2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining

Mechanistic and Data-Driven Agent-Based Models to Explain Human Behavior in Online Networked Group Anagram Games

Vanessa Cedeno-Mieles[†§], Zhihao Hu[‡], Xinwei Deng[‡], Yihui Ren[#],
Abhijin Adiga[*], Christopher Barrett[*], Saliya Ekanayake[*], Gizem Korkmaz[*], Chris J. Kuhlman[*], Dustin Machi[*],
Madhav V. Marathe[*], S. S. Ravi[*], Brian J. Goode[&], Naren Ramakrishnan[¶†], Parang Saraf[¶†], Nathan Self[¶†]
Departments of Computer Science[†] and Statistics[‡], Discovery Analytics Center[¶], Biocomplexity Institute[&]
Virginia Tech - Blacksburg, VA
Escuela Superior Politécnica del Litoral, ESPOL - Guayaquil, Ecuador[§]
Brookhaven National Laboratory - Upton, NY[#]
Biocomplexity Institute, University of Virginia - Charlottesville, VA[*]

Noshir Contractor
Department of Industrial Engineering and Management Sciences
Northwestern University
Evanston, IL

Joshua M. Epstein
Department of Epidemiology, NYU
New York, NY
Santa Fe Institute
Santa Fe, NM

Michael W. Macy
Department of Sociology
Cornell University
Ithaca, NY

***Abstract*—In anagram games, players are provided with letters for forming as many words as possible over a specified time duration. Anagram games have been used in controlled experiments to study problems such as collective identity, effects of goal-setting, internal-external attributions, test anxiety, and others. The majority of work on anagram games involves individual players. Recently, work has expanded to group anagram games where players cooperate by sharing letters. In this work, we analyze experimental data from online social networked experiments of group anagram games. We develop mechanistic and data-driven models of human decision-making to predict detailed game player actions (e.g., what word to form next). With these results, we develop a composite agent-based modeling and simulation platform that incorporates the models from data analysis. We compare model predictions against experimental data, which enables us to provide explanations of human decision-making and behavior. Finally, we provide illustrative case studies using agent-based simulations to demonstrate the efficacy of models to provide insights that are beyond those from experiments alone.**

I. Introduction

A. Background and Motivation

In one form of an **individual anagram game**, a player is provided with a set of alphabetical letters to form as many words as possible in a prescribed time duration. The performance of a player is often quantified based on the number of words formed.

In a **group anagram game** (GrAG), multiple players collaborate. Each player is given letters and forms words with her own letters, and can share letters with her neighbors to enable everyone to form more words. Figure 1 provides a schematic of a 3-player GrAG. Each player (v_1, v_2, and v_3) is initially provided with $n_l = 3$ letters as shown. A player may form words, and through the communication channels in gray, may request letters and reply to letter requests.

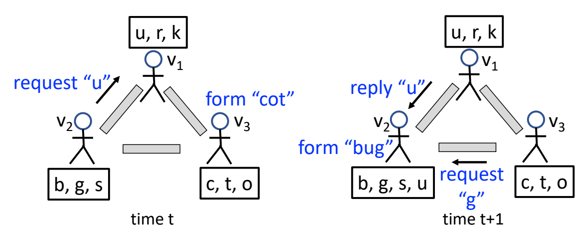

Fig. 1. Simplified view of a networked group anagram game (GrAG), with illustrative actions among $n = 3$ players that communicate and share letters through the gray channels. Each player is initially given $n_l = 3$ letters. Letters that a player has "in-hand" to form words are shown in boxes. Player actions are shown in blue. At time t, v_2 requests a "u" from v_1 and v_3 forms the word "cot." At the next time, v_2 receives a "u" from v_1, forms the word "bug," and receives a request from v_3.

Overwhelmingly, research on anagram games considers the individual setting. It has been extensively studied (over 20 published works) for more than 60 years to analyze phe-

ASONAM '19, August 27-30, 2019, Vancouver, Canada

ACM ISBN 978-1-4503-6868-1/19/08
http://dx.doi.org/10.1145/3341161.3342965

nomena such as goal-setting, compensation types, internal-external attributions, and test anxiety (e.g., [1], [2]). Other names for anagram game are *word formation game* and *word construction game*.

There are several reasons to study GrAGs. The research in [3] used them to study experimentally the formation of collective identity (CI), defined in social psychology as an individual's cognitive, moral, and emotional connection with a broader community, category, practice, or institution [4]. A second motivation is their relevance to other types of group dynamics, notably intergroup and intragroup cooperation and competition (e.g., [5]). A third motivation is that many of the phenomena listed above for the individual anagram game (e.g., goal-setting) could be studied in group settings with models of group behavior.

Overall, researches involving anagram games encompass a broad range of disciplines like sociology, economics, management science, and (social) psychology [1], [2], [6]. It is clear that using anagram games is valuable in various fields of research. The first and only work on modeling GrAGs was recently completed [7]. We enumerate the differences between our work and [7] in Section I-B immediately below.

B. Our Work Scope and Differentiators from Previous Work

Work scope. Our work starts with data from online social network GrAGs. (The game platform and online experiments are *not* the focus in this work.) With these data: (i) data analytics are performed to support model development; (ii) different models for different player actions in the GrAG are developed; (iii) the models are evaluated against experimental data; and (iv) these models are then recast as agent-based models and executed within an agent-based modeling and simulation (ABMS) platform to produce computational results that go beyond the experiments.

Based on this work scope, all of the following are completely different in this work, compared to that in [7]: data analytics, the aspects of the game that are being modeled, the types of modeling techniques used, the models themselves, and the quantities that the models predict. We address particular differences between [7] and our work now.

Work in Ref. [7]. The subject of [7] is the *action **type** and time (ATAT) model*, which uses multinomial logistic regression to build the model. In that work, the goal was to develop models to predict the *type* of action taken in time, e.g., predictions of the form: player v_i takes action type "form word" at time t. Also, if a player action is form word, and the player has letters that cannot form a word (e.g., letters q, z, and r) then that model will nevertheless form an unspecified (unrealistic) word from these letters. Moreover, the models of [7] do not consider the particular letters assigned to players in a game. Consequently, all player behaviors will tend toward the same mean behavior in agent-based simulations (ABSs).

Our work. In contrast, our work focuses on three *component models*. Different models are developed for the actions "form word," "request letter," and "reply to (letter) request." Our models account for network structure, letter assignments and letters in-hand (i.e., letters that a player has to form words), and particular player parameter assignments—all of which can vary among players—so results will remain distinct across agents. That is, we capture heterogeneity in several ways.

Our ABMS framework uses a **composite model**: a combination of the ATAT model (to determine what action types players take in time) and the three component models developed herein (to predict the specifics of each action). The composite model is our agent-based model (ABM). This ABMS system simulates GrAG scenarios beyond those of the experiments.

C. Novelty of Our Work

First, our work is an exemplar of a detailed procedure for combining mechanistic and data-driven models to form single models of human *decision-making* that output human *actions* in a game. *Mechanistic models*, for our purposes, have the following characteristics: (i) the models are based on first principles and are not tied to any particular domain; and (ii) the models are specified, implemented, and executed without any experimental data. To augment mechanistic models by accounting for variability in player behaviors, *data-driven models* are constructed from analyses of experimental data. Second, because we prove that the mechanistic models capture player behavior, these models *explain* behaviors, as described in our contributions below. Third, our mechanistic models are novel: Levenshtein Distance (LD) [9] (see Section IV-A) and a greedy optimization procedure describe human decision-making and have not been used in anagrams contexts (we could not find LD used in any modeling of human behavior, as we do here). As called for in the social sciences, our focus is on model construction and predictions, and explanations of human behavior [10], [11].

D. Contributions

1. A process for combining mechanistic and data-driven approaches to build models of human decision-making. We provide the details of our process in Section IV. See Figure 2. First, mechanistic models are conjectured and evaluated by comparing their predictions to experimental data. This does three things: (i) enables comparisons of model predictions with experimental data, and if these comparisons are favorable (which they are), then (ii) the structures of the models, and the mechanisms embedded in them, provide *explanations* for human decision-making [12], [13], and (iii) the mechanistic models form the basis of the ABMs. Second, because the mechanistic models can be improved by including data from experiments, we use data-driven modeling approaches to introduce stochasticity to account for variability across human subject game players. Hence we utilize these two modeling approaches in a well-defined process.

2. Mechanistic models. We use concepts such as LD, word corpora, word proximity networks (WPNs), and a greedy optimization algorithm (all defined in Section IV) to develop mechanistic models for two of the three player actions (see Figure 2). The LD model, used for word formation, could be used within any agent that is required to form words, and

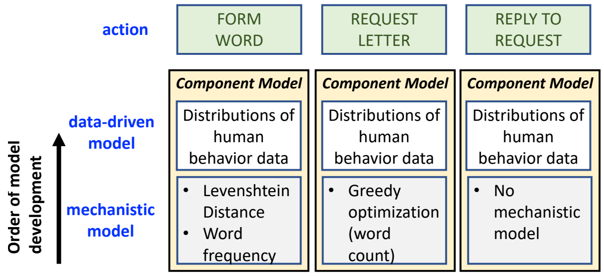

Fig. 2. **Component models** (i.e., combined *mechanistic* and *data-driven* models) for the three player actions in the GrAG. These are models of human decision-making, which output specific player actions in the game. The particulars of the mechanistic and data-driven models are given in the respective boxes under the actions and are detailed in Section IV. Mechanistic models are built first, and then augmented with data-driven models.

the greedy optimization algorithm, used for requesting letters, could be used by agents to make a choice from among a finite set of options. That is, these models are not tied to our GrAG. But the next contribution presents their utility within the GrAG.

3. New experimental findings and explanations of player behaviors based on cognitive and economic theories. The analyses focus on data for three types of player actions: (1) form a word; (2) request a letter; and (3) reply to a letter request. See Figure 2. A summary of some explanations follows. A word w_2 that a player forms is explained by considering (i) the letters that the player has in-hand (i.e., in her possession) and (ii) LD [9] between the most recently formed word w_1 and the next word to be formed w_2 from a candidate set of words (Section IV-B). This is motivated by, and consistent with, cognitive load theory [14] in that people try to reduce cognitive load during learning. Here, the closer the next word formed is to the previously formed word—as measured by LD—the lesser the cognitive load in forming a new word. For letter requests, we use the idea that player action is based on rational choice theory [15]. Our analyses (Section IV-C) demonstrate that the letter that a player requests from her neighbors is explained by identifying the letter that maximally increases the number of words that the player can form, when also considering the letters that the player has in-hand (greedy optimization algorithm). This behavior is consistent with rational choice theory. This is because players' earnings in games are proportional to the number of words formed, so it is rational for a player to choose a letter to maximize the size of their candidate word set. It is interesting that our explanation means that players are reasoning beyond more naive approaches, such as simply requesting some "most frequently" used letter (e.g., preferring e over z). (We have modeled this naive approach—results not shown here—and this model's results are not consistent with the data.) Also, the experimental data clearly show that players do not request all the available letters at the outset of a game. Rather, they request letters throughout the game as they identify use for them. Finally, we also show that there are four types of behavior in replying to letter requests (Section IV-D).

4. Agent-based models and results. A family of ABMs are developed, yielding a composite model, where each ABM is comprised of a distinct model for each of the three actions, with user-specified parameter values for player/agent characteristics, such as the agent's vocabulary and their aptitude, i.e., the degree to which they perform optimally. The multi-logit regression model based on [7] is adopted to determine which action type each agent selects at each discrete time in a simulation (time granularity is seconds). The selected action type then determines the appropriate model developed herein to predict details of the action. Note that there is a fourth action, a no-operation (no-op), where the agent does nothing at particular times, which represents agent thinking and requires no model. We also provide new insights from exercising the ABMs (see Section V), such as demonstrating how player performance decreases with decreasing player aptitude and the effects of heterogeneous initial letter assignments to players.

II. Related Work

By far, the most relevant study to our work is the modeling in [7], which is **agent-based modeling of anagram games**. To the best of our knowledge that is the only work prior to ours that models the GrAG [7]. That work was discussed in detail in relation to our work in Sections I-A and I-B. We now address other topics related to our work.

Anagram experiments. Over 20 experiment works (e.g., [1], [2]) use *single player* anagram games. The only cooperative GrAG, which is *face-to-face*, is reported in [3]. The game is used to foster CI among teammates.

Networked experiments and modeling. There are several other online (e.g., [16]) and in-person (e.g., [3]) experiments with interacting participants that can be represented as networks, and analyses of network populations (e.g., [17], [18]), where edges represent interaction channels.

Mechanistic and data-driven modeling. Several works use AI methods and data to model behavior (e.g., tutoring and learning [19]). Also, neuroscientists are using neuro-imaging to understand human decision-making; [20] discusses optimization methods, such as the one we use in the model for requesting letters.

Explanatory modeling. There are many works (e.g., [12], [13]) that describe different definitions of *explanations*, different types of explanations that models provide, and procedures for arriving at explanations. We follow ideas from [12], [13]: that the structure of mechanistic models that adequately predict human behavior can be used to explain behavior.

III. Online Social-Networked Group Anagram Game

We built a customized web application (web app) for an online GrAG. Players recruited through Amazon Mechanical Turk (MTurk), are provided game instructions, participate in the GrAG through their web browsers, and are paid based on their performance. A total of 48 experiments were performed using a total of 367 players, with numbers of players per game ranging from 3 to 17. The game duration is 5 minutes. In the following, we describe the GrAG/experiment.

Figure 1 provides a description of the game setup and actions. A game begins with n players, v_1 through v_n. Each player has a degree d that specifies the number of connections to other players. A connection (edge) between two players denotes a communication channel where a letter ℓ can be requested and sent (sending a letter is a reply). Thus, an experiment configuration is a graph $G(V, E)$ with player set V and communication channels E. In experiments, G is a k-regular random graph ($k \equiv d$), with uniform degree $2 \leq k \leq 8$. Each player starts the game with n_ℓ initial letters, which they can use to form words or share among their neighbors, when requested. At the beginning of a game, a word corpus C^W is defined with a list of words a player can form during the game. For this we use a list of the top 5000 words from the 450 million word Corpus of Contemporary American English, the only large and balanced corpus of American English [22]. The three major player actions in a game are now described.

Player action: forming a word. At any point during a game, a player v_i can form a word w_i. All letters in the word w_i must come from the set of letters v_i has in-hand L_i^{ih} (superscript ih). A single letter ℓ in L_i^{ih} can appear any number of times in a word. For a word submission to be accepted in the game, the word has to be in the word corpus C^W. The C^W is specified, but it is not provided to players. Rather, players have to recognize possible words that can be formed from the letters they have. The C^W is the same for all players in all games. A player can submit a word only once; multiple players can form the same word.

Player action: requesting a letter. At any point during a game, a player v_i can request a letter ℓ_{ij}^{req} from a neighbor v_j's set of n_ℓ *initial* letters L_j^{init}. The anagram game screen shows all neighbors' *initial* letters as available for request. A letter received by v_i is put into the set L_i^{ih}.

Player action: replying with a letter. At any point during a game, a player v_i can reply with a letter ℓ_{ij}^{rep} to a neighbor v_j's request (ℓ_{ij}^{rep} must be in L_i^{init}). The anagram game screen for v_i shows all of the letters requested of v_i.

To encourage cooperation, any letter in L_i^{ih} can be used any number of times in forming words, and the letter is not lost; the letter bestows an infinite supply of use. Similarly, if v_i requests a letter ℓ from v_j, and v_j replies with it, v_j still retains a copy of the letter and can use it. Also, earnings for the team are based on the total number of words formed, and all players receive $(1/n)$ of the total earnings. Typical player earnings are \$7 to \$10 per game.

IV. Data Analysis and Model Development

Figure 2 provides the roadmap for building the models for the three player actions, which is the focus of this section. Ultimately, our goal is to use these models as ABMs in an ABMS framework to study GrAGs well beyond those of experiments.

For each action—which is a component model of the ABM—we provide: (i) our premise for understanding player behavior and the key concepts for this premise, (ii) experimental analyses and results for these key ideas that construct and justify (i.e., give evidence for) the component model of the composite ABM, and (iii) a formal algorithm for the component model for the action in Figure 2. Note that the steps of algorithms that we specify below are not focused on efficient implementation, but rather on conveying the steps of the algorithms as they relate to the data analyses. First, we address preliminaries.

A. Preliminaries

We introduce two concepts used in data analysis and modeling. **Levenshtein distance** (d^L) [9], an edit distance, is prominent in our work and the work's novelty, and is motivated by work in linguistics and bioinformatics [21]. It quantifies the difference in letters between two words. In starting with one word to obtain a second word, a letter substitution counts as one, as does each of letter insertion and letter deletion. Hence, going from *had* to *hats* requires $d^L = 2$: one to substitute t for d and one for inserting an s.

A **word proximity network** (WPN) is a clique graph $H(V_H, E_H)$ where vertices V_H are words that can be formed, according to a word corpus C^W, with the letters that a player currently has in-hand and E_H is the set of edges between pairs of words, labeled with the d^L between the two words.

B. Player Action: Form Word

Basic premise, assumptions, and key concepts. We seek to identify a method that explains the process of players selecting words to form. Our premise is that given the last word w_1 that v_i has formed, the next word w_2 that v_i will form will be one with minimal d^L from w_1 because this requires a minimal number of letter manipulations (i.e., lesser cognitive load [14]). For the first word, v_i selects the most frequent word from the corpus that can be formed with its letters in-hand L_i^{ih}. (The word corpus provides the frequency of occurrence of each word.) We note that for each player v_i, there is a set L_i^{ih} of letters that she has in-hand and a corresponding set $W_i^{ih} \subseteq C^W$ of words that v_i can form from the entire corpus C^W of words, based on the letters in L_i^{ih}. As v_i requests and receives more letters from her neighbors, the cardinalities of L_i^{ih} and W_i^{ih} will (typically) increase. Also note that for a given word w_1 formed by v_i in a game, W_i^{ih} can be partitioned based on $d^L(w_1, w_2)$ for fixed w_1 and for each $w_2 \in W_i^{ih}$ using the WPN. Let $W_i^{ih}(w_1, d^L) \subseteq W_i^{ih}$ be the set of words at d^L from w_1 that v_i can form.

Our data analysis is based on two central ideas, for each player v_i. First, we compare d^L values between two consecutive words formed (w_1 and then w_2), both the actual value $d^L_{i,act}(w_1, w_2)$ measured from experiments and the optimal (i.e., minimal) value of d^L, denoted $d^L_{min}(w_1, w^*)$, for some w^* in W_i^{ih} that is at a minimum LD from w_1. Both d^L values are based on v_i's set L_i^{ih}. (We drop the arguments when they are obvious from context.) Second, for a given set of words at some d^L from w_1, denoted $W_i^{ih}(w_1, d^L)$, we select w_2 based on the popularity of words as provided by the rank (frequency of use) from [22]. All of these parameters are either inputs (e.g., C^W), measured in experiments, or computed from experimental data. These high-level steps enable us to understand players' behavior in forming words, as described next.

Data analysis. Analysis step 1. For each player v_i in a game, we consider pairs of consecutive words formed, (w_1, w_2). From this, we compute $d^L_{i,act}(w_1, w_2)$, the actual d^L. Also from these data and from L^{ih}_i at the time w_2 was formed, we can compute d^L_{min} and the word set $W^{ih}_i(w_1, d^L_{min})$. We compute $\Delta d^L = d^L_{i,act} - d^L_{min}$. A value of zero means that the player is performing optimally according to our premise; a value > 0 means that v_i is performing suboptimally—v_i is making more letter edits (expending greater effort) than is required by the data.

We rank the players by their average Δd^L, Δd^L_{ave}, over all pairs of words (w_1, w_2) that they form in a game. We partition the ranking of players into five equi-sized bins, P_1 through P_5, such that players in P_1 (resp., P_5) have the smallest (resp., largest) values of Δd^L_{ave}. That is, the players in P_1 perform closest to optimal. A player v_i's aptitude b^{wf}_i in forming words takes a value from P_1 through P_5. We take this player-centric approach because we want to produce agent models based on individual player and groups of players' behaviors.

Analysis step 2. For each of the five groups of players P_j $(1 \leq j \leq 5)$, we plot all data points $(x, y) = (d^L_{min}, d^L_{i,act}(w_1, w_2))$ for each person in that group, in Figure 3. In each plot, for each d^L_{min} on the x-axis (the *mechanistic model prediction*), there is a range of $d^L_{i,act}(w_1, w_2)$ (from the data) for all v_i in a particular 20% bin. If we break the players down into 10% bins (instead of the 20% bins), the top 30% of players perform such that the median value of $d^L_{i,act}(w_1, w_2)$ equals d^L_{min}. That is, in a median sense, these top 30% of players form words w_2 such that $d^L_{i,act}(w_1, w_2) = d^L_{min}$, and hence w_2 is formed optimally (i.e., according to the mechanistic model). Moreover, if we look at the top 80% of players, then $d^L_{min} \leq d^L_{i,act}(w_1, w_2) \leq d^L_{min} + 1$. That is, for 80% of all players, the pairs of consecutive words, (w_1, w_2), produce $d^L_{i,act}$ values that differ by at most 1 from the d^L_{min}. *These data for* $|C^W| = 5000$ *substantiate our premise that players form word* w_2 *based on* d^L. Although not shown, similar results are generated for $|C^W| = 1000$, 2000, 3000, and 4000, if we take these sets as the 1000, 2000, 3000, and 4000 most frequently used words in the original corpus of 5000 words.

Analysis step 3. For each box plot in Figure 3, we form a frequency distribution $\mathcal{D}^{d^L}$ as a function of the triple $(C^W_i, b^{wf}_i, d^L_{min})$. Figure 4 provides one such distribution. So, given a C^W_i, an aptitude b^{wf}_i for forming words, and a d^L_{min}, one can sample an actual LD, $d^L_{i,act}$, in forming w_2 from w_1.

Analysis step 4. For a given w_1 and $d^L_{i,act}$, $W^{ih}_i(w_1, d^L_{i,act}) \subseteq W^{ih}_i$ is the candidate set of words that v_i can form as w_2. The issue is how players extract a particular word from $W^{ih}_i(w_1, d^L_{i,act})$ as w_2. Figure 5 provides the answer. For each v_i, we rank the words in $W^{ih}_i(w_1, d^L_{i,act})$

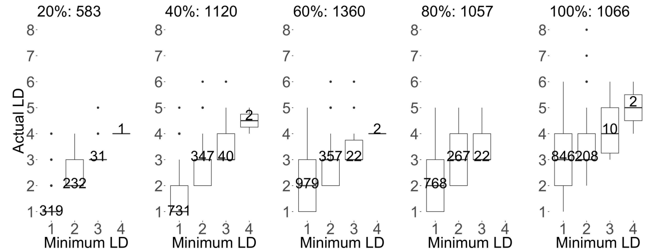

Fig. 3. Comparison of ***mechanistic*** model predictions against data for the *form word* model. Mechanistic predictions are the values on the x-axis (d^L_{min}); data are on the y-axis ($d^L_{i,act}$). We use the $|C^W| = 5000$ word corpus. Each plot corresponds to a grouping of players by 20% bins of player performance in forming words according to d^L, and represents, in turn, P_j, $j \in \{1, 2, 3, 4, 5\}$, moving left to right. Numbers are numbers of observations in the data. If $d^L_{i,act}(w_1, w_2) = d^L_{min}$, then the experimental data correspond exactly with the mechanistic model.

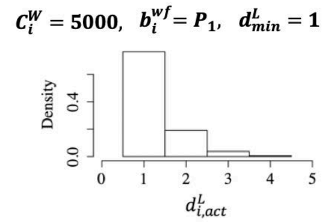

Fig. 4. For $(C^W_i, b^{wf}_i, d^L_{min}) =$ (5000 words, P_1, 1), the distribution $\mathcal{D}^{d^L}$ of $d^L_{i,act}$ from experiments is shown. For a given d^L_{min} computed for optimal behavior, the appropriate distribution is sampled to obtain $d^L_{i,act}$ for v_i. These distributions are formed from the data in Figure 3 and they are part of the ***data-driven*** model of form word.

in decreasing order of frequency of occurrence (which is obtained from the word corpus itself), such that the first ranked word is the most frequently used word. This plot shows the number of times the chosen word w_2 is of a particular rank. It is clear that players select w_2 based on the frequency of the word's use, e.g., the top-ranked word is selected almost 700 times from the corpus. This result also holds over different corpus sizes from 1000 to 5000 words. These data support our use of a mechanistic model of selecting the word with highest frequency of use in a word corpus from the candidate set of words.

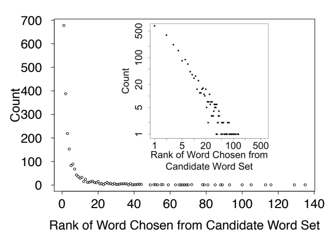

Fig. 5. Experimental data for $|C^W_i| = 5000$. Log-log scale plot (inset) of the distribution of ranks of words formed by players from the word set $W^{ih}_i(w_1, d^L_{i,act})$. Lesser rank means higher word frequency from corpus. Players most often choose words with lesser rank (i.e., greater frequency).

Remark: These data analyses substantiate our claim that our models are explanatory. The data are consistent with the explanation that humans reason about what word to form using LD and word frequency (familiarity), consistent with cognitive load theory [14].

Remark: It is emphasized that players in the experiments are not given a word corpus, frequency of letter use, d^L concepts and values, etc. Our construction and procedures presented here are our representation of the mental decision-making processes that players engage in, resulting in human behavior in the form of detailed actions. In experiments, players are

Input: Agent $v_i \in V$. Agent word-forming aptitude b_i^{wf}. Word corpus or vocabulary C_i^W for v_i. Letters in-hand L_i^{ih}. Most recent word formed by v_i, w_1. Words W_i^f formed up to now by v_i. Distribution $\mathcal{D}^{wr}$ of word frequencies from C_i^W and distribution $\mathcal{D}^{d^L}$ of $d_{i,act}^L$ frequency as a function of tuple $(C_i^W, b_i^{wf}, d_{min}^L)$.

Output: Next word w_2 that v_i forms, if any.

Steps:

1) From letters in-hand L_i^{ih}, construct the set W_i^{ih} of words that v_i can form (and that v_i has not yet formed). Set $V_H = W_i^{ih}$ and let H be the WPN network induced by V_H. Let the edge set be E_H, with edge labels of d^L.
2) If V_H is empty, terminate algorithm and return no word.
3) From the values of the edge labels $d^L(w_1, w_j)$, for all edges $\{w_1, w^*\} \in E_H$ of WPN H, where $w_1, w^* \in V_H$, determine the minimum LD, d_{min}^L.
4) For the triple $(C_i^W, b_i^{wf}, d_{min}^L)$, sample from the distribution $\mathcal{D}^{d^L}$ to obtain the actual LD, $d_{i,act}^L$, that v_i will use to form the next word. (Example provided in Figure 4.)
5) From the set $W^{d_{i,act}^L} \subseteq V_H$ of words at $d_{i,act}^L$ from w_1, order the words from most frequently used word to least (C^W provides this ranking).
6) From the frequency distribution $\mathcal{D}^{wr}$ of words in $W^{d_{i,act}^L}$, draw a rank r_i of a word. Select the unique word w_2 that corresponds to rank r_i. Return w_2.

Fig. 6. Steps of the Algorithm FORM WORD. This algorithm returns a word that an agent forms.

only given letters and the ability to share them. This remark holds for the next two models, too.

Algorithm for form word. The algorithm is in Figure 6, and follows directly from the above data analysis. This is cast as the *agent* model in the ABMS.

C. Player Action: Request Letter

Basic premise, assumptions, and key concepts. Our goal is to uncover a process that explains how players select the next letter to request from their neighbors. Our premise is that player v_i will select the next letter to request as the letter from the set of candidate neighboring letters L_i' that produces the greatest increase in the number of words that v_i can form. The key idea is to examine each candidate letter ℓ and determine the number of *new* words $|W_i^{ih\ell}|$ that can be formed with existing letters in L_i^{ih} and the requested letter combined (this word set is $W_i^{ih\ell}$), rank these letters in decreasing order of $|W_i^{ih\ell}|$, and select the letter to request based on this ranking. This is a greedy process—in the sense of selecting the best letter (i.e., the letter that ranks first), one at a time—and is our mechanstic model. This is a rational choice approach [15] because players are incentivized to form as many words as possible, so it is rational to select a letter that maximally increases the number of words that can be formed. Note that as more letters have been requested and received, the number of letters to request, $|L_i'|$, decreases because once a player has a letter, she can use it any number of times. We now provide the evidence for behavior that is aligned with this premise.

Data analysis. Analysis step 1. We rank all players by their performance in requesting letters in the GrAG, as follows. For each v_i, and for each actual letter request, we rank the candidate letters to request in L_i' according to our greedy model (given immediately above), and then identify the rank $r_{i,act}$ of the letter $\ell_{i,act}$ actually requested. Then we compute an average rank of letter requests $r_{i,ave}$ for each v_i, over the first 1/2 of all v_i's requests. We use only the first 1/2 of requests in computing $r_{i,ave}$ because as $|L_i'|$ decreases, the selected rank and the top-ranked letters will be more closely aligned because there are so few letters left. Hence, in order to not bias the results, we use only the first 1/2 of letter requests. The players v_i are ranked by $r_{i,ave}$, smallest to largest value, and the players are partitioned into five equi-sized bins Q_1 through Q_5, where players in Q_1 (resp., Q_5) select letters to request that are most (resp., least) conformant to our mechanistic model. A player v_i's aptitude b_i^{req} in requesting letters takes a value from Q_1 through Q_5. This partitioning is to ensure a sufficient number of observations for each bin. Again, we partition based on players because we want to develop agent behaviors based on player behavior.

Analysis step 2. We analyze each Q_j, $j \in \{1,2,3,4,5\}$, separately, as follows. We take each $v_i \in Q_j$, note each rank $r_{i,act}$ corresponding to each letter request in the first 1/2 of requests, count the number of occurrences of the ranks of each requested letter, and sum the counts over all players. Results are shown in the left-most plot of Figure 7 for $b_i^{req} = Q_1 = 20\%$. (Note that player v_i's aptitude b_i^{req} in requesting letters may take values Q_1 through Q_5.) These data are compared against our mechanistic model (in green), which predicts all letter requests will be of rank 1 in this plot. Note that for the $b_i^{req} = Q_1 = 20\%$ data, the number of occurrences of a selected rank generally increases as the rank decreases, though the effect is sometime less pronounced for some cases. We claim that the data support our premise, i.e., our model explains the data. Players request letters that generate the greatest increase in the words that they can form.

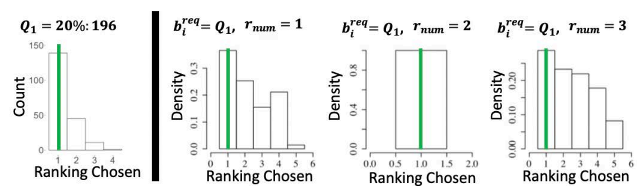

Fig. 7. Comparison of ***mechanistic*** model predictions (in green) against data (the distributions) for the *request letter* model. Our mechanistic model predicts all letter requests will be of rank-1 in each of the four plots. (LEFT) Experimental data are for the 5000-word corpus, aptitude $b_i^{req} = Q_1 = 20\%$ for letter requests (plots for Q_j, $j \in \{2,3,4,5\}$ are not shown). For aptitude Q_1, the frequency of the rank of the chosen letter is plotted. These data show that players most often choose letters with lower rank, meaning that they choose letters that can form relatively more words. (RIGHT) These three plots break down the left-most plot by showing distributions for different request numbers r_{num} of 1, 2, and 3, by v_i. These distributions $\mathcal{D}^{\ell r}$ are used to sample $r_{i,act}$ based on $(C_i^W, b_i^{req}, r_{num})$.

Analysis step 3. We break down each plot of the type in Figure 7, at the left, to account for C_i^W, b_i^{req}, and the number r_{num} of the letter request in the three right-most plots of the figure. By sampling from frequency distributions $\mathcal{D}^{\ell r}$ based on $(C_i^W, b_i^{req}, r_{num})$ for v_i, we obtain the rank of the actual letter

Input: Agent $v_i \in V$. Agent letter requesting aptitude b_i^{req}. Word corpus C_i^W. Letters in-hand L_i^{ih}. The set L_i' of letters that v_i's neighbors were initially assigned that v_i has not yet requested; this is the candidate set of letters to request. The request number r_{num}. Distributions $\mathcal{D}^{\ell r}$ of letter ranks for triples $(C_i^W, b_i^{req}, r_{num})$.

Output: Next letter ℓ^* that v_i requests, if any.

Steps:

1) If L_i' is empty, terminate and return no letter.
2) For each candidate letter to request $\ell \in L_i'$ that has yet to be requested, determine the *new* words $W_i^{ih\ell}$ that can be formed from C^W with the letters in set $L_i^{ih} \cup \{\ell\}$ (include only words that have not yet been formed).
3) If every word set $W_i^{ih\ell}$ for all ℓ is empty, remove an arbitrary letter ℓ^* from L_i', terminate this algorithm and return ℓ^*.
4) Rank the letters $\ell \in L_i'$ in decreasing values of $|W_i^{ih\ell}|$. Let $r(\ell)$ be the rank of ℓ.
5) Determine the rank $r_{i,act}$ of the letter to select for requesting by sampling from distribution $\mathcal{D}^{\ell r}$ using as input $(C_i^W, b_i^{req}, r_{num})$. (See Figure 7 for three examples.)
6) Select the letter ℓ^* such that $r(\ell^*) = r_{i,act}$. Break ties arbitrarily. Remove ℓ^* from L_i'. Return ℓ^*.

Fig. 8. Steps of the Algorithm REQUEST LETTER. This algorithm returns a letter that an agent requests.

requested $r_{i,act}$ in the model. This provides finer modeling granularity by accounting for the number of the letter request.

Algorithm for request letter. The algorithm is in Figure 8 and follows directly from the data analysis just presented. This algorithm is presented in the form of an *agent* model.

Remark: These analyses and data provide evidence for our claim that this model is explanatory. Players generally request letters by (roughly) maximizing the increase in number of words that they can form, which follows rational choice theory [15].

D. Player Action: Reply to Letter Requests

Unlike the previous two models, this model is purely data-driven. For space reasons, we provide an abbreviated description here.

Basic premise, assumptions, key ideas, and data analysis. The goal is to produce a model that explains how players respond to letter requests from their neighbors. The basic premise is that players can be partitioned into categories of behavior. We determined from the data these four categories: (1) those players that respond to all queued (pending) letter requests in their buffer (called FB for full buffer); (2) those that respond to some fraction of all pending letter requests in their buffer (called LTFB for less than full buffer); (3) those that sometime behave as FB and sometimes as LTFB (called Mixed); and (4) those that never reply to letter requests (called NR). The key ideas are that for each category, we need to determine: (i) how many replies to letter requests are made uninterrupted (i.e., contiguously) for categories LTFB and Mixed, and (ii) for each number of letter replies, the time duration over which these letter replies are made (for categories FB, LTFB, and Mixed). These are the four values for a player v_i's aptitude b_i^{rpl} in replying to letter requests.

Algorithm for reply to (letter) request. Owing to space limitations, the algorithm is not provided here, but will be in an extended version of this work.

Remark: In these various algorithms, elements of sets are returned, or a distribution corresponding to particular inputs is sampled. In some cases, there are no data for specified conditions. For these types of situations, we implement a recursive search technique to sample from the distribution or set with the closest set of inputs.

V. AGENT-BASED SIMULATIONS AND RESULTS

Remark: *Model evaluation* is an important step and has been performed. Figures 3, 4, and 7 are part of this process.

Simulation model. We conduct discrete time agent-based simulations (ABSs) of the GrAG. Each time unit is one second of the 300-second GrAG. At each time and for each agent, an action is selected. Based on the action chosen, the corresponding model for that action, developed herein, is executed (Figures 6 and 8 for "form word" and "request letter," respectively, and the thinking action is a no-op). Thus, all agents behavior in the simulations follow these models. We run $n_{runs} = 100$ runs or simulation instances and average the results. We use the 5000-word corpus C^W. These are purely simulation studies and are not tied to the experiments. The goal is to demonstrate that the models alone provide insights into human behavior.

Study 1: Effects of model aptitude properties. We use a game configuration $G(V, E)$ consisting of six players that form a circle, with each player having two neighbors. The initial letter assignments are given in Table I. We systematically vary the aptitudes of players in forming words b_i^{wf}, in requesting letters b_i^{req}, and in replying to letter requests b_i^{rpl}. See Table II. Recall that these aptitudes correspond to the skill levels of players.

TABLE I
STUDY 1 INITIAL LETTER ASSIGNMENTS TO PLAYERS IN SIMULATIONS FOR SIX PLAYERS ARRANGED AS 2-REGULAR GRAPH.

Player #:	1	2	3	4	5	6
Init. Ltrs:	b, a, t	m, e, n	l, u, t	s, o, p	h, u, g	r, i, e

TABLE II
PARAMETERS THAT ARE SYSTEMATICALLY VARIED IN THE SIMULATIONS OF STUDY 1. THESE APTITUDE (b_i^{wf}, b_i^{req}, b_i^{rpl}) SETTINGS ARE THE SAME FOR ALL AGENTS IN A SIMULATION.

Sim. No.	b_i^{wf}	b_i^{req}	b_i^{rpl}	**Sim. No.**	b_i^{wf}	b_i^{req}	b_i^{rpl}
1	P_1	Q_1	FB	5	P_5	Q_5	FB
2	P_2	Q_2	FB	6	P_5	Q_5	LTFB
3	P_3	Q_3	FB	7	P_5	Q_5	NR
4	P_4	Q_4	FB	–	–	–	–

Figure 9 (left) shows the average number of interactions (requests sent, replies received, requests received, replies sent) and the average number of words formed per player for the first five simulation numbers (sim. no.) of Table II. There is a drop-off in performance in going from $b_i^{wf} = P_1$ to P_5, $b_i^{req} = Q_1$ to Q_5, for fixed b_i^{rpl} =FB. We observe that decreasing the letter request aptitude b_i^{req} and the word formation aptitude b_i^{wf} decreases the quality of letters requested and hence the number of words that can be formed.

To determine how b_i^{rpl} affects performance, we plot in Figure 9 (right) results from simulation numbers 5, 6, and 7 of Table II. Using $b_i^{wf} = P_5$ and $b_i^{req} = Q_5$ as a reference, there is a large decrease in numbers of reply interactions in going from b_i^{rpl} =LTFB to b_i^{rpl} =NR, as expected, since NR means that agents do not reply to letter requests.

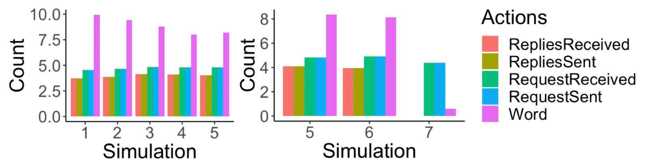

Fig. 9. (Left) Simulation results for Sim. nos. 1 through 5 of Table II. The average number of words formed per player drops in going from $b_i^{wf} = P_1$ to P_5, $b_i^{req} = Q_1$ to Q_5, for fixed b_i^{rpl} =FB. (Right) Simulation results for Sim. nos. 5, 6, and 7 of Table II. Using $b_i^{wf} = P_5$ and $b_i^{req} = Q_5$ as a baseline, these results show a precipitous drop-off in replies to letter requests, and to words formed, in going from b_i^{rpl} =LTFB to b_i^{rpl} =NR. Results in counts for b_i^{rpl} =LTFB are slightly less than those for b_i^{rpl} =FB.

Study 2: Effects of heterogeneity: network connectivity and quality of letter assignments to players. We use a game configuration $G(V, E)$ consisting of four players v_i $(1 \leq i \leq 4)$ that form a star. The initial letter assignments are given in Figure 10. All players have the following conditions $b_i^{wf} = P_1$, $b_i^{req} = Q_1$, and b_i^{rpl} =FB. Players are assigned heterogeneous numbers and qualities of letters; see the figure caption. The numbers of requests received and replies sent are greatest for player v_1 owing to its centrality; this affects the number of words player v_1 forms, which is less than those for v_2 and v_3. Players v_2 and v_3 have more requests received from v_1 (compared to v_4) because their letters (i.e., popular consonants) create larger sets of possible words to form. The number of words formed is least for player v_4 because of the poorer quality of assigned letters.

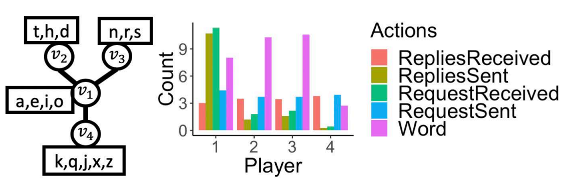

Fig. 10. Simulation for players v_i $(1 \leq i \leq 4)$, arranged in a star. All players have the following conditions $b_i^{wf} = P_1$, $b_i^{req} = Q_1$, and b_i^{rpl} =FB. Player v_1 is at the center with three neighbors. v_1 is assigned the four most popular vowels in the alphabet; v_2, v_3 are assigned the six most popular consonants, and v_4 is assigned the five least popular consonants. See text for discussion of results.

VI. Summary and Future Work

We have developed mechanistic and data-driven models for representing the decision-making and actions of humans in online networked GrAGs. Our contributions are in Section I-D. We would like to conduct more experiments with more network structures. This would also (ideally) produce sufficient data to more finely partition aptitudes—player behavior—into ten 10% bins (currently, we have five 20% bins). These experiments would be used to further evaluate the models and improve them.

Acknowledgment

We thank the anonymous reviewers for their helpful feedback. We thank our colleagues at NSSAC and computer system administrators: Dominik Borkowski, Jason Decker, Miles Gentry, Jeremy Johnson, William Marmagas, Douglas McMaster, and Kevin Shinpaugh. This work has been partially supported by NSF CRISP 2.0 (Grant 1832587), DARPA Cooperative Agreement D17AC00003 (NGS2), DTRA CNIMS (Contract HDTRA1-11-D-0016-0001), DTRA Comprehensive National Incident Management System Contract HDTRA1-17-D-0023. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon.

References

[1] W. L. Davis and D. E. Davis, "Internal-external control and attribution of responsibility for success and failure," *J. of Personality*, 1972.
[2] G. Charness, P. Kuhn, and M. C. Villeval, "Competition and the ratchet effect," *Journal of Labor Economics*, vol. 29, pp. 513–547, 2011.
[3] G. Charness, R. Cobo-Reyes *et al.*, "Identities, selection, and contributions in a public-goods game," *Games and Economic Behavior*, 2014.
[4] F. Polletta and J. M. Jasper, "Collective identity and social movements," *Annual Review of Sociology*, vol. 27, pp. 283–305, 2001.
[5] M. Goldman, J. W. Stockbauer *et al.*, "Intergroup and intragroup competition and cooperation," *J. of Exper. Soc. Psych,*, 1977.
[6] C. B. Cadsby *et al.*, "Sorting and incentive effects of pay for performance: An experimental investigation," *Acad. of Mgmt. Jour.*, 2007.
[7] Y. Ren *et al.*, "Generative modeling of human behavior and social interactions using abductive analysis," in *ASONAM*, 2018, pp. 413–420.
[8] V. Cedeno-Mieles *et al.*, "Modeling anagram games," Tech. Rep., 2019, https://github.com/vcedeno/asonam19/blob/master/tr.pdf.
[9] V. I. Levenshtein, "Binary codes capable of correcting deletions, insertions and reversals," *Soviet Physics Doklady*, 1966.
[10] J. M. Hofman, A. Sharma, and D. J. Watts, "Prediction and explanation in social systems," *Science*, vol. 355, pp. 486–488, 2017.
[11] D. J. Watts, "Should social science be more solution-oriented?" *Nat. Hum. Behav.*, pp. 1–5, 2017.
[12] A. Bokulich, "How scientific models can explain," *Synthese*, vol. 180, pp. 33–45, 2011.
[13] J. Jebeile and A. G. Kennedy, "Explaining with models: The role of idealization," *Int. Studies in the Philos. of Science*, pp. 383–392, 2015.
[14] J. Sweller, "Cognitive load during problem solving: Effects on learning," *Cognitive Science*, vol. 12, pp. 257–285, 1988.
[15] G. S. Becker, *The economic approach to human behavior*. University of Chicago Press Chicago, 1976.
[16] W. Mason and D. J. Watts, "Collaborative learning in networks," *PNAS*, vol. 109, pp. 764–769, 2012.
[17] N. P. Nguyen, G. Yan *et al.*, "Containment of misinformation spread in online social networks," in *Web Science Conference*, 2012, pp. 213–222.
[18] D. O'Callaghan *et al.*, "Uncovering the wider structure of extreme right communities spanning popular online networks," in *Web Sci*, 2013.
[19] Z. Zhao, M. Madaio *et al.*, "Socially-conditioned task reasoning for a virtual tutoring agent," in *AAMAS*, 2018, pp. 2265–2267.
[20] J. O'Doherty and P. Bossaerts, "Toward a mechanistic understanding of human decision making; contributions of functional neuroimaging," *Current Directions in Psychological Science*, vol. 17, pp. 119–123, 2008.
[21] S. Spaulding *et al.*, "A social robot system for modeling children's word pronunciation: Socially interactive agents track," in *AAMAS*, 2018.
[22] "Word frequency data: Corpus of contemporary american english," https://www.wordfrequency.info/free.asp, accessed: 2018-11-16.