

Modeling Human Temporal Dynamics in Agent-Based Simulations

James Flamino
Rensselaer Polytechnic Institute
Troy, NY
flamij@rpi.edu

WeiKe Dai
Rensselaer Polytechnic Institute
Troy, NY
daiw2@rpi.edu

Boleslaw K. Szymanski
Rensselaer Polytechnic Institute
Troy, NY
szymab@rpi.edu

ABSTRACT

Time-based habitual behavior is exhibited in humans globally. Given that sleep has such an innate influence on our daily activities, modeling the patterns of the sleep cycle in order to understand the extent of its impact allows us to also capture stable behavioral features that can be utilized for predictive measures. In this paper we show that patterns of temporal preference are consistent and resilient across users of several real-world datasets. Furthermore, we integrate those patterns into large-scale agent-based models to simulate the activity of users in the involved datasets to validate predictive accuracy. Following simulations reveal that incorporating clustering features based on time-based behavior into agent-based models not only result in a significant decrease in computational overhead, but also result in predictive accuracy comparable to the baseline models.

KEYWORDS

Temporal preference, temporal dynamics, human dynamics, behavioral trends, agent-based model, simulation overhead

ACM Reference Format:

James Flamino, WeiKe Dai, and Boleslaw Szymanski. 2019. Modeling Human Temporal Dynamics in Agent-Based Simulations. In *SIGSIM Principles of Advanced Discrete Simulation (SIGSIM-PADS '19)*, June 3–5, 2019, Chicago, IL, USA. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3316480.3322885>

1 INTRODUCTION

For some time now, humans have experienced unprecedented connectivity between friends, family, and even complete strangers thanks to the invention of communicative tools such as the telephone and the internet. As these tools evolve, information is generated and shared in a constant flood between countless users. In an effort to understand the nature of human behavior, researchers have analyzed these interactions and crafted many descriptive features of behavior on social networks. However, a sometimes overlooked characteristic of social network interactions is intrinsic human behavior that is consistently and globally exhibited. One of the less latent of these features is of course a human's need to sleep, which

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SIGSIM-PADS '19, June 3–5, 2019, Chicago, IL, USA

© 2019 Association for Computing Machinery.

ACM ISBN 978-1-4503-6723-3/19/06...\$15.00

<https://doi.org/10.1145/3316480.3322885>

introduces a temporal preference to sequences of the information stream [4] [13]. And despite the seemingly simple nature of sleep cycles in social network analysis at a high level of abstraction, there are multiple challenges that must be addressed to enable a simulator to fine-tune to the presence of temporal preference. These challenges can be represented as the following analytical questions: (i) *How stable are the temporal preference patterns across different mediums, time zones, seasons, and agents?* (ii) *How can these patterns be accounted for in complex systems with multiple agents?* (iii) *What ultimate effects does temporal preference have on simulations?* We address the first of these questions in Section 3, where we perform empirical analysis on two real-world datasets. Section 4 discusses the second question, utilizing temporal preference in agent-based models [5] [10]. Section 5 discusses the third question by validating the results from the previous sections.

2 DATASETS

To illustrate the intrinsic nature of temporal preference across mediums, we choose two essentially different networks for analysis. The first one is created from a Call Detail Record (CDR) dataset [1] [3]. This dataset contains records of the time of outgoing calls and texts for 2300 cellphone users across multiple districts within a three-months time range. Our reasoning for choosing this dataset was the fact that its metadata represent simple calls and texts which are a fairly generalizable type of communication. We primarily use this data as a baseline for model comparisons. The second dataset contains multiple user interactions with GitHub data. This choice was motivated by the fact that GitHub is a social collaborative network, which effectively captures the work life of many people within an online medium. The GitHub data was created by LEIDOS for DARPA's SocialSim Challenge [2]. This dataset contains non-private, anonymized event streams on GitHub for a three year period. There are over 19 million recorded users and over 64 million recorded repositories involved. We use this dataset for the principal simulations that are discussed in Section 5 of this paper.

3 ANALYSIS OF TEMPORAL BEHAVIOR

Hypothetically, the plot of a person's activity should be cyclical, within a 24-hour period. Regardless of activity levels during the day, any productivity at night will lag and diminish as tiredness sets in. This will be reflected in all communicative systems that the person utilizes as they settle down to sleep. This makes sense intuitively, but is this hypothesis verifiable?

3.1 Baseline analysis

To set a reliable paradigm for our analysis of temporal preference, we use our largest dataset: GitHub. An important step in analyzing

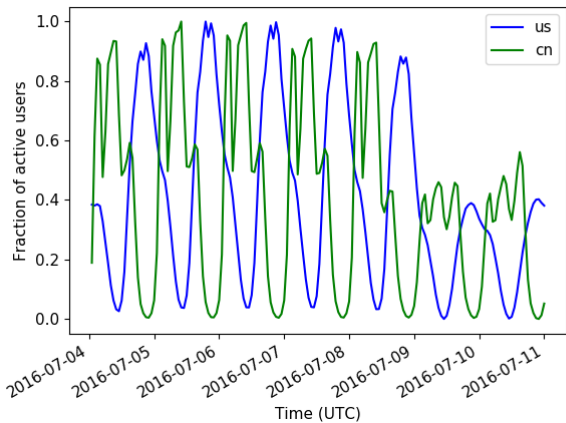


Figure 1: Weekly activity trend for USA and China users for the GitHub dataset

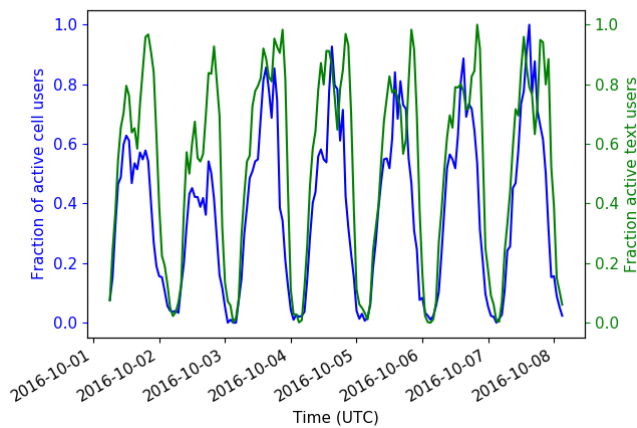


Figure 2: Weekly text and call trend for CDR users within a single time zone

the temporal behavior of users for such a large dataset is to set up conditions for user clustering. Due to an interest in the tools of GitHub, events are generated from users all over the globe, thus time zones must be accounted for. In our dataset, we are able to retrieve the country codes for large number of users in the GitHub dataset and assign them to one of the 243 detected countries. For each country, we gather the number of active users per hour for the entire dataset. A user is considered active for a given hour as long as an action was generated by this user within that defined time range. Following this method for a week for two of the most active countries yields Figure 1. We apply this method to our cell data for a week as well, which is presented in Figure 2.

We observe discernible, cyclical activity trends for the top 87 countries for the GitHub data. In addition to this, a distinctive temporal pattern is visible in both call and text actions. The baseline analysis of these activity trends confirm our initial conjectures, as

there are drops and rises in activity during the respective sleep cycle of users, especially so for the collaborative social network. According to Figure 1 the maximum activity levels for GitHub are reached at around midday, and the minimum activity levels are reached during the night. In addition to this, there is a consistent drop in activity during weekends.

3.2 Consistency testing

After observing consistent activity trends for a majority of the countries in the GitHub data, we check to see if these time series were also all stationary by analyzing the temporal preference patterns for a three-month range. For US users in particular, we find via the Dickey-Fuller test [8] that for 2208 observations (hour unit) for stationarity, the yielded p-value was 0.000029, which indicates that all activity trends were stationary, validating the lack of seasonality for workers. This also agrees with intuition.

Next, we test the stability of the sinusoidal pattern. Using all 2300 available users in the cellphone data we calculate the average number of active users within that set for every hour in a week. This time series acts as a ground truth to the shape of the sinusoidal temporal preference pattern. Dropping N number of users, finding the exhibited time series, and then statistically comparing the normalized shape of this time series to the ground truth can show the resilience of the expected temporal preference pattern. For the comparison we use the Kolmogorov-Smirnov (KS) test [9] and mean absolute error (MAE). The former test shows, given a small enough D statistic, that the two samples must come from the same distribution. The latter test is an error statistic so we can check that the two normalized curves are not dissimilar in pattern directly. As seen in Table 1 the D statistic and MAE are relatively low up until $N = 2000$, and even then the MAE for the two curves remains under 0.1. This, coupled with an average KS p-value of 0.83 (for the null hypothesis that the two samples came from the same distribution) indicates that despite the increasingly shrinking subset of active users, the innate phenomena of temporal preference holds both from the perspective of sample distribution and pattern.

Table 1: Resilience of the general shape of temporal preference for CDR users

Num of users dropped	KS D stat	MAE
10	0.0179	0.0016
100	0.0238	0.0038
1000	0.0655	0.0139
2000	0.2023	0.0489

3.3 Generalized user clustering

As mentioned above, the GitHub analysis done only covers the activity trends for the top 87 countries. In order to harness the complete functionality of user temporal preference for simulation purposes for such a large dataset, the activity trends for the GitHub users under the remaining country codes are assessed. However, it must be noted that of the remaining 156 countries, we exclude 24 completely due to sheer lack of user activity, as the associated users contribute less than 1000 events in the entirety of the GitHub event dataset. The remaining 132 countries have reasonable activity trends, though their time series are not as distinctive or clearly

cyclical as the top 87. We cluster these countries by similar time series shape. Once these clusters were found, an average weekly event time series was formed as an aggregate of all countries within that set. This clustering is useful for the following simulations.

4 INCORPORATING THE BEHAVIORAL TRENDS INTO AGENT-BASED MODELS

Given our empirical analysis of behavioral trends shows features of temporal preference, the questions we must now address is *how can this be accounted for in complex systems with multiple agents?* Below, we describe how we construct our agent simulators and incorporate temporal preference into them.

4.1 Simulation of the cellphone network

Due to the small scale of the cellphone data, we built a simple, generalizable agent-based model (ABM) where agent i at time t samples a Probability Mass Function [11] where discrete random variable X is the number of calls (or text messages) agent i initiates within that timestep. The generation of each agent’s action probability distributions follows a simple replay heuristic where action probabilities are calculated from the agent’s past event counts. This process does not directly simulate agent interactions, but due to the nature of the agents’ empirical behaviors, that dynamic is inherently captured in the system, being exhibited more as a latent feature than a behavioral model. This choice saves on baseline overhead since the agents do not need to store and act upon constantly changing state variables during runtime.

To incorporate the temporal preference of agents within the cellphone dataset into the ABM, we calculate the hourly fraction of active agents for calls and texts separately. These yield cyclical graphs similar to those shown in Figure 2. Additionally, the overall participation level is calculated for each agent by finding the average daily activity level for calls and texts separately, and then ranking them against the other users. This gives each agent a ranked score of how active they are compared to all other agents. During the simulation, at each timestep t , the participation level is compared to the fraction of active agents at the given t . If the agent in question is not active enough to be among the average number of active agents, that agent is skipped without even sampling their action distribution (regardless of respective probabilities).

4.2 Simulation of GitHub

For the dense event stream of the GitHub dataset, the agent-based model that was built follows the generalizable architecture outlined in [14]. By clustering users by activity levels and analyzing user trends (both specific and general) we can retrieve high-level event-type-specific statistics that can be fed into the ABM without any superfluous features or data that could increase overhead. This was done for this particular study, with the ABM being built solely based on statistics retrieved from a separate statistics engine that was fed the bulk of the training data. Inside the ABM, each agent has some behavior model associated with it that determines how it handles the statistics (e.g. a Bernoulli model with $q = 1 - p$ probability of either generating an event or not). Of the models discussed in this study, we utilize the user-centric time-based model, which has the finest time granularity. Specifically, this model iterates through

each user at an arbitrary timestep (defined here as an hour) until completion of a defined time range. Like the CDR ABM, this model does not directly simulate user interactivity, opting to represent it as a latent feature of the behavior model. In order to incorporate temporal preference into the GitHub ABM, we first extract the country codes from the provided dataset and cluster users by those codes using the generalized user clustering techniques. We then define each individual user’s participation level (as calculated in the previous subsection). Following this, we enhance the initialization process of the ABM itself, allowing the engine to tag each agent with their country code and participation level upon creation. During a time-based simulation, a country’s activity trend is sampled for the given timestep to give the fraction of active users for that country. That fraction determines the number of associated users that act at that timestep, given that the participation level ranks of these users rank them higher than an established level of activity. Any other users with that country code are completely skipped.

5 SIMULATION RESULTS AND COMPARISONS

With the specifics and initial conditions for the ABMs for both datasets described, we now turn to the corresponding results of these simulations and analyze the outputs of both the baseline ABM and the ABM with temporal preference incorporated (ABMTP) as illustrated in the previous section. Two natural questions arise when considering the approach for our analysis: (i) *What is the change in the overhead of the simulation?* (ii) *What is the difference between the resultant simulated data?*

5.1 Runtime difference and information loss between CDR simulations

Due to the smaller scale of the cellphone data, it is easy to answer these questions in detail. We answer each question by calculating runtime (rt) difference between our ABMTP and our regular ABM and then comparing the results with a set of statistical measures, respectively. In particular, we compute the RMSE and Spearman’s Rho (r_s) [6] between the resultant data of the ABMTP and ABM to show direct time series similarity and trend correlation between the two simulated curves. We also assemble a general hourly action distribution for both models and compute the Kullback-Leibler (KL) divergence [7] between the two to measure the amount of information lost. For the simulations we run both models for all 2300 users within the entire 3 month range of the dataset. We do this since for this subsection we are interested in a direct statistical comparisons between model types as opposed to predictive computation, which is saved for the GitHub simulations.

Table 2: Runtime and statistical comparisons between the CDR’s ABM and ABMTP simulation results for multiple simulation time ranges

Sim Range (hr)	ABM rt (sec)	ABMTP rt (sec)	RMSE	r_s	KL Div
24	66.95	35.21	69.07	0.9563	0.0076
168	473.4	215.5	819.8	0.9619	0.0178
336	944.4	432.7	800.7	0.9621	0.0158

Table 2 shows the results of the analysis for several simulation time ranges, which reveal that despite the proportional decrease of overhead in the ABMTP, there is little statistical difference between the two simulated time series, further indicating simulation stability for the ABMTP.

5.2 Ground truth comparisons for GitHub

As seen in Table 2, one of the most significant differences between the ABMTP and ABM for the cellphone dataset is the simulation overhead, where the ABM is 2.8 times slower than the ABMTP. For the GitHub data, this observation continues to hold for a set of simulations with 9 million agents, as seen in Table 3. The runtime (rt) difference between the two models for GitHub is substantial as well, where the ABM is 3.1 times slower than the ABMTP.

Table 3: Runtime comparisons between ABM and ABMTP for GitHub’s time-based simulations

Sim Range (hr)	ABM rt (sec)	ABMTP rt (sec)
24	15.21	5.346
168	103.2	33.09
336	205.4	74.15

Now what about statistical accuracy? For the test on the Github dataset, we evaluate against ground truth to gain an understanding of how the two models hold against real-world dynamics. In this test we simulate data to predict the entirety of the last month of a seven-month dataset snapshot. Despite the single month time range, a single simulation of the ABM and ABMTP independently result in about 66 million events due to the large number of active agents. Considering this, direct comparisons of time series is neither computationally efficient nor logical, since the primary goal here is to capture trends and behavior, as opposed to the exact event counts or time series. Considering this, we choose to assess the accuracy of the simulated data via group trends and emergent behavior at a high level. Specifically, we looked at three statistical measures: the Ranked Bias Overlap [12] to compare the top 1000 expected most active users, the average RMSE to compare the activity patterns for the top 1000 most active users, and the KL Divergence of the daily activity distribution (as done with the cellphone dataset). The results are presented in Table 4, comparing the accuracy of all four models when evaluated against the ground truth, indicating that accuracy loss is negligible between the ABM and ABMTP for both the GitHub and CDR datasets.

Table 4: Statistical ground-truth evaluations for GitHub’s ABM and ABMTP simulation results for a one-month simulation time range

Model	RBO	RMSE	KL Div
user ABM	0.6592	43.24	0.0762
user ABMTP	0.6588	43.26	0.0695

6 CONCLUSION

We analyzed two greatly differing datasets in order to find temporal preference of users. We found that the cyclical patterns exhibited in the event streams were consistent, stationary, and highly resilient, indicating strong temporal preference for all users. We incorporated these behavioral trends into agent-based models in order to

capture those intrinsic trends of each agent and use it as a clustering technique to reduce overhead in simulations.

Using the cellphone dataset we found that despite the noticeable speed-up achieved by the enhanced temporal preference agent-based model, the statistical difference between it and the regular agent-based model was negligible. When using the GitHub data to predict a month of user activity of 9 million users, we found that even with the proportional decrease in overhead there was still little loss of accuracy in our ground truth comparison.

Ultimately, our work has shown that temporal preference is indeed an innate behavior of humans that is reflected in communicative media. Utilizing these behavioral trends can yield significant decreases in simulation overhead for agent-based modeling with negligible accuracy loss. These trends can also potentially be used in other methods of simulation aside from computational efficiency, such as feature clustering and behavioral modeling. Another venue for future work is the expansion of the agent-based models to account for dynamic inter-agent dependencies in order to model phenomena such as information cascades.

ACKNOWLEDGMENTS

This work was sponsored in part by DARPA under contract W911NF-17-C-0099, the Army Research Laboratory (ARL) through the Cooperative Agreement W911NF-09-2-0053 (NS CTA), and the Office of Naval Research (ONR) under Grant N00014-15-1-2640. The views and conclusions contained in this work are those of the authors.

REFERENCES

- [1] Vincent D Blondel, Adeline Decuyper, and Gautier Krings. 2015. A survey of results on mobile phone datasets analysis. *EPJ Data Science* 4, 1 (2015), 10.
- [2] DARPA. 2017. Computational Simulation of Online Social Behavior (SocialSim). <https://www.darpa.mil/program/computational-simulation-of-online-social-behavior>
- [3] Casey Doyle, Zala Herga, Stephen Dipple, Boleslaw K. Szymanski, Gyorgy Kormiss, and Dunja Mladenic. 2019. Predicting complex user behavior from CDR based social networks. arXiv:1903.12579
- [4] Alceu Ferraz Costa, Yuto Yamaguchi, Agma Juci Machado Traina, Caetano Traina Jr, and Christos Faloutsos. 2015. Rsc: Mining and modeling temporal activity in social media. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 269–278.
- [5] Nigel Gilbert. 2008. *Agent-based models*. Number 153. Sage.
- [6] Jan Hauke and Tomasz Kossowski. 2011. Comparison of values of Pearson’s and Spearman’s correlation coefficients on the same sets of data. *Quaestiones geographicae* 30, 2 (2011), 87–93.
- [7] Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics* 22, 1 (1951), 79–86.
- [8] James G. MacKinnon. 1994. Approximate Asymptotic Distribution Functions for Unit-Root and Cointegration Tests. *Journal of Business & Economic Statistics* 12, 2 (1994), 167–176. <https://doi.org/10.1080/07350015.1994.10510005>
- [9] Frank J Massey Jr. 1951. The Kolmogorov-Smirnov test for goodness of fit. *Journal of the American statistical Association* 46, 253 (1951), 68–78.
- [10] Steven F Railsback, Steven L Lytinen, and Stephen K Jackson. 2006. Agent-based simulation platforms: Review and development recommendations. *Simulation* 82, 9 (2006), 609–623.
- [11] William J Stewart. 2009. *Probability, Markov chains, queues, and simulation: the mathematical basis of performance modeling*. Princeton University Press.
- [12] William Webber, Alistair Moffat, and Justin Zobel. 2010. A similarity measure for indefinite rankings. *ACM Transactions on Information Systems (TOIS)* 28, 4 (2010), 20.
- [13] Jaewon Yang and Jure Leskovec. 2011. Patterns of temporal variation in online media. In *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 177–186.
- [14] Shuocho Yao, Yifan Hao, Dongxin Liu, Shengzhong Liu, Huajie Shao, Jiahao Wu, Mouna Bamba, Tarek Abdelzaher, James Flamino, and Boleslaw Szymanski. 2018. A Predictive Self-Configuring Simulator for Online Media. In *Winter Simulation Conference (WSC)*. IEEE, 1262–1273.