International Journal of Data Science and Analytics (2024) 18:369–378
https://doi.org/10.1007/s41060-023-00424-6

REGULAR PAPER

Modeling the time to share fake and real news in online social networks

Cooper Doe[1] · Vladimir Knezevic[2] · Maya Zeng[3] · Francesca Spezzano[4] · Liljana Babinkostova[4]

Received: 3 April 2023 / Accepted: 21 June 2023 / Published online: 12 July 2023

Abstract
In this paper, we address the problem of predicting the time to share (real or fake) news items on online social media. Specifically, given the scenario where a user u is influenced on some given (real or fake) news item n by at least one of the people they follow v, we predict when the user u will re-share the news item n among their followers. We model the problem as a survival analysis task, which is a statistical analysis method aimed at predicting the time to event (the re-sharing event in our case). Survival analysis differs from other methods such as regression in that it also considers the data where the event (sharing) never occurs (censored data) in the considered time window. We considered Twitter data containing information on real and fake news shares to test our proposed survival analysis approach and modeled different aspects of the problem including user, news, and network characteristics. We show the superiority of survival analysis as compared to regression to model this problem in both the cases of real and fake news sharing.

Keywords Misinformation · Survival analysis · Time-to-event prediction

1 Introduction

The spread of online misinformation[1] is one of the largest threats we face every day. People regularly consume news from online platforms,[2] and many are unable to assess its veracity correctly [2]. Researchers have modeled different aspects of the fake news sharing process on different social media platforms, including defining diffusion models for fake news spreading [3] and profiling and detecting fake news spreaders [4]. Being able to understand better and model how misinformation spreads can help simulate its diffusion and reasoning among possible interventions.

However, current research lacks models able to compute *when* a user will share a piece of news they have received

✉ Francesca Spezzano
francescaspezzano@boisestate.edu

Cooper Doe
rc_doe@coloradocollege.edu

Vladimir Knezevic
vknezevi@mail.ccsf.edu

Maya Zeng
maya.zeng.259@my.csun.edu

Liljana Babinkostova
liljanababinkostova@boisestate.edu

1 Colorado College, Colorado Springs, USA

2 City College of San Francisco, San Francisco, USA

3 California State University, Northridge, USA

4 Boise State University, Boise, USA

[1] Misinformation is incorrect or misleading information that can be shared accidentally, causing different levels of harm. Misinformation comes in many forms, such as satire, propaganda, hoax, rumor, conspiracy, and fake news. This latter, in particular, explicitly implies false or misleading information spread deliberately to deceive and is one form of misinformation [1]. Although they can have slightly different connotations in specific contexts, misinformation and fake news are often used interchangeably in popular discourse. In this manuscript, we use the terms synonymously to refer to the dissemination of false or misleading news.

[2] https://www.pewresearch.org/journalism/2021/09/20/news-consumption-across-social-media-in-2021/

from another user they follow. Adding this time component to existing models is important to model misinformation diffusion better, as not everyone shares information as soon as it is received. Some people do it immediately, some are exposed to it for a while before taking any action, while others will never re-share (skeptic users) [5]. Predicting the time to share can inform the diffusion model on the time step when a node will be infected, which not necessarily is the time step right after a neighbor got infected (as assumed by classical information diffusion models [6, 7]). Creating better news diffusion models contributes to simulating news spreading more realistically which helps with reasoning better on the efficacy of possible interventions. Also, modeling time to share, especially at the user level, can be helpful to prioritize better who might need an intervention quickly.

Thus, in this paper, we address for the first time the problem of predicting the time to re-share real and fake news on online social networks. We consider different aspects of the problem, including modeling user, news, and user network characteristics, as they are all factors already shown to be relevant to model fake news sharing [8]. We use Twitter data, which is more readily available than other social media platforms, and model the problem via survival analysis.

Survival analysis is a statistical method to estimate the expected duration of time until an *event of interest* occurs [9]. In our case, the event of interest is when a user re-shares a piece of real or fake news received by a user they follow. Survival analysis is better suited than regression models for this problem as it allows to effectively work with censored information in input, i.e., the fact that the event did not occur in the considered time window and may or may not have occurred after the end of the time window.

Our experimental results on two Twitter datasets, one containing sharing and non-sharing instances of fake news items and the other containing sharing and non-sharing instances of real news items, show the superiority of survival analysis approaches compared to regression models to address the problem. Specifically, we show that the best of the considered survival analysis models can predict the time to re-share a news item with a normalized RMSE (NRMSE) of 0.43 and a concordance index (C-Index) of 0.84 for fake news (vs. an NRMSE of 0.50 and a C-Index of 0.83 achieved by the best regression model) and an NRMSE of 0.46 and a C-Index of 0.83 for real news (vs. an NRMSE of 0.52 and a comparable C-Index of 0.83 achieved by the best regression model). We also show that the considered features differ in importance in predicting the time to share real vs. fake news.

The manuscript is organized as follows. Section 2 summarizes related work, Sect. 3 describes the dataset we used in this work, Sect. 4 presents our proposed survival analysis-based framework to predict the time to re-share fake and real news in online social networks, Sect. 5 reports our experimental evaluations, and, finally, conclusions are drawn in Sect. 6.

2 Related work

In studying information diffusion, many existing studies have focused on modeling fake news propagation via epidemiological models [3], where the problem is addressed similarly to the diffusion of an infectious disease (epidemiological models) or as a Hawkes process [10]. In both cases, these models assume an implicit network (unknown connections among individuals). This makes these models more suitable to study global patterns, such as trends and ratios of people sharing stories of a given topic, but not the local node-to-node diffusion patterns.

However, in case there is the need to work with an explicit network and compute who is infected (i.e., shares fake news) and by whom is infected, classical models such as the Independent Cascade and the Linear Threshold models can be used to model misinformation spread [6]. However, some works went beyond just considering the network to explain news diffusion and tested hypotheses inspired by the Diffusion of Innovation Theory, which also finds user and news characteristics as essential factors to explain news sharing behavior [11]. Ma et al. [12] found opinion leadership, news preference, and tie strength to be the most important factors in predicting news sharing, while homophily hampered news sharing in users' local networks. Also, people driven by gratifications of information-seeking, socializing, and status-seeking were more likely to share news on social media platforms [13]. Joy et al. [8] showed how combining user, news, and network characteristics lead to better predictions of people sharing or not a piece of news than models that are only based on network information such as the Independent Cascade and the Linear Threshold models. The importance of using user characteristics in fake news spreading behavior has also been highlighted in work addressing the problem of profiling fake news spreaders. For instance, Vosoughi et al. [14] found out that fake news spreaders had, on average, significantly fewer followers, followed significantly fewer people, and were significantly less active on Twitter. Concerning the social media platform Twitter, although bots contribute to spreading fake news by injecting fake news into social media [15], the dissemination of fake news on Twitter is mainly caused by human activity.

Shu et al. [16] analyzed user profiles to identify the characteristics of users who are likely to trust or distrust fake news. They found that, on average, users who share fake news tend to be registered for a shorter time than the ones who share real news. Additionally, while bots were shown to be more likely to post a piece of fake news than real news, users who spread fake news are still more likely to be humans than bots. They also show that real news spreaders are more likely to be more popular than fake news spreaders and that older people and females are more likely to spread fake news.

Table 1 Size of the PolitiFact dataset

# Users	# News	# Tweets	# Retweets	# Real news	# Fake news
281,596	992	438,504	619,239	560	432

Guess et al. [17] also analyzed user demographics as predictors of fake news sharing on Facebook and found that political orientation, age, and social media usage be the most relevant. Additionally, researchers theorized that senior citizens tend to share more fake news because they may have a lower digital media literacy, and that people who post more news on social media are less likely to share fake news because those users are more familiar with the platform's features.

The author profiling shared task at the PAN 2020 conference focused on determining whether or not the author of a Twitter feed was keen to spread fake news [4]. Participants proposed different linguistic features to address the problem, including (a) n-grams, (b) style, (c) personality and emotions, and (d) tweet text embeddings. Shrestha and Spezzano [18] showed that a combination of user demographics, writing style, personality traits, emotions, and behavioral and network features strongly predict fake news spreaders, outperforming the best models proposed at the PAN 2020 fake news spreaders profiling shared task.

However, current research on fake news spreading models is limited in predicting when a user will re-share a received piece of news. In fact, not everyone takes the same time. For example, some users may share it as soon as it is received, others may take some time to analyze the received news item (exposed), while others may never re-share the received news (skeptic) [5]. In this paper, we introduce a user-news-dependent sharing time component that, coupled with news diffusion models working with an explicit network, can make the simulation of news propagation more realistic.

3 Dataset

We used the PolitiFact dataset from FakeNewsNet to carry out our experiments. FakeNewsNet [19] is a well-known data repository comprising two datasets, PolitiFact and GossipCop, from two different domains, i.e., politics and entertainment gossip, respectively. Each dataset contains details about news content, publisher, social engagement information, and social network. In this paper, we only used the PolitiFact dataset (as gossip is different from fake news), which contains news with known ground truth labels collected from the fact-checking website PolitiFact[3] where journalists and domain experts fact-checked the news items as fake or real. The size of this dataset is shown in Table 1.

[3] https://www.politifact.com/

In order to test our proposed method to predict time to share, we first computed the labels for user sharing or not sharing a given piece of news, as follows. First, we computed pairs of influencer and influenced users. An *influencer* is a user who tweeted a given piece of news, and an *influenced user* is a follower of that influencer. We considered (a) users (influencers) who have shared at least one piece of news and have at least one follower (influenced user) and (b) followers who shared at least five instances of news (and ordered those instances in chronological order of shared time). Hence, we annotated news items as shared or not as follows:

- Shared a piece of news that is shared/tweeted by a user (influencer) and then shared/retweeted by its follower (influenced user) as one of their two most recently shared news items.
- Not Shared a piece of news that is shared/tweeted by a user (influencer) but never shared/retweeted by their follower.

Next, for each instance of sharing, we computed the time elapsed between the influencer and influenced user sharing, and used this time as the variable to predict. The cases where the influenced user did not share the news item are considered censored. Also, since the news sharing after the first 48 h is sporadic, random, and with no effect on viral spread, we limited our instances of shared news items to less than 48 h and counted the rest as non-shared (or censored). Furthermore, we divided the time window into 48 time intervals of one hour each.

We used the remaining three or more news articles shared by followers to profile these users and compute the user's interest similarity with the given news item, as discussed in 4.2.1. In addition, for each follower, we crawled all tweets posted one month before the shared news article's publish date to compute the remaining user-based features.

We have a total of 1,572 influenced users in our dataset sharing 127 real and 169 fake news items, and 18,080 sharing and non-sharing instances. To compare real and fake news sharing dynamics, we consider two different datasets, one containing only sharing/non-sharing instances of real news items and the other containing only sharing/non-sharing instances of fake news items (cf. Table 2).

Table 2 Size of the datasets used in our experiments

Dataset	# Users	# News	# Shared	# Not shared
Fake news sharing	1,557	169	465	7,196
Real news sharing	1,572	127	2,409	8,010

4 Methodology

In this paper, we address for the first time the following problem: *given that a user u is influenced on some given (real or fake) news item n by at least one of their followees v (i.e., u is following v and v has shared some news item n among their followers), predict when the user u will also share news item n among their followers.* It is worth noting that the fact that a follower re-shares a news article based on shares made by an influencer is a reasonable and realistic assumption according to how re-sharing of information works on social media platforms such as Twitter, i.e., it is enough that a single neighbor has shared a piece of content for it to potentially be seen and retweeted by its followers [20–22].

We model the problem mentioned above as a survival analysis task. Survival analysis is a statistical method to estimate the expected duration of time until an *event of interest* occurs [9]. The benefit of using survival analysis over other techniques for modeling the problem of predicting the time to share real and fake news on online social networks is the fact that, in this scenario, censored instances can occur. *Censored* instances are instances (news articles in our case) for which the event (re-share of the article) did not occur within the study duration or for which we may have incomplete or missing, or unavailable data about the event occurrence. This occurs because we are observing the problem in the limited time window or we miss the traces of some instances. Data including censored instances can be effectively approached using survival analysis. Other modeling methods such as, for instance, regression are not able to handle censored instances.

The time when the event of interest (time to share a news item by an influenced user) happens for the instance $i = (v, u, n)$ is denoted by T_i. If the event of interest is not observed for an instance i, we say that i is censored and denote by C_i the censored time. In our case we have *right-censored* instances, which means that the censored time is the end of the considered time window. Given an instance i, we denote by X_i the feature vector. Let δ_i be a Boolean variable indicating whether or not the instance i is not censored, i.e., if $\delta_i = 1$ then the instance i is not censored. We denote by y_i the observed time for the instance i that is equal to T_i if i is uncensored and C_i otherwise, i.e.,

$$y_i = \begin{cases} T_i & \text{if } \delta_i = 1 \\ C_i & \text{if } \delta_i = 0 \end{cases}$$

Given a new instance j described by the feature vector X_j, survival analysis estimates a *survival function* S_j that gives the probability that the event for the instance j will occur after time t, i.e., $S_j(t) = Pr(T_j \geq t)$. Hence, the expected time $\hat{y_j}$ of event occurrence for an instance j can be computed by integrating the survival function.

In survival analysis, another commonly used function is the hazard function $h(t)$, also called the instantaneous death rate or the conditional failure rate, and indicates the rate of event at time t given that no event occurred before time t. Mathematically, the hazard functions can be computed by the survival function (and vice versa) as $h(t) = f(t)/S(t)$, where $f(t) = -\frac{d}{dt}S(t)$ is the death density [9].

4.1 Survival analysis models

There are different types of statistical methods to estimate the survival function. In this paper, we consider machine-learning methods such as Random Forest Survival (RSF) and Gradient Boosting Survival (GBS) and Multi-Task Logistic Regression (MTLR) models such as the Linear MTLR and the Neural MTLR, and semi-parametric methods such as the Penalized Cox Proportional Hazards (Cox PH) model where the knowledge of the underlying distribution of survival times is not required [9].

4.1.1 Penalized cox proportional hazards model

The Penalized Cox Proportional Hazards Model estimates the survival function for a population by assuming that the hazard functions across time are proportional. The basic Cox Proportional Hazards Model computes the hazard function (which is the instantaneous rate of event occurrence) by

$$h(t|X_i) = h_0(t)\exp(X_i \cdot \beta)$$

where X_i is the vector of covariates (or feature vector), and β are the various impact weights of the covariates. The sksurv python package implements the Cox Proportional Hazards Model with an Elastic Net penalty, which allows for features to be correlated (as many of our features for the considered datasets are). The Elastic Net penalty uses a weighted combination of two penalty choices and selects a certain subset of all highly correlated features for the penalty terms.

4.1.2 Gradient boosting survival model

Gradient Boosting is not a specific model but rather a learning framework through which several survival models can be optimized. Gradient Boosting optimizes a specified loss function; here, we use the Cox Proportional Hazards Model to take advantage of the sksurv *predict_survival_function* capa-

bility. Gradient Boosting combines the predictions of many base learners (CoxPH models).

4.1.3 Random survival forest model

Random Survival Forest builds survival trees from bootstrap samples of the original training data, unlike Gradient Boosting (which uses the entire training set for each base learner). Predictions for the survival function are formed by combining the predictions of each survival tree in the overall ensemble.

4.1.4 Multi-task logistic regression models

The Multi-Task Logistic Regression (MTLR) Model [23] provides an alternative to the Cox PH model as this model has some drawbacks [24], including (i) it relies on the proportional hazard assumption, which specifies that the ratio of the hazards for any two in the exact formula of the model that can handle ties is not computationally efficient, and is often rewritten using approximations, such as the Efron's or Breslow's approximations, in order to fit the model in a reasonable time; (iii) the fact that the time component of the hazard function remains unspecified makes the CoxPH model ill-suited for actual survival function predictions.

Linear Multi-Task Logistic Regression Linear Multi-Task Logistic Regression (LMTR) is a series of logistic regression models built on different time intervals to estimate the probability that the event of interest happened within each interval [24].

Neural Multi-Task Logistic Regression Neural Multi-Task Logistic Regression (NMTLR) is a type of Multi-Task Logistic Regression Model that uses a Neural Network to resolve the performance of the prediction models when there are nonlinear elements in the data [24].

4.1.5 Deep learning models

Survival analysis models based on deep learning use artificial neural networks to learn the hazard function or the distribution of survival times and alleviate the problems of making specific assumptions regarding these functions, e.g., assuming linearity.

DeepSurv DeepSurv is a Cox Proportional Hazard Model that leverages deep neural network to alleviate the Cox Proportional Hazard assumption that risk is linear [25]. Specifically, this method uses a deep feed-forward neural network to predict the log-risk function $h(t)$ parameterized by the weights of the neural network.

DeepHit DeepHit uses deep neural networks to learn the distribution of survival times directly [26]. This model is meant to alleviate the reliance on often violated, strong parametric assumptions. It makes no assumptions about the underlying stochastic process, allowing for the possibility of a relationship between covariates and risk(s) that changes over time.

4.2 Features

As we report in Sect. 2, the current literature highlights several features as important predictors for distinguishing fake news spreaders. These include demographics (age, gender, and political ideology) [17], emotions and sentiment of the news article or expressed by users in their timeline before sharing fake news [18, 27, 28], user interest in the news [29], user Twitter explicit features and behavior [16, 18], user centrality in the follower–followee network [14] as well as weak and strong ties [12, 30]. Hence, we have considered all of these factors when defining features to be used in input to our proposed survival analysis framework to predict the time to re-share a news article. We group them into user-based, news-based, and network-based features and describe how these features are computed in the following.

4.2.1 User-based features

Demographics This information is often not publicly available on social media; thus, we used M3inference, a deep neural architecture [31] trained on Twitter data, to infer the users' age and gender. Predicted *gender* is classified as male or female, while *age* is grouped in these categories: ($\leq$18, 19−29, 30−39, $\geq$40). To infer users' political leaning (left or right), we used the #Polar Score algorithm [32], which leverages the polarity of hashtags used by the users in their tweets, and we trained it on the U.S. Congress member tweet dataset by Chamberlain et al. [33].

Explicit Features and Activity These features include the number of tweets and retweets by the user (status count), the number of liked tweets (favor count), tweets being protected or not (protected), account verification (verified), and the length of registration of the account (registered). Additionally, we analyzed the user tweeting behavior within the day (24 h) by computing the user *insomnia index*, i.e., the difference between the number of night and day posts upon the total number of user posts.

Emotion Users' emotions are extracted from their tweets. This information is captured by concatenating all the user's tweets together and then calculating every single emotion using Emotion Intensity Index (NRC-EIL) [34]. VADER [35] is used to calculate the sentiment analysis, and it measures the average sentiment across all the tweets (positive, negative, neutral). Finally, stress was included using the lexical dictionary created by Wang et al. [36] for LIWC.

User Interest in News Two similarity measures are used to compute the user's interest in a given news item. *Similarity*

news is the cosine similarity between the topics extracted from the shared news item and the three or more previously shared news items. *Similarity tweets* is the cosine similarity between the topic extracted from the given news item and all the tweets the user authored. We used an LDA model [37] with 100 topics and trained on Wikipedia articles. This model was utilized for retrieving the topical similarity between the users' interests and shared news.

4.2.2 Network-based features

User Centrality We considered the number of followers (in-degree), the number of followees (out-degree), the PageRank of the user, and the ratio of followers to followees (TFF) [16] calculated as $TFF = \frac{\#Follower+1}{\#Followee+1}$ (the greatness of this ratio, the higher the user's popularity).

Weak and Strong ties News sharing intention is correlated with the tie strength between the influencer and the influenced user in the social network [12, 30]. We used two approaches to compute tie strength: *receiver's perspective*—the percentage of user's tweets that are retweets of the influencer's posts, and *time-based tie strength*—the average time the user would take to share the influencer's tweet.

4.2.3 News-based features

These features help capture the various aspects (stylistic, psychological, and complexity) of the news items users are exposed to. These features have been used in [27] for fake news classification.

Stylistic Features These features capture the writing style of a user and include word count (WC), words per sentence (WPS), number of personal and impersonal pronouns, number of exclamation marks, punctuation symbols, and quotes. We computed these features by using the LIWC tool [38] and the Python Natural Language Toolkit part of speech tagger.

Psychology Features The emotion Intensity Lexicon (NRC-EIL) was used to calculate news emotion features: anger, joy, sadness, fear, disgust, anticipation, surprise, and trust [34]. The LIWC tool was used for computing the positive (pos) and negative (neg) sentiment metrics.

Complexity Features Simple Measure of Gobbledygook Index (SMOG) measures readability as a complexity feature (ease of reading and understanding text). The higher the score, the easier it is to read the text. Besides this, we also computed the average length of each word (avg word length) and the Type-Token Ratio (TTR, lexical diversity).

5 Experiments

In this section, we compare survival analysis against regression models on the task of predicting when a user will re-share a received (real or fake) news item.

5.1 Experimental setup

We considered a time window of 48 h since the news item is shared by the influencer and divided the time into intervals of 1 h each. If the influenced user did not re-share the news item within 48 h, we consider that instance censored.

We used the implementation of the survival analysis models described in Sect. 4.1 provided by the Python packages sksurv [39], PySurvival [40], and pycox [41].

For regression models, we considered Linear Regression [42], Ridge regression [43], Lasso regression [44], Random Forest regression [45] and Multi-layer Perceptron (MLP) regression [46] as implemented in the sklearn [47] Python package. For a fair comparison, we considered the same set of features presented in Sect. 4.2 in input to the regression task. In addition, since regression cannot deal with censored instances, we approximated the occurrence time of censored instances with a time value much larger than the end of the considered time window, in our case we chose $200 \gg 48$. This is a typical setting when comparing regression with survival analysis [48]. It is worth noting that the news article features and user features such as user demographic, personality, and others are not temporal; hence, we did not compare with other regression methods such as AutoRegressive Integrated Moving Average (ARIMA) [49], Long Short-Term Memory (LSTM) [50], etc.

We considered the C-Index [9], a commonly used metric to evaluate survival analysis models, and the Normalized RMSE (NRMSE) to compare the performances of these two classes of algorithms.

The C-Index is defined as

$$\text{C-Index} = \frac{1}{N_{\text{cnt}}} \sum_{i:C_i=1} \sum_{y_j > y_i} 1(\hat{y_j} > \hat{y_i})$$

where N_{cnt} is the number of all pairs (y_i, y_j) such that $C_i = 1$ (non-censored instances) and it holds that $y_j > y_i$, $\hat{y_i}$ is the estimated duration predicted by the model. The C-Index measures the concordance probability between the actual observation times and the predicted values.

The Normalized RMSE (NRMSE) is computed by dividing the RMSE by the difference between the maximum ($y_{\max}$) and minimum values (y_{min}) of the observed times:

$$\text{NRMSE} = \frac{1}{(y_{\max} - y_{\min})} \sqrt{\frac{\sum_{i=1}^{N} (y_i - \hat{y}_i)^2}{N}}$$

Table 3 Experimental results on the **real news** dataset

Model	NRMSE	C-Index
Survival analysis models		
Cox PH	0.75377	0.52316
Gradient Boosting Survival	0.48354	0.8276
Random Survival Forest	**0.46178**	**0.8292**
Neural MTLR	1.75419	0.80569
Linear MTLR	1.56596	0.82788
DeepSurv	0.66844	0.77134
DeepHit	1.06052	0.78078
Regression models		
Linear regression	2.53825	0.78452
Lasso regression	2.16395	0.77823
Ridge regression	2.57118	0.78697
Random forest regression	0.50813	0.82733
Multi-layer Perceptron (MLP) Regression	5.40171	0.74841

Best results are in bold

Table 4 Experimental results on the fake news dataset

Model	NRMSE	C-Index
Survival analysis models		
Cox PH	0.75118	0.6867
Gradient Boosting Survival	**0.43352**	0.839
Random Survival Forest	0.47004	0.823
Neural MTLR	1.70562	0.78083
Linear MTLR	1.40477	**0.85764**
DeepSurv	2.96060	0.72086
DeepHit	0.98162	0.52258
Regression models		
Linear regression	3.71823	0.7841
Lasso regression	0.69516	0.78227
Ridge regression	0.71818	0.78139
Random forest regression	0.49897	0.8311
Multi-layer perceptron (MLP) regression	0.55257	0.83267

Best results are in bold

A higher C-Index and lower NRMSE are desirable.

Furthermore, we performed five-fold cross-validation and majority under-sample to deal with class imbalance.

5.2 Experimental results

Tables 3 and 4 show survival analysis models' performances compared to regression models on the real news dataset and the fake news dataset, respectively.

Table 3 shows that Random Survival Forest is the best at predicting time to share for real news among all the considered survival analysis models, achieving an NRMSE of 0.46 and a C-Index of 0.83. In comparison, Random Forest regression turned out to be the best regression model, achieving a worse NRMSE of 0.52 and the same C-Index as the Random Survival Forest Model.

Table 4 shows that considering both an NRMSE of 0.43 and a C-Index of 0.84, Gradient Boosting Survival is the best survival analysis to predict the time to re-share fake news. On the other hand, linear MTLR achieves a better C-Index of 0.86, but a much worse NRMSE of 1.4. In comparison, Random Forest regression turned out to be the best regression model also in this dataset, achieving, however, worse scores of NRMSE (0.50) and C-Index (0.83) compared to the Gradient Boosting Survival Model.

Fig. 1 Top-ten most important features when computed using the RSF model on the **real news** dataset (top) and the GBS model on the **fake news** dataset (bottom)

Overall, these experimental results show that, in general, survival analysis is better than regression at predicting the time to share for both the cases of real and fake news.

Feature Importance For the best-performing models (Random Survival Forest for real news and Gradient Boosting Survival for fake news), we used the Mean Decrease in Accuracy (MDA) algorithm from the *eli5* Python package [4] to compute the feature importance for each model when trained on the entire dataset.[5] This algorithm systematically runs a specified model on the data while removing one feature each time and computes the mean decrease in model performance via the C-Index metric. The features are weighted and ordered according to importance. (The higher the mean C-Index decrease, the most important the feature.) Results are shown in Fig. 1.

Among the top-ten features computed to have the highest importance for predicting time to share on the real news dataset, we see that *similarity news* is the most important feature for predicting real news sharing. The two features relating to tie strength follow, implying that a user who closely follows everything the influencer (a user who shared the given news article to this user) shares, carefully responds to most shares. *AllPunc*, which is the measure of proper punctuation used by the user in their tweets, ranks fourth, while *similarity tweets* ranks fifth.

Among the top-ten features computed to have the highest importance for predicting time to share on the fake news dataset, we see that *num_propernouns*, the measure of proper nouns used by the given user in their tweets, ranks highest for predicting time to share, followed by the measure of possessive nouns used in the news article. The similarity to previously shared news articles ranks third, compared to first for real news, followed by the two influencer tie strength measures. Gender, number of followers, and the number of posts during the weekend are features that appear to be more important for predicting the time to share real news.

6 Conclusions

Through survival analysis, we have addressed the problem of predicting the time to share real and fake news on online social networks. Our experimental results show that survival analysis achieves better results than regression on the considered problem. Specifically, Random Survival Forest performs the best for real news (NRMSE of 0.46 and C-Index of 0.83), and Gradient Boosting Survival performs the best for fake news (NRMSE of 0.44 and C-Index of 0.84). We also investigated feature importance and showed similarities and differences between real and fake news.

One challenge in applying survival analysis to the prediction of news sharing is the choice of the time window to

[4] https://eli5.readthedocs.io/en/latest/overview.html

[5] We used the Mean Decrease in Accuracy method to compute feature importance as this method is independent of the prediction model in use and hence allows for comparison among different models. In our case we compare Random Survival Forest and Gradient Boosting Survival.

consider for the event of interest. According to our data, the majority of the news articles are re-shared within two days, while re-sharing after the first 48 h is sporadic. Hence, we considered a time window of two days. Extending the end of this time window would have introduced outliers that could have biased the predicted times. As a consequence, one limitation is that events occurring after the 48 h are considered censored, i.e., as non-sharing instances.

In future work, we plan to combine our survival analysis models with diffusion models working with explicit network information to simulate better how real and fake news diffuses in online social networks. We will use these new diffusion models to simulate news spread in the social network and reason about the efficacy of possible interventions to combat the spread of fake news.

Acknowledgements This research has been supported by the National Science Foundation under the award numbers CCF-1950599 and CNS-1943370.

Author Contributions F.S. proposed and lead the research, and wrote the main manuscript text; C.D., V.K., and M.Z. took care of the implementation, produced the experimental results, and wrote the paper; L.B. contributed to the mathematical foundations of this research.

Declarations

Competing interests The authors declare no competing interests.

References

1. Molina, M.D., Sundar, S.S., Le, T., Lee, D.: "Fake news" is not simply false information: A concept explication and taxonomy of online content. Am. Behav. Sci. **65**(2), 180–212 (2021). https://doi.org/10.1177/0002764219878224
2. Spezzano, F., Shrestha, A., Fails, J.A., Stone, B.W.: That's fake news! reliability of news when provided title, image, source bias & full article. Proc. ACM Hum. Comput. Interact. **5**(CSCW1), 1–19 (2021). https://doi.org/10.1145/3449183
3. Raponi, S., Khalifa, Z., Oligeri, G., Di Pietro, R.: Fake news propagation: a review of epidemic models, datasets, and insights. ACM Trans. Web (TWEB) **16**(3), 1–34 (2022)
4. Rangel, F., Giachanou, A., Ghanem, B., Rosso, P.: Overview of the 8th Author Profiling Task at Pan 2020: Profiling Fake News Spreaders on Twitter. In: CLEF (2020)
5. Jin, F., Dougherty, E., Saraf, P., Cao, Y., Ramakrishnan, N.: Epidemiological modeling of news and rumors on twitter. In: Proceedings of the 7th Workshop on Social Network Mining and Analysis, pp. 1–9 (2013)
6. Kempe, D., Kleinberg, J., Tardos, É.: Maximizing the spread of influence through a social network. In: SIGKDD, pp. 137–146 (2003)
7. Raponi, S., Khalifa, Z., Oligeri, G., Di Pietro, R.: Fake news propagation: A review of epidemic models, datasets, and insights. ACM Trans. Web **16**(3) (2022)
8. Joy, A., Shrestha, A., Spezzano, F.: Are you influenced?: Modeling the diffusion of fake news in social media. In: Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, pp. 184–188. ACM
9. Wang, P., Li, Y., Reddy, C.K.: Machine learning for survival analysis: a survey. ACM Comput. Surv. (CSUR) **51**(6), 1–36 (2019)
10. Murayama, T., Wakamiya, S., Aramaki, E., Kobayashi, R.: Modeling the spread of fake news on twitter. PLoS ONE **16**(4), 1–16 (2021)
11. Zafarani, R., Abbasi, M.A., Liu, H.: Social Media Mining: an Introduction. Cambridge University Press, Cambridge, UK (2014)
12. Ma, L., Lee, C.S., Goh, D.H.: Understanding news sharing in social media from the diffusion of innovations perspective. In: GREENCOM-ITHINGS-CPSCOM, pp. 1013–1020 (2013). IEEE
13. Lee, C.S., Ma, L.: News sharing in social media: the effect of gratifications and prior experience. Comput. Hum. Behav. **28**(2), 331–339 (2012)
14. Vosoughi, S., Roy, D., Aral, S.: The spread of true and false news online. Science **359**(6380), 1146–1151 (2018)
15. Ruffo, G., Semeraro, A., Giachanou, A., Rosso, P.: Studying fake news spreading, polarisation dynamics, and manipulation by bots: a tale of networks and language. Comput. Sci. Rev. **47**, 100531 (2023)
16. Shu, K., Wang, S., Liu, H.: Understanding user profiles on social media for fake news detection. In: 1st IEEE International Workshop on Fake MultiMedia (FakeMM 2018) (2018)
17. Guess, A., Nagler, J., Tucker, J.: Less than you think: prevalence and predictors of fake news dissemination on facebook. Sci. Adv. **5**(1), 4586 (2019)
18. Shrestha, A., Spezzano, F.: Characterizing and predicting fake news spreaders in social networks. Int. J. Data Sci. Anal. **13**(4), 385–398 (2022)
19. Shu, K., Mahudeswaran, D., Wang, S., Lee, D., Liu, H.: Fakenewsnet: a data repository with news content, social context, and spatiotemporal information for studying fake news on social media. Big Data **8**(3), 171–188 (2020)
20. Bak-Coleman, J.B., Kennedy, I., Wack, M., Beers, A., Schafer, J.S., Spiro, E.S., Starbird, K., West, J.D.: Combining interventions to reduce the spread of viral misinformation. Nat. Hum. Behav. **6**(10), 1372–1380 (2022)
21. Bakshy, E., Rosenn, I., Marlow, C., Adamic, L.: The role of social networks in information diffusion. In: Proceedings of the 21st International Conference on World Wide Web, pp. 519–528 (2012)
22. Kimura, M., Saito, K.: Tractable models for information diffusion in social networks. In: Knowledge Discovery in Databases: PKDD 2006: 10th European Conference on Principles and Practice of Knowledge Discovery in Databases Berlin, Germany, September 18-22, 2006 Proceedings 10, pp. 259–271 (2006). Springer
23. Yu, C.-N., Greiner, R., Lin, H.-C., Baracos, V.: Learning patient-specific cancer survival distributions as a sequence of dependent regressors. In: Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems, vol. 24. Curran Associates, Inc. (2011)
24. Fotso, S.: Deep neural networks for survival analysis based on a multi-task framework. CoRR (2018) arXiv:1801.05512
25. Katzman, J.L., Shaham, U., Cloninger, A., Bates, J., Jiang, T., Kluger, Y.: DeepSurv: personalized treatment recommender system using a cox proportional hazards deep neural network. BMC Med. Res. Methodol. **18**(1), 24 (2018)
26. Lee, C., Zame, W., Yoon, J., Schaar, M.: DeepHit: A deep learning approach to survival analysis with competing risks. Proc. Conf. AAAI Artif. Intell. **32**(1) (2018)
27. Shrestha, A., Spezzano, F.: Textual characteristics of news title and body to detect fake news: A reproducibility study. In: ECIR, Proceedings, Part II. Lecture Notes in Computer Science, vol. 12657, pp. 120–133. Springer, Cham (2021)
28. Solovev, K., Pröllochs, N.: Moral emotions shape the virality of covid-19 misinformation on social media. In: Proceedings of the ACM Web Conference 2022. WWW '22, pp. 3706–3717. Associ-

ation for Computing Machinery, USA (2022). https://doi.org/10.1145/3485447.3512266
29. Yaqub, W., Kakhidze, O., Brockman, M.L., Memon, N., Patil, S.: Effects of credibility indicators on social media news sharing intent. In: Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. CHI '20, pp. 1–14 (2020)
30. Granovetter, M.: The strength of weak ties: a network theory revisited. Sociol. Theor. **1**, 201–233 (1983)
31. Wang, Z., Hale, S., Adelani, D.I., Grabowicz, P., Hartman, T., Flöck, F., Jurgens, D.: Demographic inference and representative population estimates from multilingual social media data. In: The World Wide Web Conference, pp. 2056–2067 (2019)
32. Hemphill, L., Culotta, A., Heston, M.: Polar scores: measuring partisanship using social media content. J. Inform. Technol. Politic. **13**(4), 365–377 (2016)
33. Chamberlain, J.M., Spezzano, F., Kettler, J.J., Dit, B.: A network analysis of twitter interactions by members of the us congress. ACM Trans. Soc. Comput. **4**(1), 1–22 (2021)
34. Mohammad, S.M.: Word affect intensities. arXiv preprint arXiv:1704.08798 (2017)
35. Hutto, C.J., Gilbert, E.: Vader: A parsimonious rule-based model for sentiment analysis of social media text. In: AAAI ICWSM (2014)
36. Wang, W., Hernandez, I., Newman, D.A., He, J., Bian, J.: Twitter analysis: studying us weekly trends in work stress and emotion. Appl. Psychol. **65**(2), 355–378 (2016)
37. Řehůřek, R., Sojka, P.: Software Framework for Topic Modelling with Large Corpora. In: Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, pp. 45–50. ELRA, Paris, France (2010)
38. Tausczik, Y.R., Pennebaker, J.W.: The psychological meaning of words: Liwc and computerized text analysis methods. J. Lang. Soc. Psychol. **29**(1), 24–54 (2010)
39. Pölsterl, S.: scikit-survival: a library for time-to-event analysis built on top of scikit-learn. J. Mach. Learn. Res. **21**(212), 1–6 (2020)
40. Fotso, S., et al.: PySurvival: Open source package for survival analysis modeling (2019–). https://www.pysurvival.io/
41. pycox. https://github.com/havakv/pycox
42. Bishop, C.M.: Pattern Recognition and Machine Learning Information Science and Statistics., 5th edn. Springer, Cham (2007)
43. Hoerl, A.E., Kennard, R.W.: Ridge regression: biased estimation for nonorthogonal problems. Technometrics **12**(1), 55–67 (1970)
44. Tibshirani, R.: Regression shrinkage and selection via the lasso. J. Royal Stat. Soc. Ser. B (Methodological) **58**(1), 267–288 (1996)
45. Ho, T.K.: Random decision forests. In: Proceedings of 3rd International Conference on Document Analysis and Recognition, vol. 1, pp. 278–282 (1995). IEEE
46. Murtagh, F.: Multilayer perceptrons for classification and regression. Neurocomputing **2**(5), 183–197 (1991). https://doi.org/10.1016/0925-2312(91)90023-5
47. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E.: Scikit-learn: machine learning in python. J. Mach. Learn. Res. **12**, 2825–2830 (2011)
48. Dave, V.S., Hasan, M.A., Zhang, B., Reddy, C.K.: Predicting interval time for reciprocal link creation using survival analysis. Soc. Netw. Anal. Min. **8**(1), 16 (2018)
49. Box, G.E., Jenkins, G.M., Reinsel, G.C., Ljung, G.M.: Time Series Analysis: Forecasting and Control. Wiley, USA (2015)
50. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997)