Simulating Temporal User Activity on Social Networks with Sequence to Sequence Neural Models

Renhao Liu and Frederick Mubang and Lawrence O. Hall

Abstract— **The prediction of long-term activities of groups of users and clusters of activities around a subject in social networks is a very challenging task. In this paper, we propose a novel temporal neural network framework that tracks user engagement and activity associated with particular subjects (e.g. CVE IDs) across online platforms. The framework is able to simulate which user will do what activity and at what time. Furthermore, this framework captures groups of users reacting to an event. It also captures responses to an event on a platform and the influence of the event on activity on other platforms over time. The proposed framework aims to predict future user activity related to specific subjects across platforms. The framework also illustrates the importance influence of activities that occur on other platforms when predicting user activity for particular events on a different platform. The learned model can do simulations in a timely manner. We evaluated our user group activity prediction method on the CVE (Common Vulnerabilities and Exposures) related user groups (software vulnerability) using 3 public online social network datasets: Github, Reddit, and Twitter. Groups of users who work on a particular CVE ID are identified. Each user group has information on all users' activities related to a CVE ID. The 3 datasets from Github, Reddit, and Twitter contain more than 490,000 cross platform activities related to over 20,000 user groups (CVE IDs) from more than 50,000 users. Compared to the proposed baseline, our simulation method is better in both predictions of total activity volume over time and activity associated with an individual CVE ID.**

I. Introduction

Simulating individual users and groups of users activity resulting in cascades of activity from an event (e.g. tweet) across complex online social environments is challenging due to multiple factors. The tension between scale and accuracy [15] can cause prediction accuracy to hardly improve with big data. Users' actions and interactions are often complex enough to make it difficult to extract simple explanatory patterns. On the other hand, as the interest in online activity influence on events is increasing, accurately predicting users' online activities in a group for long periods of time takes on more importance in recognition of anomalies, projecting computational loads, finding emergent undesired social phenomena for intercession, etc.

This paper addresses the challenge of predicting user related cross platform activities 1 year out into the future in a focused online collaborative environment using 3 online platforms, GitHub, Reddit, and Twitter. Our framework also captures what we call "partial cascades" that connect responses to a post on Reddit or Tweet on Twitter. Note, that it is not currently possible to connect users across platforms due to the anonymized user IDs available on each platform. Our focus is on predicting the activities of a user associated with an ID in a particular platform, while using the activities of users in other platforms as informative features. The activities we focus upon are those related to the Common Vulnerabilities and Exposures (CVE) [16] domain with all user activity records containing at least 1 mention of a CVE ID in a comment, quote, retweet, or reply, or in the name of a Github repository, Reddit thread, or Twitter thread. CVE's describe known computer security vulnerabilities that have been publicly disclosed.

A user can do multiple activities across platforms. In Github, users can contribute to different software repositories via different types of activities (e.g., push, fork, watch, issue comment). Users can also mention and discuss CVE IDs on Reddit and Twitter (e.g., post, comment, tweet, reply). Analyzing how software vulnerabilities evolve through online collaborative environments is important.

Predicting user activity across the GitHub, Reddit, and Twitter platforms is significantly challenging. First, in Github, the open-source repositories are generally contributed to by different types of users, thus there usually is no predefined activity pattern. Second, users on Github, Reddit, and Twiiter are generally from all over the world, which makes for wide variations in action time. Third, and also the most important one, information diffusion across platforms is complicated by the existence of different kinds of user networks. Influence of a user's activities on others still merits further exploration.

If you view activity simulation from the user perspective, as noted previously, in most hours a user does nothing related to a CVE ID on any platform. So, a highly accurate predictive model will account for few events. In longer time intervals (e.g. a day) many users take no action. Probabilistic learning approaches will under predict to get maximum accuracy on the "no action" class and almost all machine learning approaches have a probabilistic description.

One way to look at the problem is to treat CVE ID related user groups as an entity and simulate user activity at the group level. The question is can we capture the sequence of events on each of the three platforms for particular CVE IDs? Our approach is a temporal neural network that learns the likely sequence of activities. This forms the basis for a simulator that can predict activity in the future. Other information to predict during the simulation like the time associated with an activity and user who did the activity are learned as sequences too. Our focus is not precisely on the user as our data is anonymized and we cannot track

Department of Computer Science and Engineering ENG030, University of South Florida, Tampa, FL 33620, lohall@mail.usf.edu

users across platforms (think different logins or identities). As a result, our user prediction for an activity is learned and evaluated by type of user (new user or old user).

Our experiments show the proposed simulation method predicted CVE-ID related user activity including partial cascades across platforms with reasonably low error. Compared to the proposed baseline, our simulation method is better in both total activity volume over time and individual CVE ID predictions. Our experiments also show that using features from other platforms increases future activity prediction accuracy on a given platform.

The remainder of this paper is organized as follows. We discuss related work in Section II. In Section III, we describe the datasets from the 3 online platforms. We present our model design and prediction results in Section IV, Section V, and Section VI. Finally, we summarize the contributions of this work in Section VII.

II. Related Work

To the best of our knowledge there is no previous work that explores both long-term and cross-platform user group activity prediction. However, as we will discuss below, there has been some related work on the prediction of future events using learned models.

In [13], a novel framework was proposed which extracts user interests inferred from activities (a.k.a., activity interests) in multiple social collaborative platforms to predict users' platform activities. It explored 2 software development communities: GitHub and Stack Overflow. The experiments showed that combining both direct and cross platform activity prediction approaches yielded the best accuracies for predicting user activities on GitHub (AUC=0.75) and Stack Overflow (AUC=0.89).

There are various papers that show positive results using websites and/or social media to predict the actions of a population within some paradigm. In [19], Pagolu et. al observed a correlation between the sentiment scores of Twitter tweets, and stock market movements. They used an N-Gram representation with Word2Vec to extract sentiment features. The extracted sentiment features were then used with a Random Forests classifier to train an ensemble model. The classifier was used to predict stock price movement (i.e. previous day stock price greater than current day stock price). In [22], the paper showed the correlation between sentiment and changes in stock prices. In [10], some success at predicting Cryptocurrencies price movement was shown. The use of sentiment of comments in related online communities enabled cryptocurrency price movement prediction with some accuracy. Cyber-security (cyber) and cryptocurrency (crypto) domain user activity analysis have a considerable presence in Github as well [14] [9] [12].

In a large-scale study of news in social media, one paper analyzed 11 million posts. They investigated the propagation behavior of users that directly interact with news accounts identified as spreading trusted versus malicious content. The goal was to examine how evenly, how many, how quickly, and which users propagate content from various types of news sources on Twitter [5].

Pedestrian motion prediction using a short history of their and neighbors past behavior was discussed in [4]. The prediction of people's trajectory in crowded spaces was addressed in [1] using LSTM based neural networks. There has been work on predicting how a patient will do over time in the ICU using a learned model [3].

There has also been work on agent based simulations of social systems [8] which has a learning component for the agents, but it is a different type of low-level approach than taken here. Here, we use purely learned models and look at simulating results over time using daily predictions to make predictions further in the future.

III. Datasets

The focus of our experiments is user group activity related to the CVE domain on 3 public datasets obtained from GitHub, Reddit, and Twitter. CVE IDs in comments or an indication that a repo was affected by the CVE domain were used to extract related data for Github. Related subReddit's were used in Reddit and Tweets in Twitter are also used from the CVE domain. Groups of users who work on a particular CVE ID are identified. Each user group records all users' activities related to a CVE ID.

The Common Vulnerabilities and Exposures (CVE) system provides a reference-method for publicly known information-security vulnerabilities and exposures. The National Cybersecurity FFRDC, operated by the Mitre Corporation, maintains the system, with funding from the National Cyber Security Division of the United States Department of Homeland Security. The system was officially launched for the public in September 1999. The Security Content Automation Protocol uses CVE, and CVE IDs are listed on MITRE's system as well as in the US National Vulnerability Database [17].

Data in our focused domain was available from March 1, 2016 to March 31, 2018 for training which includes 24,381 CVE IDs. This training data has 348,378 activities related to the CVE domain from 26,269 users on Github, 22,351 activities from 7,035 users on Reddit, and 103,048 activities from 9,998 users on Twitter respectively. Our data was collected in a collaboration with Leidos[1]. The data is summarized in Table I. Note that since users are anonymized and can have different names on different platforms, we typically cannot track a user across platforms, though some are active on multiple platforms.

TABLE I: CVE Related Activities Across Platforms in Training

Dataset	Activities	Users
Reddit	22,351	7,035
Twitter	103,048	9,998
Github	348,378	26,269

[1] Thanks to Leidos for providing the data.

Initial conditions data is from April 1, 2018 to May 31, 2018 with 64 CVE IDs. It records users' 2,315 activities related to the CVE domain from 1,251 users on Github, 642 activities from 324 users on Reddit, and 3,187 activities from 2,033 users on Twitter respectively.

Data from June 1, 2018 to April 30, 2019 was used for testing which includes 99 CVE IDs. It has users' 9,321 activities related to the CVE domain from 1,812 users on Github, 1,794 activities from 772 users on Reddit, and 2,481 activities from 1,443 users on Twitter respectively.

Users can do different activities on the three platforms. The 10 GitHub activities are (1) Push, (2) Create, (3) Watch, (4) Issue Comment, (5) Pull Request, (6) Issues, (7) Fork, (8) Delete, (9) Pull Request Review, (10) Commit Comment. The 2 Reddit activities are (1) Post, (2) Comment. The 4 Twitter activities are (1) Tweet, (2) Retweet, (3) Quote, (4) Reply. For Reddit and Twitter the most common activities are Comment and Retweet, respectively.

A. GitHub

Github is primarily an open-source software collaboration platform where users contribute to Github repositories via (code) commits, pushes, pull-requests, and issues raised. Users can also "watch" repositories to receive alerts on updates, and can "fork" (i.e., copy) public repositories to make their own local software modifications and start a repository contribution. GitHub is home to over 28 million public repositories and 40 million users. The dataset of events on the public repositories is publicly available [2].

B. Reddit

Reddit is a popular website where users can post content on a bulletin board system, comment on each other's posts, and vote on them to show agreement and disagreement. Content in Reddit is organized into topic-specific subreddits. As a rich definition of community, subreddits are known as the place where users can self-organize into topic-based groups. Users can post content, comment, or vote once they are logged into their account. Similar to an online social network, users in Reddit can also add other users as friends so they can receive a notice of their friends' updates. Users can also subscribe to subreddits to see more updates on the contents they are interested in which allows for personalization of content [21].

C. Twitter

Twitter is a micro-blogging platform where users broadcast messages (i.e., tweets) publicly or share privately to their follower network. Twitter allows tweets to be tagged with hashtags, and users can post messages, URLs, images, etc., under one or multiple hashtags [18].

IV. Model Design

In order to simulate user activity across platforms, a temporal neural network framework was designed to learn the temporal concepts for all the features needed. Each CVE ID will have a group of users who work on it and has its own sequence data. This includes the activities (e.g tweets, posts, pushes) associated with it. The proposed framework was trained to capture the temporal pattern and partial cascades on Twitter and Reddit from CVE IDs sequence data. We could predict a tweet and retweets, replies, and quotes responding to it or a post and the resultant cascade of comments.

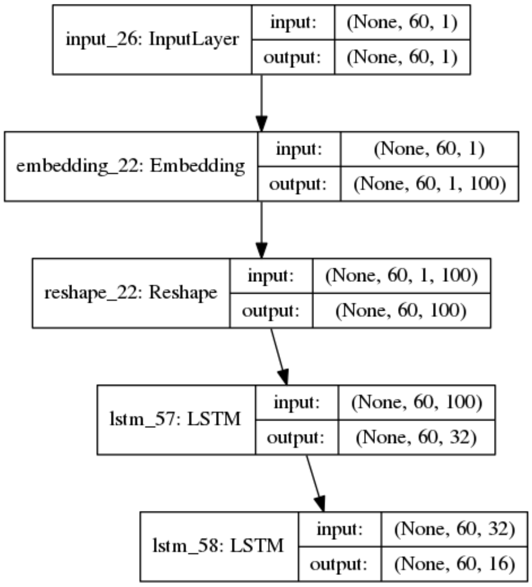


Fig. 1: Model Architecture Branch for Each Feature before Concatenate Layer

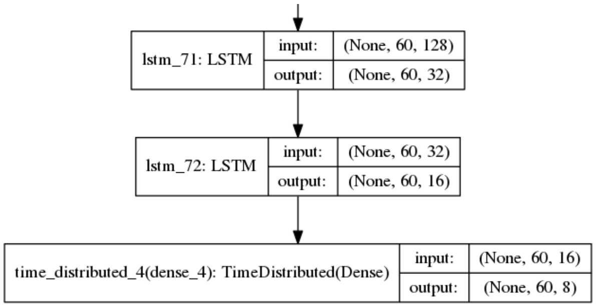


Fig. 2: Model Architecture after Concatenate Layer

Our framework used 11 features to learn and to simulate future user activity which are (1) nodeID, (2) nodeUserID, (3) parentID, (4) rootID, (5) actionType, (6) nodeTime, (7) informationID, (8) platform, (9) has URL, (10) links to external, and (11) domain linked features. The NodeID feature records the ID of one activity; nodeUserID records the ID of the user who did that activity; parentID is the ID of the activity to which the specified nodeID is an immediate reply; rootID is the ID of the original start activity which is the predecessor of, for example, a Reddit comment; actionType is the type of activity performed by the user (e.g tweets, posts, pushes); nodeTime is the timestamp of when the activity happened; informationID is the CVE ID of the unit of information, which would be the focus of a group of users; platform records the platform on which the activity happened, the has URL feature is a binary indicator of whether this activity contains at least one URL; links to external feature is a binary indicator of whether this activity contains links outside the platform; and the domain linked feature records the set of URL domain(s) mentioned.

An integer encoder was used to represent different entities for each of the features except the nodeUserID feature and binary features. However, nodeID, parentID, and rootID features shared one single integer encoder because those features are related and we can extract partial cascades based on these features. Remember, since users are anonymized in our dataset and have different IDs on different platforms, we typically cannot track a user across platforms. In our data, the nodeUserID feature was converted to a binary indicator to represent whether or not this user is new. An integer encoder was not used for the nodeTime feature. Instead of using timestamps in the model directly, we converted the timestamps belonging to a CVE ID to that CVE's lifetime feature from the very first seen action for that CVE in the dataset. The granularity used for the nodeTime feature is daily, but it doesn't have to be integers. For example, a certain CVE ID related action may start with an initial post done by user A and 1 day and 12 hours later user B made a comment on that post; then the nodeTime feature for the post action is 0 and it is 1.5 for the comment action. Min-max normalization was also applied to each feature.

Each feature first went through an embedding layer to capture the relationship between entities, and each embedding layer was set to have 100 units; it was followed with 2 layers of LSTMs [7] in order to capture the temporal pattern for each feature. The first layer of LSTM had 32 units and second had 16; the LSTMs outputs from each feature were concatenated into one feature vector to capture feature correlation; then the feature vector was sent to another 2 layers of LSTMs, the first layer of LSTM also had 32 units and second had 16; finally the output of the LSTM was connected with a fully connected layer to generate a prediction. The optimizer used in the framework was Adam and the loss function used was MAE (Mean Absolute Error). Training epochs were set to 100. Figure 1 shows what a branch in the model looks like for each feature before the concatenate layer. Figure 2 shows the model architecture after the concatenate layer.

How many timesteps to use in a LSTM model is an open question related to the specific prediction task. For the 24,381 CVE IDs in our training data, the most active one has 5,302 action events; the minimally active one has only 31. In our experiment, we found when setting timesteps to 60 that the framework delivered good performance. If one CVE ID has less than 60 action events in its lifetime we padded 0s at the beginning. In order to optimize simulation time for the multiple CVE IDs, parallel computing was implemented in our framework and we set the output timesteps of the framework to be 60, as well, rather than 1. To generate training samples, we used a sliding window with a size of 90 timesteps. Inside the 90 timesteps, the first 60 timesteps were used as the feature vector, and the last 60 timesteps which have 30 timesteps which overlap the feature vector, were used as the target vector. Based on the slicing strategy, we generated 242,418 training samples with 60 timesteps sequences from the 473,777 activities in the training set.

Each training sample was a representation of users' activity within a particular 60 timestep window belonging to a CVE ID. The feature vector contained 660 values, corresponding to 60 timesteps * 11 features used. The target vector for each sample was comprised of 660 values as well, but because of the overlapping strategy, the last 30 timesteps * 11 features in the target vector were used as the needed prediction.

V. Simulation Process

After model training, we can use the trained model to simulate all CVE IDs (with associated users) related user activities and capture partial cascades of events in the future.

The future user activity simulation task focused on the subset of 99 CVE IDs in the testing data from June 1, 2018 to April 30, 2019. Each of the 99 CVE IDs has a group of users who will interact with events involving it and each user's activities and associated time were simulated. Since we performed experiments with cold-start forecasting [23], we needed at least 60 timesteps of feature vector from initial conditions data for those 99 CVE IDs. If one CVE ID doesn't have 60 activities in the initial condition data, we padded more activity records from training data in front of the first occurring activities in initial condition to make 60 timesteps available. If a CVE ID did not have 60 activities when combining training and initial condition data, we padded with 0's in front.

After generating the input feature vector for all 99 CVE IDs, we fed it into the trained model. All 99 CVE IDs were simulated in parallel to accelerate simulation time. Because of the overlapping strategy used between input feature vector and output target vector, every time when the model generated 60 timesteps output, the last 30 timesteps of the target vector were used as prediction. The framework will check the nodeTime feature predicted for each CVE ID to verify whether all CVE IDs reach the ending timestamp in the test data (April 30, 2019). If not, the framework will concatenate the last 30 timesteps of feature vector with the predicted last 30 timesteps of target vector, generating the updated 99 CVE IDs' sequence input feature vector and feed it into the model again. The framework will stop simulation when all 99 CVE IDs' predicted nodeTime feature reaches the ending timestamp in the test data, April 30, 2019.

All the predicted target vectors were converted back to the expected range. Activity associated timestamps in the simulation can be generated from the NodeTime feature predicted as well. This conversion process was also designed to operate in parallel in the framework.

VI. Simulation Results

To evaluate our simulation results, we checked multiple aspects of our proposed framework's performance. We evaluated our simulation results on the total activity volume over time analysis from 3 platforms. In addition, we also evaluated our simulation results on the individual CVE ID analysis including multiple measurements.

TABLE II: CVE Domain Event Occurrences in Test Data

Event	Event Count	Frequency
Push	840	6.17%
Create	207	1.52%
Watch	1,230	9.04%
Issue Comment	1,114	8.19%
Pull Request	1,186	8.72%
Issues	3,797	27.92%
Fork	921	6.77%
Delete	10	0.07%
Pull Request Review Comment	13	0.09%
Commit Comment	3	0.02%
Post	63	0.46%
Comment	1,731	12.73%
Tweet	484	3.55%
Retweet	1,980	14.56%
Quote	9	0.06%
Reply	8	0.05%
Total	13,596	100%

TABLE III: WSMAPE Results for CVE Domain. PF stands for our proposed framework, SB means the shift baseline; and SPF means the single platform framework. C - Comment, PR - Pull request

Event	PF	SB	SPF	Winner
Push	2.95%	4.99%	5.84%	PF
Create	0.96%	0.72%	1.21%	SB
Watch	3.59%	4.91%	8.71%	PF
Issue C	5.01%	3.99%	7.50%	SB
Pull Request	1.43%	3.74%	2.81%	PF
Issues	16.91%	18.54%	13.68%	SPF
Fork	4.12%	3.59%	15.80%	SB
Delete	0.05%	0.05%	0.03%	SPF
PR Review C	0.08%	0.09%	0.06%	SPF
Commit C	0.01%	0.01%	0.02%	PF
Post	0.36%	0.30%	0.24%	SPF
Comment	5.57%	7.55%	12.49%	PF
Tweet	2.30%	2.55%	3.51%	PF
Retweet	11.13%	8.29%	14.40%	SB
Quote	0.05%	0.05%	0.01%	SPF
Reply	0.04%	0.03%	0.01%	SPF
WSMAPE	54.67%	59.44%	77.05%	PF

A. *Total Activity Volume Over Time Analysis*

The symmetric mean absolute percentage error (SMAPE) [6] per event is reported in the total activity volume over time analysis. This analysis measures whether the proposed framework predicted enough activities daily in the test period and how similar it is when compared to the ground truth regardless of CVE IDs. Because the activities on the 3 platforms are not evenly distributed, results are summarized using a weighted SMAPE (WSMAPE), where the weight is the percent of activities from the total number of activities for a particular event. Table II shows the number of events of each type from June 1, 2018 to April 30, 2019 in the test data and their percentage of the overall total number of events in the CVE domain.

In order to compare the proposed simulation framework to another prediction method, we used a baseline method which predicted future CVE-ID related activities by shifting history records to future timesteps. In particular, the ”shifted” baseline takes observed historical activity for the specified CVE IDs and shifts it such that the timestamps in the data correspond to the test period. If there is not enough historical data available, the same data will be shifted multiple times until the full period is covered. All the necessary data in training from July 1, 2017 to May 31, 2018, and related to the 99 CVE IDs simulated, was shifted to the test period.

Figure 3 shows an example of the predicted total Watch activity volume over time compared to ground truth and shifted baseline for the 99 CVE IDs. From the figure, our proposed framework performed much better in total activity volume over time prediction than the shifted baseline. The shifted baseline over predicted almost 5 times more than our proposed method for the Watch event. However, our approach does miss some small spikes of activity.

In order to explore the usefulness of cross platform features, we also created a single platform simulation method for comparison purposes. The single platform simulation method used the exact same model framework to capture knowledge and patterns from the training data, but unlike our cross platform framework whose sequence samples include features and records from all 3 platforms, it has sequence samples with features from a single platform (e.g. Twitter or GitHub). For example, a single platform simulation for Github only used features from the Github platform itself to predict future Github platform events. Three single platform simulations were used to simulate CVE-ID related activities from Github, Reddit, and Twitter respectively. Final simulation output was combined from these 3 individual platform simulation outputs.

Table III shows the WSMAPE results for the 3 platforms' activities from our 3 proposed simulation methods. Overall, our proposed cross platform frameowrk (PF) had the lowest WSMAPE (54.67%) compared to the shifted baseline (SB) and the single platform framework (SPF) method.

Based on the results, it seems that in order to achieve the lowest overall WSMAPE considering cross platform features is the best solution. Cross platform features are particularly useful (Table III) for activities prediction like Push, Watch, Pull Request, Commit Comment, Comment, and Tweet.

Especially when compared to the single platform framework proposed, our cross platform framework showed that cross platform features helped with the overall community action event volume for Tweet, Comment, Watch, Push events, and etc.

The results matched our expectation in general. Reddit and Twitter features helped greatly with predicting Github Forks and Watches. For Fork, the WSMAPE decreased by 11.68% when Reddit and Twitter features were used. For Watch, the decrease was by 5.12%. Note that a Fork is an event in which a user creates a new repository from an existing one so they can make their own edits to it without affecting the original. A Watch is an event in which a user marks a repository

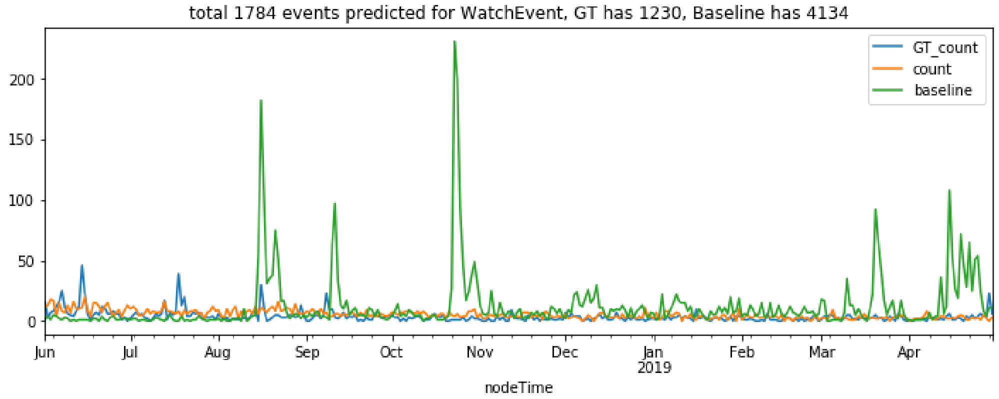
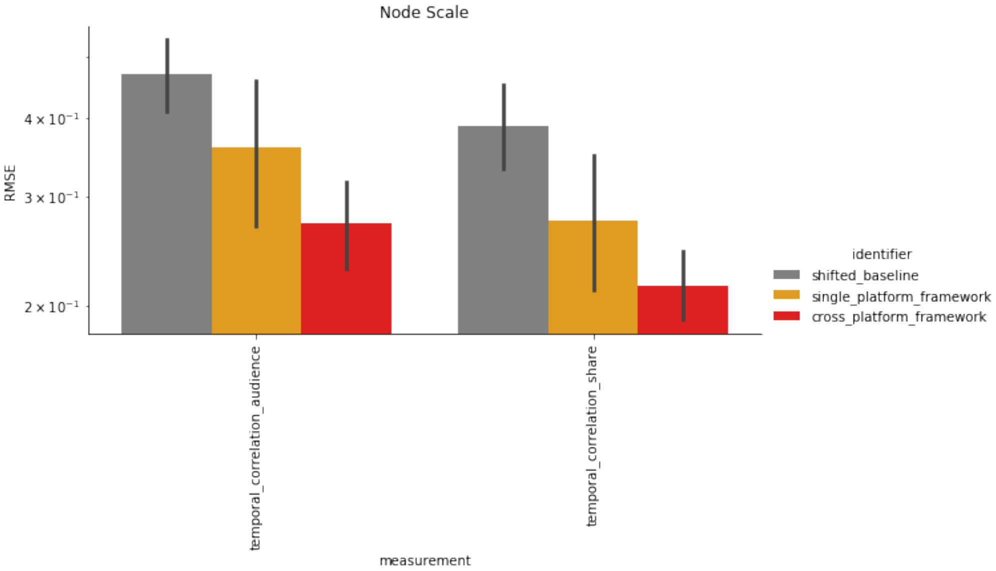


Fig. 3: Predicted Watch Event Volume Over Time Compared to Ground Truth and Shifted Baseline. The orange line plot *count* is our proposed framework results. The blue *GT-count* line plot is the ground truth. The green *baseline* line plot is the baseline shifted-model.



(a) RMSE (lower better)

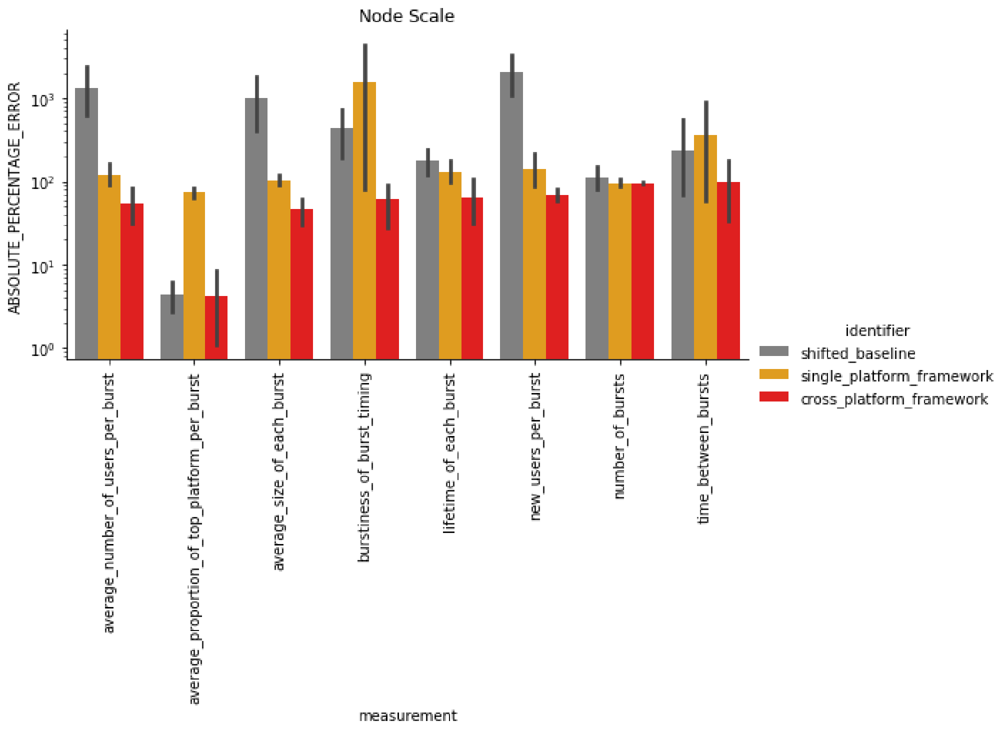
Fig. 4: Temporal Correlation Audience and Temporal Correlation Share for Cross Platform Spread Measurement

so she can be notified of any changes made to it. In other words, these 2 events are directly related to the popularity of a repository. It stands to reason that a popular repository will have more people "watching" it, and more people copying it, or "forking" it to use it for themselves. The results show that there are some exogenous Twitter and Reddit events that inform the neural network of an incoming increase or decrease in the number of Forks and Watches. This can be useful information for software developers or companies as it would help them know if a repository related to their product is capturing the interest of consumers.

Also, it is very interesting that for some events the single platform framework (SPF) achieved the lowest error. For example, the Issues event is a high-frequency event, with a frequency 27.92% in the test data. However, the results showed that cross platform features did not help with prediction very much. Only GitHub features appear to be the key features for predicting the Issues event. Recall that an Issues Event is any action related to the signalling or rectifying of a repository issue, such as a coding bug, for instance. From the results, we speculate that Reddit and Twitter do not help with predicting Issues because if a developer finds a problem with code on Github, they will not go to Twitter or Reddit to tweet or post about it. Instead, they will raise an issue on Github. Other developers who see the issue on Github will work within Github to rectify it. They will not discuss it on other platforms.

B. Individual CVE ID Analysis

Unlike the total activity volume over time analysis which measured activity volume regardless of CVE IDs, In individual CVE ID analysis, we measured our simulations based on each of the CVE ID, which gave us a comprehensive



(a) Absolute Percentage Error (lower better)

Fig. 5: Multiple Measurements Related to Future Structure of Spread Prediction

evaluation on each group of users' activity prediction related to the CVE domain. All the measurements reported in this section are calculated from the code repository [20].

User group activity prediction for the 99 CVE IDs on all 3 platforms across the days from June 1, 2018 to April 30, 2019 was simulated and compared with the ground-truth data. Multiple measurements were calculated and the average values from the 99 CVE IDs were plotted. Measurement values distribution for the 99 CVE IDs was also included in Figures 4 and 5. Similar to the total activity volume over time analysis, for comparison purposes, we also examined our framework's performance on individual CVE IDs against the shifted baseline method and the single platform simulation method.

Temporal correlation of audiences measures whether different platforms show similar temporal patterns of audience growth for a CVE ID. Temporal correlation sharing measures whether sharing behaviour (e.g. retweets, comments) for IDs is similar between platforms. In Figure 4, these measurements show our cross platform framework is much better than the shifted baseline and the single platform framework to predict temporal patterns of sharing behavior and temporal patterns of audience growth across the 3 platforms.

Based on the feature setup in our framework design, not only were the group of user events related to CVE IDs predicted, but we also implicitly predict the properties of bursts for CVE IDs, where an increase in activity happens across platforms. An increase (even relatively small) of activity across platforms associated with a CVE ID constitutes a burst here. Measurements related to future structure of spread prediction that focused on "burst" were reported. Bursts were detected as described in [11]. A model that can predict the best hyperparameters of the burst detection algorithm based on the features of the input time series was learned. Burst detection code was implemented in the code repository [20] too.

The measurements shown in Figure 5 demonstrate our cross platform framework is much better than the shifted baseline and the single platform framework to predict future structure of spread focused on a "burst". Detailed measurement descriptions follow. Average number of users per burst measures how many users are involved in events for a CVE ID during each burst of activity on average. Average proportion of top platform per burst measures whether individual bursts of activity tend to occur on a single platform or are they distributed among platforms. Average size of each burst measures how many times the CVE-ID is shared per burst on average. So, you could think of how many retweets, etc. a tweet caused. Burstiness measures whether multiple bursts of renewed activity tend to cluster together in time. The lifetime of each burst measures how long each burst lasts on average. New users per burst measures how many new users interact with a CVE ID during each burst on average. Number of bursts measures how many renewed bursts of activity there are over time. Time between bursts measures how much time elapses between renewed bursts of activity on average.

Due to page limitations, only selected, representative plots are included.

VII. CONCLUSIONS

How user cross platform activity influences future users/activity when looking at groups of users who focus on a topic is important to understand in social networks. This work involved CVE related group activity simulation on social networking platforms GitHub, Twitter and Reddit. Due to the challenges for long term simulation using machine learning, user activity prediction across multiple online platforms is rarely explored. We proposed a novel cross platform framework to address the challenges.

Our proposed cross platform framework was found to provide solid simulation performance and be scalable. Results are reasonably close to ground truth on the simulation of user activity and bursts in terms of cascades in the CVE domains. Our framework also provides an approach to combining multiple user related features together inside one model which can provide fast long term multiple users simulation. Importantly, it was also shown that features from other platforms increase the accuracy of users' future activity on a given platform.

We applied our temporal learning models to do up to 1 year simulation with over 50,000 users. We used one Nvidia GTX 1080ti GPU to train the whole framework and training took 5 hours. In simulation, we completed the all predictions in 2 hours with that GPU. Our simulator can handle imbalanced data for predictions. The trained model didn't show bias towards a certain platform in the cross platform prediction. Generally, the performance of our proposed cross platform prediction framework produced solid results and it can also predict both the properties of partial cascades for CVE IDs and the distributions of these properties across CVE IDs.

ACKNOWLEDGEMENTS

This work was partially supported by the DARPA SocialSim Program and the Air Force Research Laboratory under contract FA8650-18-C7825 and National Science Foundation grant 1513126. The authors would like to thank Leidos for providing data.

REFERENCES

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L Fei-Fei, and S Savarese. Social LSTM: Human trajectory prediction in crowded spaces. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 961–971, 2016.

[2] Github Archive. Gh archive. http://www.gharchive.org/, 2018.

[3] Meiring C, Dixit A, Harris S, MacCallum N.S., Brealey D.A., Watkinson P.J., Jones A., Ashworth S., Beale R., Brett S.J., Singer M, and Ercole A. Optimal intensive care outcome prediction over time using machine learning. *PLoS ONE*, 13(11):1 – 19, 2018.

[4] T. Fernando, S. Denman, S. Sridharan, and C. Fookes. Soft + hardwired attention: An LSTM framework for human trajectory prediction and abnormal event detection. *Neural Networks*, 108:466 – 478, 2018.

[5] Maria Glenski, Tim Weninger, and Svitlana Volkova. Propagation from deceptive news sources who shares, how much, how evenly, and how quickly? *IEEE Transactions on Computational Social Systems*, 5(4):1071–1082, 2018.

[6] Paul Goodwin and Richard Lawton. On the asymmetry of the symmetric mape. *International journal of forecasting*, 15(4):405–408, 1999.

[7] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[8] Jesse Hoey, Tobias Schröder, Jonathan Morgan, Kimberly B. Rogers, Deepak Rishi, and Meiyappan Nagappan. Artificial intelligence and social simulation: Studying group dynamics on a massive scale. *Small Group Research*, 49(6):647 – 683, 2018.

[9] Sameera Horawalavithana, Abhishek Bhattacharjee, Renhao Liu, Nazim Choudhury, Lawrence O. Hall, and Adriana Iamnitchi. Mentions of security vulnerabilities on reddit, twitter and github. In *IEEE/WIC/ACM International Conference on Web Intelligence*, pages 200–207, 2019.

[10] Young Bin Kim, Jun Gi Kim, Wook Kim, Jae Ho Im, Tae Hyeong Kim, Shin Jin Kang, and Chang Hun Kim. Predicting fluctuations in cryptocurrency transactions based on user comments and replies. *PLoS ONE*, 11(8):1 – 17, 2016.

[11] J Kleinberg. Bursty and hierarchical structure in streams. *Data Mining and Knowledge Discovery*, 2003.

[12] Naoki Kobayakawa and Kenichi Yoshida. Study on influencers of cryptocurrency follow-network on github. In *Pacific Rim Knowledge Acquisition Workshop*, pages 173–183. Springer, 2019.

[13] Roy Ka-Wei Lee and David Lo. Wisdom in sum of parts: Multiplatform activity prediction in social collaborative sites. In *Proceedings of the 10th ACM Conference on Web Science*, WebSci '18, page 77–86, New York, NY, USA, 2018. Association for Computing Machinery.

[14] R. Liu, F. Mubang, L. O. Hall, S. Horawalavithana, A. Iamnitchi, and J. Skvoretz. Predicting longitudinal user activity at fine time granularity in online collaborative platforms. In *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)*, pages 2535–2542, Oct 2019.

[15] Ilias N. Lymperopoulos and George D. Ioannou. Understanding and modeling the complex dynamics of the online social networks: a scalable conceptual approach. *Evolving Systems*, 7(3):207–232, Sep 2016.

[16] Peter Mell and Tim Grance. Use of the common vulnerabilities and exposures (cve) vulnerability naming scheme. Technical report, NATIONAL INST OF STANDARDS AND TECHNOLOGY GAITHERSBURG MD COMPUTER SECURITY DIV, 2002.

[17] Peter Mell and Tim Grance. *Use of the common vulnerabilities and exposures (CVE) vulnerability naming scheme [electronic resource] : recommendations of the National Institute of Standards and Technology / Peter Mell, Tim Grance.* NIST special publication: 800-51. Gaithersburg, MD : U.S. Dept. of Commerce, Technology Administration, National Institute of Standards and Technology, 2002, 2002.

[18] Dhiraj Murthy. *Twitter*. Polity Press Cambridge, UK, 2018.

[19] V. S. Pagolu, G. Reddy, K.N.and Panda, and B. Majhi. Sentiment analysis of twitter data for predicting stock market movements. In *Signal Processing, Communication, Power and Embedded System (SCOPES), 2016 International Conference on*, pages 1345–1350. IEEE, 2016.

[20] PNNL. Pacific northwest national laboratory, socialsim, 2018.

[21] Philipp Singer, Fabian Flöck, Clemens Meinhart, Elias Zeitfogel, and Markus Strohmaier. Evolution of reddit: from the front page of the internet to a self-referential community? In *Proceedings of the 23rd international conference on world wide web*, pages 517–522, 2014.

[22] Huiwen Wang, Shan Lu, and Jichang Zhao. Aggregating multiple types of complex data in stock market prediction: A model-independent framework. *Knowledge-Based Systems*, 164:193 – 204, 2019.

[23] Christopher Xie, Alex Tank, Alec Greaves-Tunnell, and Emily Fox. A unified framework for long range and cold start forecasting of seasonal profiles in time series. *arXiv preprint arXiv:1710.08473*, 2017.