dataset, they are representative in terms of coverage. As evidence, each dataset not only contains commonly shared topics (e.g. job, economy, country), but also contains its own unique topics. For instance, several topics relevant for politics can be found in the *Fatherlanders* dataset such as 'top politicians' and 'ideology' (see Appendix Table 6), while 'environment', and 'pollution' appear in the *Treehuggers* dataset.

## 5. Topic-Network Representation

In this section, we introduce how to map the relationship between topics into a network representation. Visualizations of topic-networks are also illustrated and the evolution of the network's structure for three datasets are discussed.

**Table 2** Generated Topics ($N = 38$) from Words for *Treehuggers* Dataset

| No. | Topic | Top-5 words by frequency | No. | Topic | Top-5 words by frequency |
|---|---|---|---|---|---|
| 1 | **nation** | country, American, national, Indian, Jewish | 20 | **politicians** | Trump, Dorian, Warren, Graham, Sanders |
| 2 | **Jews** | Jews, religious, holocaust, misinformation, radical | 21 | **government** | court, government, law, justice, political |
| 3 | **news** | daily, journal, news, latest, CNN | 22 | **police** | police, killed, FBI, arrested, shooting |
| 4 | **pollution** | clear, light, plastic, natural, carbon | 23 | **business** | media, social, job, community, business |
| 5 | **war** | attack, death, war, action, dead | 24 | **natural scenery** | nature, ocean, park, view, beach |
| 6 | **photo** | shared, photo, release, photos, image | 25 | **India** | Delhi, Mumbai, Kashmir, ipl, Chennai |
| 7 | **school** | book, school, history, group, students | 26 | **family** | family, children, kids, child, mother |
| 8 | **society** | human, freedom, faith, society, movement | 27 | **president** | president, impeachment, democrats, congress, election |
| 9 | **transport** | travel, car, road, crash, flight | 28 | **fashion & shopping** | set, thread, art, museum, style |
| 10 | **economy** | economy, trade, tax, oil, economic, market | 29 | **leisure** | chrismas, gift, including, lots, giveaway |
| 11 | **Gandhi** | bjp, modi, Gandhi, Singh, shri | 30 | **color & suit** | white, black, red, blue, green |
| 12 | **legal** | journalist, rabbi, reported, lawyer, activist | 31 | **US states** | western, state, south, California, York |
| 13 | **home** | house, near, outside, inside, room | 32 | **animal** | animals, dog, species, dogs, birds |
| 14 | **video** | video, live, story, breaking, shows | 33 | **money** | million, money, save, pay, worth |
| 15 | **IT** | power, case, air, Amazon, phone | 34 | **health** | health, research, safety, medical, aid |
| 16 | **sport** | win, team, nation, game, season | 35 | **Israel** | Israel, Iran, border, Ukraine, Israeli |
| 17 | **countries** | India, Nigeria, America, china, Australia | 36 | **food** | water, food, ice, eat, coffee |
| 18 | **office** | public, minister, security, chief, leader | 37 | **disease** | cancer, abuse, risk, mental, loss, pressure |
| 19 | **environment** | wildlife, plant, climate, environment, biodiversity | 38 | **weather** | weather, hurricane, snow, rain, storm |

## 5.1 Network Creation

For each day, a new topic-network is created. Such a topic-network can be viewed as a simple undirected graph $G = (V, E)$, where $V$ denotes the set of nodes corresponding to the topics defined in Section 4.3 which remain unchanged across time, while $E$ denotes the set of internal links between topics, which is constructed according to *whether the same user (author) has worked on both topics on a specific day.* A similar idea has been successfully used in creating information networks like academic citation (Cawkell 1971, Hummon and Dereian 1989) and knowledge mapping of Wikipedia (Kleeb et al. 2012). The network would be undesirably dense if one would create a link for two topics as long as they are mentioned by the same user. To extract useful information embedded in the interaction of topics, we only draw *high frequency* links by using a weight threshold $\omega$. Specifically, a link $e_{ij} \in E$ is constructed only if topics $i$ and $j$ are mentioned on a specific day

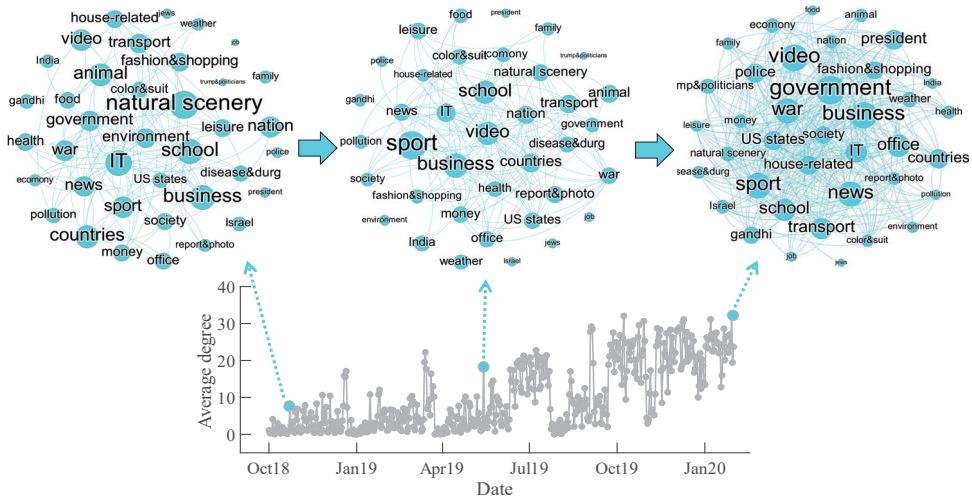


**Figure 2** Topic-Network Evolution on the *Treehuggers* Dataset

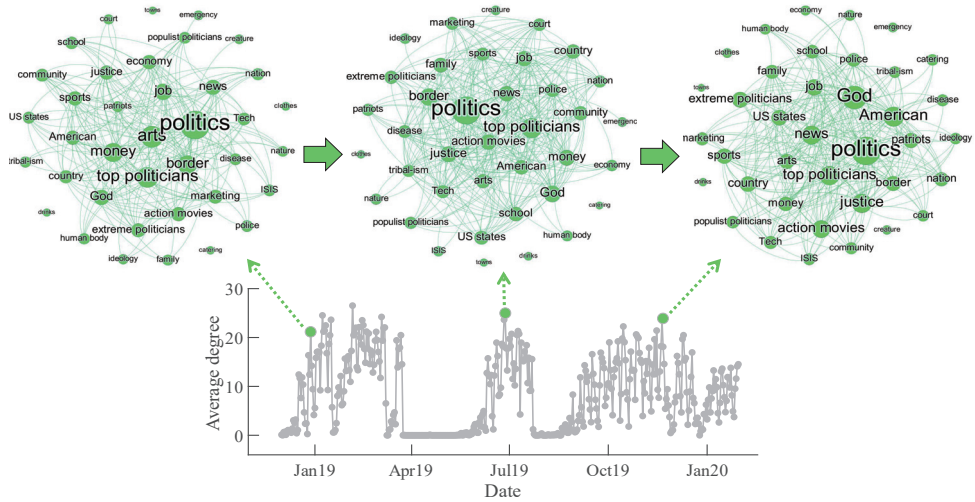


**Figure 3** Topic-network Evolution on the *Fatherlanders* Dataset

by at least $\omega$ individuals.

## 5.2 Visualization

To prevent the generated networks from being neither too sparse nor too dense, we set the link-weight threshold $\omega$ to 5, 30 and 2 for *Treehuggers*, *Fatherlanders* and *HuffPost news* datasets, respectively, according to the varying number of daily published contents in the three datasets. Figure 2 illustrates the variation of a topic-network's average degree over time by showing three snapshots of network visualizations selected at the beginning, middle and the end of the entire period. As shown in Figure 2, one can observe a roughly increasing trend of average degree in the *Treehuggers* datasets, implying growing activity in this group over time. The main topic discussed by these environmentalists changes greatly, ranging from the natural scenery, sport to government. On the other hand, the network average degree of the *Fatherlanders* dataset changes dynamically. However, when looking at the network structure, one can observe that the politics-related topics are constantly the most discussed top-

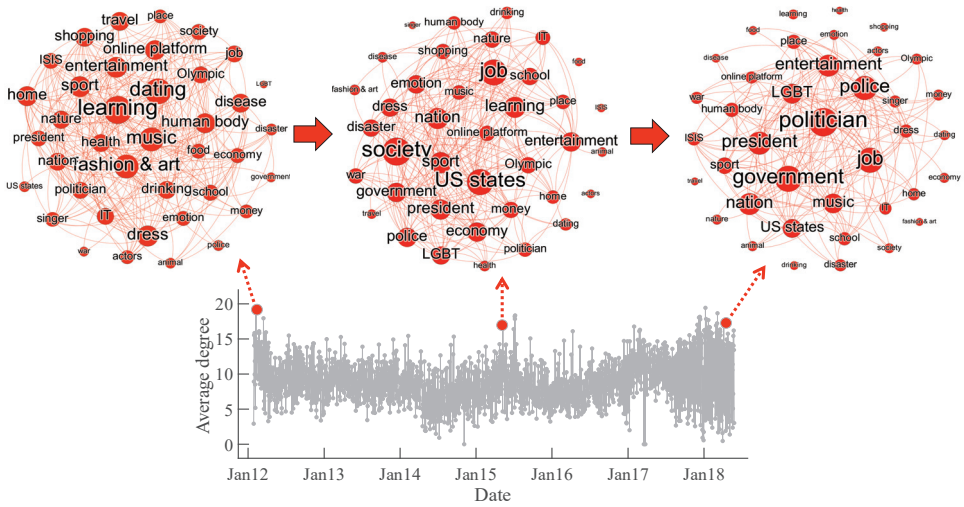


**Figure 4** Topic-Network Evolution on the *HuffPost news* Dataset

ics with the term "politics" being most popular (see Figure 3). Regarding the *HuffPost news* dataset, the average degree of the network remains relatively stable given that the number of news is roughly identical each day. Due to the nature of daily news, many topics are widely discussed with sufficient popularity and there is no obviously most active one (see Figure 4). The diversity of the topic-network's structure will be utilized in our prediction model.

## 6. Measuring Topic's Popularity and Features

In this section, we define a topic's popularity. Then we introduce the features extracted from networks and contents. Finally, we present the proposed network structure based method to select relevant topics.

### 6.1 Topic's Popularity

In this work, a topic's popularity is measured by *the size of the crowd who mentions it*. Specifically, for Twitter data, it is defined as the number of registered users adopting (e.g. posting or retweeting) tweets that contain the given topic, while for news feeds data, it is defined as the number of authors who work at the articles involving the topic. Similar notations of a topic's popularity have been used in prior research (Weng et al. 2014). Mathematically, the popularity $f_t(u)$ of the topic $u$ at day $t$ is given by:

$$f_t(u) = \sum_{k=1}^{M} \delta_u(k,t) \tag{1}$$

where $\delta_u(k,t) = 1$ if user $k$ mentions a given topic $u$ at day $t$ and $\delta_u(k,t) = 0$ otherwise; $M$ is the total number of individuals in the dataset.

We implement $f_u(t)$ in Eqn. 1 as follows: For each user (author) $k$ at day $t$, we collect his/her tweets (articles). From these tweets we identify high frequency words included in a pre-trained Glove model. Then we map these words onto the topics defined in Section 4.3, resulting in a set of topics used by user $k$ at day $t$. For each mapped topic, we simply increment its popularity by one. Note that a single tweet (article) may belong to multiple topics. We carry out the above calculation for every user. The popularity $f_u(t)$ for all topics u can be obtained at any time.

To reduce the fluctuation of the original popularity across time, we denote a *smoothed* topic's popularity $f_u^s(t)$ as the rolling sum of popularities over a fixed duration of $s$ days,

i.e.:

$$f_u^s(t) = f_u(t) + f_u(t-1) + \cdots + f_u(t-s) \quad (2)$$

which will be used as the objective in the prediction task, as described in the next section.

### 6.2 Prediction Features

In this subsection we identify features used for our prediction model. In particular, we use network and graph theory to examine the structure of topic networks. We also include content and early observations of popularity as complementary inputs. Consequently, for each topic, we study a total of 5 features denoted by $f_1 - f_5$, which can be divided into four groups: node position, position variation, content features and past popularity.

1. **Node position (NP)**: We first investigate the position of a topic (node) in a topic-network which can be characterized by its nodal centrality. In graph theory, indicators of nodal centrality identify the most important node within a graph (Newman 2018). In our scenario, centrality corresponds to the degree of interaction between a topic and other topics and thus quantifies the influence of a topic in the network. In particular, we compute the betweenness centrality ($f_1$) of each node, which measures the probability with which a node is on the shortest paths between other nodes (Freeman 1977). Mathematically, it is given by:

$$BC(v) = \sum_{i,j \neq v} \frac{m_{ij}(v)}{m_{ij}} \quad (3)$$

where $m_{ij}$ corresponds to the number of shortest paths between $i$ and $j$, $m_{ij}(v)$ corresponds to the number of times $v$ passes through these paths.

2. **Position variation (PV)**: Another important feature is the temporal variation of a topic's network position given that the structure of a topic network changes dynamically over time. We specifically focus on the nodal betweenness oscillation ($f_2$), whose utility has been validated repeatedly in social network analysis (Kidane and Gloor 2007, Antonacci et al. 2017, Gloor 2017). But it remains to be verified in dynamic topic networks. The betweenness oscillation is calculated by quantifying the number of local maxima and minima in the betweenness centrality curve during a time window (Gloor 2017), whose length is set to 7 days (i.e., a week) in this study. To align with the other features, we denote the betweenness oscillation at time t as the value calculated from the time window $\{t-6, \cdots, t-1, t\}$.
3. **Content features (CF)**: Two kinds of content features, sentiment ($f_3$) and complexity ($f_4$) are considered. A higher sentiment score represents stronger positivity of the language used. In particular, we employ the sentiment analysis tool VADER (Gilbert and Eric 2014), to calculate sentiment for each tweet (article). On the other hand, the complexity score measures the informativeness of a given content (Aral and Alstyne 2011), which is calculated using the likelihood of each word to appear in the text based on the term frequency-inverse document frequency (TF-IDF). A text's complexity score is then the average of all TF-IDF values of the words occurring in the text. The scores of sentiment and complexity are calculated at the text level. To make them useful for our topic's popularity prediction, we define topic-level sentiment and complexity scores as the mean value of all texts' sentiment or complexity that contain the specific topic.

4. **Past popularity (PP)**: Past time series of popularity ($f_5$) has been shown as a good predictor (Ruan et al. 2012) for the crowd's future behavior in social media and thus is included in our analysis.

Given that we have nearly 40 topics for each dataset, there are about $5 \times 40 = 200$ input features, Therefore, the problem is how to identify an effective subset from these features, which will be described in the following subsection.

### 6.3 Topic Selection

Inputting all features for prediction might be undesirable, in part because some topics might not be relevant to the target topic and thus causing reductant noise, and in part because too many features might lead to over-fitting during the training process. In this work, we propose a topic selection process where topics are selected according to historical information about each topic $u$'s neighbors in the network structure. More specifically, instead of using all topics, we only select $K$ topics that are the most frequent neighbors of $u$ (i.e., having the largest cumulative link connection times) based on the time sequence of topic-networks presented in Section 5. Using the features of these $K$ topics along with the topic $u$ itself can indeed lead to better performance than using all features and existing feature selection method, as we will show later.

## 7. Topic Popularity Prediction

In this section, we present the prediction results of a topic's popularity based on the preliminary analyses introduced above.

### 7.1 Task Definition

We consider the forecasting of a topic's popularity as a time-series *regression* prediction problem, which is defined as predicting the continuous value of mentions of a specific topic on the subsequent day $t+1$ using the acquired historical features at previous $\tau$ days $\{t-\tau, \cdots, t-1, t\}$. In particular, we use the smoothed popularity $f_u^s(t+1)$ in Eqn. (2) as our prediction objective $y_{t+1}$ to reduce the fluctuation of the raw popularity curve and improve the prediction performance. Note that the original popularity $f_u(t+1)$ can be easily obtained by subtracting the true observations at the prior time steps $\{t-s, \cdots, t-1, t\}$ from the forecasting value. Similarly, we smoothen the features described in Section 5.2 with the same smoothened factor $s$ except the betweenness oscillation as it is already a statistical value over a time window, resulting in a feature set $\mathbf{x}_t$. Consequently, the prediction task is to predict $y_{t+1}$ given the smoothened historical features $\{\mathbf{x}_{t-\tau}, \cdots, \mathbf{x}_{t-1}, \mathbf{x}_t\}$, which can be solved by machine learning.

### 7.2 LSTM Model

The LSTM neural network architecture is employed in this work based on previous successful use in many practical applications involving sequential data, such as speech recognition (Graves et al. 2013), image generation (Gregor et al. 2015) and human action detection (Baccouche1 et al. 2011). Typically, a LSTM basic unit, often called a cell, consists of three gating functions: 1) the input gate processing new inputs, 2) the forget gate deciding what information should be thrown away or kept and 3) the output gate controlling the extent to which the value in the cell is used to compute the output activation. At each timestamp, LSTM maintains and updates the weight matrix along with hidden vectors and memory vectors by incorporating the above gating functions into their state dynamic.

In this work, the LSTM last layer's output is fed to a fully connected layer, in which the number of input features is the number of output units in LSTM and the output size is 1 since we only forecast the topic's popularity on the subsequent day. We adopt the mean

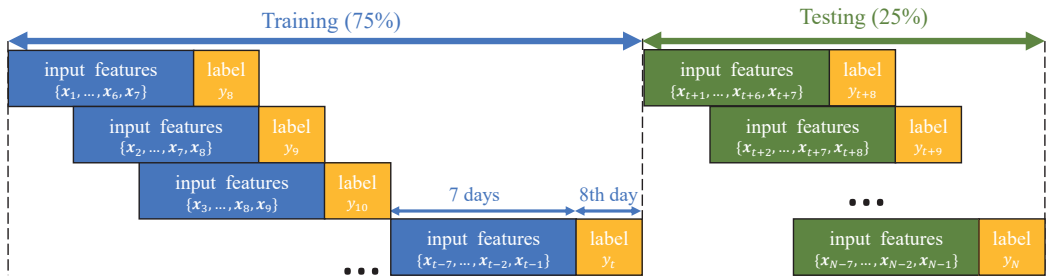


**Figure 5** Dataset Arrangement for Training and Testing During the Whole Sample period $N$ days

square error (MSE) criterion as the loss function. To minimize MSE, an Adam optimizer is employed in the gradient descent process.

### 7.3 Performance Evaluation

To evaluate the prediction accuracy, we measure three different criteria, root mean square error (RMSE), mean absolute percentage error (MAPE) and the Pearson correlation coefficient (R). The RMSE is given by

$$\text{RMSE} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(y_t - \tilde{y}_t)^2} \tag{4}$$

where $T$ denotes the number of test samples and $y_t$,$\tilde{y}_t$ denote the true and the forecasted value, respectively. MAPE measures the size of error, calculated as

$$\text{MAPE} = \frac{1}{T}\sum_{t=1}^{T}\left|\frac{y_t - \tilde{y}_t}{y_t}\right| \tag{5}$$

The Pearson correlation coefficient R quantifies the linear correlation between two variables, defined as

$$\text{R} = \frac{\sum_{t=1}^{T}(y_t - \langle y_t\rangle)(\tilde{y}_t - \langle \tilde{y}_t\rangle)}{\sqrt{\sum_{t=1}^{T}(y_t - \langle y_t\rangle)^2(\tilde{y}_t - \langle \tilde{y}_t\rangle)^2}} \tag{6}$$

where $\langle y_t\rangle$ and $\langle \tilde{y}_t\rangle$ denote the mean of the real and the forecasted values, respectively. Generally, a lower RMSE or MAPE and a higher R value indicate better prediction performance.

We also validate the statistical difference of forecasting accuracy among different models using the Diebold Mariano (DM) test (Harvey et al. 1997, Diebold and Mariano 2002). Suppose that the difference between the first list of predictions and the actual values is $\{e_t^A\}_{1\le t\le T}$ and the second list of predictions and the actual values is $\{e_t^B\}_{1\le t\le T}$. We define the loss differential $d$ as:

$$d = \{e_t^{A\,2} - e_t^{B\,2}\}_{1\le t\le T} \tag{7}$$

The DM test evaluates the null hypothesis $H_0$ of the expectation of $d$ being zero, i.e., the accuracy of two models being statistically equivalent.

### 7.4 Experiment Results

In our experiment, we select the features of the past week, i.e., $\tau = 7$, to predict the one-step-ahead (8th day) popularity of a given topic. Using the LSTM architecture, the forget gating function can learn the optimal timestep by forgetting irrelevant data and thus an explicit investigation of choosing $\tau$ is not included in this study. To validate the model performance of our time series data, we apply a rolling window procedure to generate training and testing samples. Specifically, for the dataset where each point corresponds to a single day's observation, we first split the dataset into two separate parts to avoid that future observations will be used for constructing the forecast. The oldest 75% of data will be taken for the training set and the remaining 25% of data will be used for the test set. Then within each set, the model is fitted by a rolling-forward window of $\tau + 1$

**Table 3** Predictive Performance of Popularity - Feature Experiment Results on the Testing Set of Three Datasets

**Panel A. *Treehuggers* dataset**

| Model | Input feature | 'Nation' | | | 'Environment' | | | 'Sport' | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | RMSE | MAPE | R | RMSE | MAPE | R | RMSE | MAPE | R |
| Target topic | PP(1) | 13.520 | 0.155 | 0.622 | 9.01 | 0.175 | 0.601 | 18.524 | 0.136 | 0.695 |
| | PP+NP+PV+CF (5) | 12.544 | 0.145 | 0.674 | 8.707 | 0.176 | 0.627 | 17.839 | 0.132 | 0.707 |
| All topics | PP(38) | 14.370 | 0.171 | 0.718 | 10.257 | 0.210 | 0.603 | 23.136 | 0.189 | 0.710 |
| | PP+ NP (76) | 14.600 | 0.189 | 0.694 | 11.183 | 0.215 | 0.643 | 21.185 | 0.155 | 0.616 |
| | PP+ PV (76) | 14.440 | 0.177 | 0.659 | 10.242 | 0.228 | 0.574 | 41.836 | 0.340 | 0.522 |
| | PP+CF (114) | 12.571 | 0.159 | 0.729 | 7.947 | 0.171 | 0.702 | 15.600 | 0.114 | 0.758 |
| | PP+NP+PV+CF (190) | 16.952 | 0.186 | 0.625 | 11.127 | 0.220 | 0.435 | 27.945 | 0.207 | 0.506 |
| Pearson correlation based | $\eta$ = 4 (20) | 11.540 | 0.129 | 0.767 | **7.415** | 0.156 | 0.763 | 15.256 | 0.120 | 0.764 |
| | $\eta$ = 6 (30) | 12.080 | 0.141 | 0.750 | 8.133 | 0.159 | 0.684 | 16.800 | 0.128 | 0.745 |
| | $\eta$ = 8(40) | 11.590 | 0.137 | 0.757 | 8.565 | 0.188 | 0.670 | 17.862 | 0.135 | 0.727 |
| Network structure based | Target + top-3 neighbors (20) | **10.060** | **0.112** | **0.827** | 7.645 | 0.152 | 0.745 | **13.839** | **0.0979** | **0.825** |
| | Target + top-5 neighbors (30) | 11.566 | 0.122 | 0.783 | 7.450 | **0.137** | **0.784** | 14.990 | 0.121 | 0.768 |
| | Target + top-7 neighbors (40) | 12.212 | 0.129 | 0.768 | 8.174 | 0.162 | 0.677 | 14.300 | 0.110 | 0.798 |

**Panel B. *Fatherlanders* dataset**

| Model | Input feature | 'Politics' | | | 'God' | | | 'Populist politicians' | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | RMSE | MAPE | R | RMSE | MAPE | R | RMSE | MAPE | R |
| Target topic | PP(1) | 126.487 | 0.151 | 0.663 | 51.513 | 0.112 | 0.610 | 45.318 | 0.182 | 0.697 |
| | PP+NP+PV+CF (5) | 124.983 | 0.147 | 0.680 | 47.176 | 0.110 | 0.669 | 45.291 | 0.177 | 0.719 |
| All topics | PP(38) | 127.782 | 0.129 | 0.740 | 55.907 | 0.109 | 0.701 | 41.953 | 0.166 | 0.747 |
| | PP+ NP (76) | 272.928 | 0.240 | 0.315 | 207.216 | 0.392 | -0.08 | 55.209 | 0.246 | 0.669 |
| | PP+ PV (76) | 157.033 | 0.171 | 0.591 | 99.349 | 0.209 | 0.274 | 67.116 | 0.272 | 0.468 |
| | PP+CF (114) | 100.077 | 0.107 | 0.838 | 46.261 | 0.098 | 0.726 | 42.459 | 0.147 | 0.745 |
| | PP+NP+PV+CF (190) | 175.908 | 0.194 | 0.557 | 162.186 | 0.360 | 0.286 | 69.095 | 0.284 | 0.474 |
| Pearson correlation based | $\eta$ = 4 (20) | 91.084 | **0.094** | 0.855 | 39.809 | 0.089 | 0.800 | 35.009 | 0.130 | 0.859 |
| | $\eta$ = 6 (30) | 188.680 | 0.159 | 0.561 | 48.665 | 0.098 | 0.729 | 38.839 | 0.154 | 0.809 |
| | $\eta$ = 8(40) | 158.077 | 0.132 | 0.568 | 52.392 | 0.114 | 0.668 | 42.404 | 0.157 | 0.783 |
| Network structure based | Target + top-3 neighbors (20) | 89.542 | 0.100 | 0.852 | **36.482** | **0.078** | **0.822** | 31.930 | 0.129 | 0.875 |
| | Target + top-5 neighbors (30) | **83.299** | **0.094** | **0.874** | 48.778 | 0.106 | 0.746 | **31.190** | **0.122** | **0.887** |
| | Target + top-7 neighbors (40) | 87.210 | **0.094** | 0.861 | 50.898 | 0.114 | 0.718 | 42.247 | 0.161 | 0.763 |

**Panel C. *HuffPost News* dataset**

| Model | Input feature | 'Economy' | | | 'President' | | | 'Travel' | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | RMSE | MAPE | R | RMSE | MAPE | R | RMSE | MAPE | R |
| Target topic | PP(1) | 5.105 | 0.387 | 0.522 | 14.483 | 0.263 | 0.747 | 4.640 | 0.471 | 0.623 |
| | PP+NP+PV+CF (5) | 5.086 | 0.343 | 0.560 | 13.346 | 0.246 | 0.779 | 3.647 | 0.310 | 0.665 |
| All topics | PP(39) | 4.531 | 0.324 | 0.651 | 16.343 | 0.317 | 0.692 | 3.462 | 0.298 | 0.709 |
| | PP+ NP (78) | 5.344 | 0.281 | 0.654 | 14.446 | 0.274 | 0.744 | 4.320 | 0.313 | 0.562 |
| | PP+ PV (78) | 4.223 | 0.292 | 0.698 | 17.463 | 0.342 | 0.764 | 4.405 | 0.351 | 0.435 |
| | PP+CF (117) | 4.921 | 0.343 | 0.555 | 15.422 | 0.267 | 0.794 | 4.095 | 0.353 | 0.544 |
| | PP+NP+PV+CF (195) | 5.627 | 0.355 | 0.347 | 17.351 | 0.330 | 0.649 | 4.500 | 0.400 | 0.401 |
| Pearson correlation based | $\eta$ = 4 (20) | 4.427 | 0.297 | 0.655 | 12.015 | 0.213 | 0.823 | 3.582 | 0.276 | 0.710 |
| | $\eta$ = 6 (30) | 4.189 | 0.269 | 0.721 | 13.184 | 0.240 | 0.784 | 3.645 | 0.318 | 0.669 |
| | $\eta$ = 8(40) | 4.653 | 0.320 | 0.604 | 12.774 | 0.235 | 0.840 | 3.631 | 0.276 | 0.688 |
| Network structure based | Target + top-3 neighbors (20) | 3.518 | 0.235 | 0.799 | 9.487 | 0.166 | 0.895 | 3.727 | 0.321 | 0.656 |
| | Target + top-5 neighbors (30) | **3.463** | **0.216** | **0.822** | **8.329** | **0.143** | **0.923** | 3.315 | 0.278 | 0.737 |
| | Target + top-7 neighbors (40) | 3.875 | 0.248 | 0.757 | 10.470 | 0.171 | 0.869 | **2.954** | **0.245** | **0.796** |

length where the past $\tau$ observations' features are taken as LSTM inputs and 1-step ahead observation is made for the prediction (see Figure 5 for the prediction procedure illustration). Since each topic may exhibit different patterns of its popularity and thus there is not a universal model able to capture all patterns, we create a unique model for each specific topic. Without bias, here we specify three different topics for each dataset as illustrations of the prediction. Specifically, the three topics are 'nation', 'environment' and 'sport' in *Treehuggers*, 'politics', 'God' and 'populist politicians' in *Fatherlanders*, 'economy', 'president' and 'travel' in *HuffPost news*.

Regarding other parameters, the popularity as well as the input features are smoothened by 2 days, i.e., $s = 2$, to reduce fluctuation

**Table 4** Diebold-Mariano Test Statistic of the Proposed Network Structure Based Method Compared with Other Forecasting Methods

| Dataset | Topic | Network structure based vs. | | |
|---|---|---|---|---|
| | | Target topics | All topics | Pearson correlation based |
| *Treehuggers* (Twitter) | 'Nation' | 3.58*** | 2.78*** | 2.06* |
| | 'Environment' | 3.41*** | 3.44*** | 3.06** |
| | 'Sport' | 6.78*** | 5.85*** | 2.90** |
| *Fatherlander* (Twitter) | 'Politics' | 5.33*** | 4.45*** | 3.97*** |
| | 'God' | 4.61*** | 3.45*** | 3.22** |
| | 'Populist politicians' | 3.67*** | 3.50*** | 2.65*** |
| *HuffPost News* | 'Economy' | 9.92*** | 9.25*** | 7.81*** |
| | 'President' | 15.24*** | 6.88*** | 4.56*** |
| | 'Travel' | 8.03*** | 4.93*** | 6.14*** |

The Stars Denote the Significance Level: * $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

within original values. A min-max normalization is applied to the data before feeding it to the LSTM model. The LSTM architecture is built using Keras, in which the number of output units of LSTM is set to 8 with the ReLU activation and the fully connected layer's activation is set to linearity to estimate the continuous value of the topic's popularity. The epoch is set to 200. The learning rate of the Adam optimizer is set to 0.1. We also implement a L1L2 with a default value of 0.001 as recurrent weight regularization to prevent overfitting.

Now we perform the LSTM experiments on the three datasets, comparing different feature selection strategies. Table 3 presents detailed results for the three datasets, in which the name of each 'input feature' denotes the used feature groups defined in Section 5.2. For instance, 'PP' means using the past activity alone as the input feature, while 'PP+NP' means the combination of past activity and network position. First, we verify the utility of network features and content features by considering the predicted topic itself. Compared against solely using the past activity, we find that lower RMSE, MAPE and higher R are always achieved by adding extra features from network and contents (see the row '*Single Target*' in Table 3). Next, we add the features from the other N-1 topics to the set of input feature. In particular, we test five different strategies, starting from using all topics' past activity (PP), then adding extra features from either one of three feature groups (NP, PV, CF) and finally build the full model (PP+NP+PV+CF). Although the full model with all features cannot improve but rather deteriorates the baseline, we do find that either one of the four other strategies achieves better result than using a single topic in different cases (see the row '*ALL*' in Table 3). This implies that features from other topics indeed introduce additional explanatory power beyond the target itself.

Finally, we apply topic selection to refine the input features. For the proposed network structure-based method, we identify the most frequent $K$ neighbors of each predicted topic according to the historical topic-networks in the training set. Then we use features of these neighbors and the predicted topic as input. We test different $K$ of 3, 5 and 7. Besides, we also employ a Pearson correlation based-method (Abbar et al. 2018) as comparison. To the best of our knowledge, limited work exists applying feature-selection for topic prediction, we find that it is a highly effective method.

**Table 5** Optimal $K$ Value and Topic's Neighbors

| Dataset | Predicted topic | Best $K$ | Top $K$ neighboring topics |
|---|---|---|---|
| *Treehuggers* (Twitter) | 'Nation' | 3 | Business, School, Countries |
| | 'Environment' | 5 | Government, Natural scenery, Countries, Business, War |
| | 'Sport' | 3 | Video, School, News |
| *Fatherlander* (Twitter) | 'Politics' | 5 | Top politicians, Job, Money, God, News, |
| | 'God' | 3 | Politics, Action movies, Top politicians |
| | 'Populist politicians' | 5 | Politics, Top politicians, Extreme Politicians, Justice, News |
| *HuffPost News* | 'Economy' | 5 | Job, Money, Nation, Society, President |
| | 'President' | 5 | Nation, Politician, Job, Government, US states |
| | 'Travel' | 7 | Place, Nation, US states, Entertainment, Job, Sport, Music |

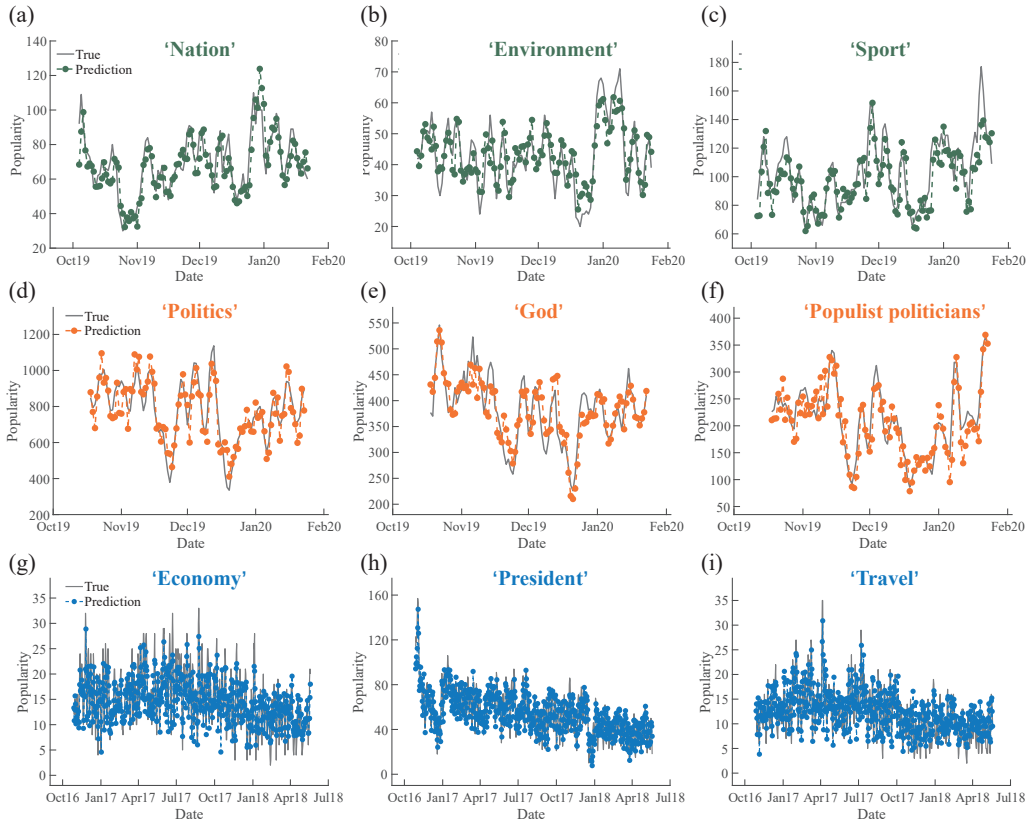


**Figure 6** The True and Predicted Popularity Obtained from the Best Model in the Testing Set (i.e., the Latest 25% of the Whole Sample Period). (a-c) *Treehuggers* (d-f) *Fatherlanders* (g-i) *HuffPost News*

In particular, we select $\eta$ topics whose popularity sequences are most correlated to the target in terms of Pearson correlation coefficient R and then use all features from these $\eta$ topics as input. The value of $\eta$ is set to 4, 6, 8 since the target itself is often included. As observed in Table 3, the network structure-based method outperforms the Pearson correlation based-method, achieving the lowest RMSE/MAPE and the highest R in almost all models. We present the DM test statistics of the network structure-based method with the compared methods in Table 4. For each model, we use the best set of input features which yields the lowest RMSE according to Table 3. We find that the DM results are greater than

zero, implying that the predicted errors obtained through the different models are significantly different and the network structure-based method is significantly better than the compared methods. In Figure 6, we display the best predictions obtained by the optimal $K$ for the 9 topics. The name of each topic's neighbors is provided in Table 5.

## 8. Discussion

Through extensive experiments, we have validated the efficiency of topic interaction for news topic prediction. In practice, the dynamic topic-network as well as the proposed forecasting model could be implemented as a visualization tool to provide real-time feedback of news topic trends for the social media industry. We are also aware of the limitations of this work. On the one hand, unpredictable and unexpected "black swan" news will inevitably happen with surprise and unexpectedness (e.g., COVID-19). Without any prior knowledge, it is almost impossible to forecast such breaking stories by universal patterns. Therefore, in this work we focus on predicting general topics (e.g. job, disease, economy) rather than particular news (e.g. COVID-19), given that these topics will not disappear or suddenly appear, but only increase or decrease in volume over time. On the other hand, the proposed method involves manual labelling to check the clustering results to ensure the topic's consistency, which may cause additional complexity in practice.

## 9. Conclusion & Future work

In summary, an effective forecasting framework has been introduced to model and predict the popularity of tomorrow's topics on both Twitter and a news feed platform. We identified a meaningful list of topics by analyzing words and studied the underlying interaction among topics in terms of network structure as well as the evolution of content over time. A combination of these features provides higher predictive power than solely using past popularity information. Using an LSTM architecture, our experimental results demonstrate that the neighboring topics of a topic in the historical networks are helpful to boost prediction accuracy, outperforming other existing approaches.

Our results contribute to advancing current research in the field of news forecasting by providing new insights on how to extract novel and powerful data features from topic relationship. The proposed framework identifies topics from text and integrates multiple approaches for analyzing it in better ways. Our findings have practical implications for intelligence organizations and news information providers. Automatically mining, in principle, all kinds of social networking data and news platforms, our system could be used for the prediction of tomorrow's news and the verification of today's news, for instance to flag potential fake news.

For our proposed framework, there remain some areas for future improvements. The topic-network, as a representation of complex relationships among topics, contains more structural information beyond nodal centrality used in this work. Integrating more network-based information as features might be a way of increasing the accuracy we have achieved. Besides, finer-grained feature-selection methods could be developed to make the framework more robust. An example is to introduce weights like the time decay rate when choosing historical neighbors. Nevertheless, we have shown that the time is finally coming where we can predict what will be happening in the future, if not for thousands of years, as Asimov envisions, but at least for tomorrow.

# Appendix A

**Table 6** Generated Topics ($N = 38$) from Words for *Fatherlanders* Dataset

| No. | Topic | Top-5 words by frequency | No. | Topic | Top-5 words by frequency |
|---|---|---|---|---|---|
| 1 | **human body** | hand, head, hands, cut, eyes | 20 | **top politicians** | Trump, Pelosi, Hillary, Schumer, chairman |
| 2 | **school** | law, history, school, book, college | 21 | **patriots** | patriot, CIA, patrol, spy, army |
| 3 | **news** | media, news, report, CNN, fox | 22 | **family** | children, woman, child, kids, babies |
| 4 | **job** | job, security, office, administration, chief | 23 | **money** | money, million, tax, pay, billion |
| 5 | **extreme politicians** | Schiff, Mueller, Ishan, Epstein, Barr | 24 | **police** | FBI, police, killed, arrested, murder |
| 6 | **catering** | food, served, turkey, eat, healthy | 25 | **country** | America, country, Russia, China, Mexico |
| 7 | **economy** | economy, crisis, climate, shutdown, funding | 26 | **US states** | state, California, Texas, Washington, York |
| 8 | **action movies** | breaking, story, war, hero, Hollywood | 27 | **American** | Americans, citizens, illegals, politicians, immigrants |
| 9 | **marketing** | free, deal, order, sign, save | 28 | **disease** | illegal, abuse, health, drug, cancer |
| 10 | **arts** | wall, white, black, case, clear | 29 | **sports** | patriots, win, won, nation, team |
| 11 | **politics** | president, democrats, Obama, democrat, Biden | 30 | **drinks** | ice, water, oil, filled, lemon |
| 12 | **ideology** | leftist, communist, partisan, phony, treasonous | 31 | **border** | border, Ukraine, Iran, military, attack |
| 13 | **ISIS** | Nancy, ISIS, Elizabeth, anna, MS | 32 | **nature** | liberty, hill, historic, haven, sanctuary |
| 14 | **technology** | power, control, speaker, car, energy | 33 | **God** | God, truth, bless, freedom, lies |
| 15 | **clothes** | wearing, hat, short, wear, ties | 34 | **nation** | American, women, national, foreign, Russian |
| 16 | **towns** | Alexandria, Lakeland, Durham, Covington, capitol | 35 | **court** | collusion, infanticide, FISA, obstruction, Baghdadi |
| 17 | **creature** | aliens, swamp, hunt, dog, alien | 36 | **populist politician** | Bernie, warren, Flynn, sanders, Harris |
| 18 | **community** | rally, members, supporters, leaders, leadership | 37 | **emergency** | emergency, reported, alert, warning, hurricane |
| 19 | **tribalism** | socialism, radical, borders, conspiracy, racism | 38 | **justice** | impeachment, justice, court whistleblower, evidence |

**Table 7** Generated Topics ($N = 39$) from Words for *HuffPost News* Dataset

| No. | Topic | Top-5 words by frequency | No. | Topic | Top-5 words by frequency |
|---|---|---|---|---|---|
| 1 | **disaster** | climate, crisis, issues, sandy, threat | 21 | **nation** | America, country, national, Russia, international |
| 2 | **US states** | state, north, California, Washington, Texas | 22 | **IT** | media, social, using, online, Internet |
| 3 | **money** | money, million, tax, pay, budget | 23 | **job** | public, business, job, company, group |
| 4 | **online platform** | Twitter, face, gay, Facebook, ask | 24 | **learning** | tips, ideas, guide, parenting, research |
| 5 | **places** | city, York, street, Paris, hotel | 25 | **LGBT** | LGBT, LGBTQ, transgender, queer, racist |
| 6 | **politician** | Donald, sanders, James, Bernie, Ryan | 26 | **travel** | travel, trip, road, flight, near |
| 7 | **dating** | photo, sex, sexual, hot, model | 27 | **sport** | season, game, united, play, win |
| 8 | **food** | food, eat, eating, recipes, dinner | 28 | **nature** | light, green, earth, dark, nature |
| 9 | **president** | president, Obama, Clinton, Americans, Hillary | 29 | **home** | house, room, wall, outside, clean |
| 10 | **war** | war, violence, attack, Muslim, military | 30 | **singer** | choice, awards, lady, Taylor, queen |
| 11 | **drinking** | water, ice, wine, fresh, coffee | 31 | **shopping** | order, gift, buy, store, shopping |
| 12 | **police** | police, court, shooting, killed, accused | 32 | **government** | government, federal, administration, supreme |
| 13 | **disease** | cancer, stress, mental, drug, disease | 33 | **ISIS** | Kate, ISIS, Michelle, Kelly, Elizabeth |
| 14 | **economy** | global, energy, industry, financial, companies | 34 | **animals** | dog, wild, dogs, homeless, animal |
| 15 | **music** | video, style, live, music, cover | 35 | **dress** | women, white, black, hair, red |
| 16 | **emotion** | experience, fear, success, opportunity, peace | 36 | **entertainment** | star, story, film, TV, movie, book |
| 17 | **society** | political, culture, human, community, education | 37 | **health** | health, healthy, weight, loss, yoga |
| 18 | **fashion & arts** | fashion, wedding, art, beauty, inspired | 38 | **actors** | actress, Kardashian, Jenner, Oscars, Jennifer |
| 19 | **Olympic** | celebrate, members, honor, Olympic, leading | 39 | **school** | school, history, college, study, students |
| 20 | **human body** | body, head, inside, hand, eyes | | | |

## Acknowledgments

The authors would like to thank the two anonymous referees for the constructive suggestions. This work has been supported in part by the China Scholarship Council Program, under grant No. 201906380135.

## References

Abbar S, Castillo C, Sanfilippo A (2018). To post or not to post: Using online trends to predict popularity of offline content. *Proceedings of the 29th on Hypertext and Social Media*: 215-219.

Ahmed M, Spagna S, Huici F, Niccolini S (2013). A peek into the future: Predicting the evolution of popularity in user generated content. *Proceedings of the 6th ACM International Conference on Web Search and Data Mining*: 607-616.

Antonacci G, Colladon A F, Stefanini A, Gloor P (2017). It is rotating leaders who build the swarm: Social network determinants of growth for healthcare virtual communities of practice. *Journal of Knowledge Management* 21(5): 1218-1239.

Aral S, Van Alstyne M (2011). The diversity-bandwidth trade-off. *American Journal of Sociology* 117(1): 90-171.

Arthur D, Vassilvitskii S (2006). k-means++: The advantages of careful seeding. *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*: 1027-1035.

Baccouche M, Mamalet F, Wolf C, Garcia C, Baskurt A. (2011). Sequential deep learning for human action recognition. *International Workshop on Human Behavior Understanding*: 29-39. Springer.

Bandari R, Asur S, Huberman B A (2012). The pulse of news in social media: Forecasting popularity. *Sixth International AAAI Conference on Weblogs and Social Media*.

Blei DM, Ng AY, Jordan MI (2003). Latent dirichlet allocation. *Journal of Machine Learning Research* 3: 993-1022.

Cawkell AE (1971). Science citation index. Effectiveness in locating articles in the anaesthetics field:" perturbation of ion transport". *British Journal of Anaesthesia* 43(8): 814.

Cova B, Cova V (2002). Tribal marketing. *European Journal of Marketing* 36(5): 595-620.

Amorim RC, Hennig C (2015). Recovering the number of clusters in data sets with noise features using feature rescaling factors. *Information Sciences* 324: 126-145.

Choudhury M, Sundaram H, John A, Seligmann DD (2008). Can blog communication dynamics be correlated with stock market activity? *Proceedings of the 19th ACM Conference on Hypertext and Hypermedia*: 55-60.

Diebold FX, Mariano RS (2002). Comparing predictive accuracy. *Journal of Business and Economic Statistics* 20(1): 134-144.

Ebrahimi M, Yazdavar AH, Sheth A (2017). Challenges of sentiment analysis for dynamic events. *IEEE Intelligent Systems* 32(5): 70-75.

Freeman LC (1977). A set of measures of centrality based on betweenness. *Sociometry*: 35-41.

Gilbert CHE, Eric H (2014). Vader: A parsimonious rule-based model for sentiment analysis of social media text. *8th International Conference on Weblogs and Social Media*: 82-91.

Gloor PA (2017). *Sociometrics and Human Relationships*. Emerald Publishing Limited.

Gloor PA, Colladon AF, de Oliveira JM, Rovelli P, Galbier M, Vogel M (2019). Identifying tribes on twitter through shared context. *Collaborative Innovation Networks*: 91-111.

Gloor P, Colladon AF, de Oliveira JM, Rovelli P (2020). Put your money where your mouth is: Using deep learning to identify consumer tribes from word usage. *International Journal of Information Management* 51: 101924.

Graves A, Mohamed AR, Hinton G (2013). Speech recognition with deep recurrent neural networks. *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*: 6645-6649. IEEE.

Gregor K, Danihelka I, Graves A, Rezende D, Wierstra D (2015). DRAW: A Recurrent Neural Network for Image Generation. *International Conference on Machine Learning*: 1462-1471.

Gruhl D, Guha R, Kumar R, Novak J, Tomkins A (2005). The predictive power of online chatter. *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*: 78-87.

Gupta RK, Yang Y (2019). Predicting and understanding news social popularity with emotional salience features. *Proceedings of the 27th ACM International Conference on Multimedia*: 139-147.

Harvey D, Leybourne S, Newbold P (1997). Testing the equality of prediction mean squared errors. *International Journal of Forecasting* 13(2): 281-291.

Hochreiter S, Schmidhuber J (1997). Long short-term memory. *Neural Computation* 9(8): 1735-1780.

Hummon NP, Dereian P (1989). Connectivity in a citation network: The development of DNA theory. *Social Networks* 11(1): 39-63.

Kidane YH, Gloor PA (2007). Correlating temporal communication patterns of the Eclipse open source community with performance and creativity. *Computational and Mathematical Organization Theory* 13(1): 17-27.

Kim SD, Kim SH, Cho HG (2011). Predicting the virtual temperature of web-blog articles as a measurement tool for online popularity. *2011 IEEE 11th International Conference on Computer and Information Technology*: 449-454. IEEE.

Kleeb R, Gloor PA, Nemoto K, Henninger M (2012). Wikimaps: dynamic maps of knowledge. *International Journal of Organisational Design and Engineering* 2(2): 204-224.

Krauss J, Nann S, Simon D, Gloor PA, Fischbach K (2008). Predicting Movie Success and Academy Awards through Sentiment and Social Network Analysis. *16th European Conference on Information Systems*: 2026-2037.

Misra Rishabh. News Category Dataset (2018). ResearchGate.

Newman M (2018). *Networks*. Oxford University Press.

Pennington J, Socher R, Manning CD (2014). Glove: Global vectors for word representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language processing*: 1532-1543.

Pinto H, Almeida JM, Goncalves MA (2013). Using early view patterns to predict the popularity of YouTube videos. *Proceedings of the 6th ACM International Conference on Web Search and Data Mining*: 365-374.

Rousseeuw PJ (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* 20: 53-65.

Ruan Y, Purohit H, Fuhry D, Parthasarathy S, Sheth AP (2012). Prediction of Topic Volume on Twitter. *Proceedings of the 4th International ACM Conference on Web Science*: 397-402.

Szabo G, Huberman BA (2010). Predicting the popularity of online content. *Communications of the ACM* 53(8): 80-88.

Tatar A, Antoniadis P, De Amorim MD, Fdida S (2012). Ranking news articles based on popularity prediction. *2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*: 106-110. IEEE.

Tatar A, Leguay J, Antoniadis P, Limbourg A, De Amorim MD, Fdida S (2011). Predicting the popularity of online articles based on user comments. *Proceedings of the International Conference on Web Intelligence, Mining and Semantics*: 1-8.

Weng L, Menczer F, Ahn YY (2014). Predicting successful memes using network and community structure. *8th International AAAI Conference on Weblogs and Social Media*.

Zhang X, Fuehres H, Gloor PA (2011). Predicting stock market indicators through twitter "I hope it is not as bad as I fear". *Procedia-Social and Behavioural Sciences* 26: 55-62.

**Jiachen Sun** is a PhD student at Sun Yat-sen University, China and a visiting graduate student at the MIT Center for Collective Intelligence. His research interests are in complex networks, data science and information theory.

**Peter Gloor** is a research scientist at the MIT Center for Collective Intelligence. He completed his PhD in computer science at University of Zurich, Switzerland. He conducts research on swarm creativity, collaborative innovation networks, and predictive analytics. He is also Founder and Chief Creative Officer of software company galaxyadvisors where he puts his academic insights to practical use, helping clients to coolhunt by analyzing social networking patterns on the Internet.