

Supplemental Information for
Interdependent Diffusion:
The social contagion of interacting beliefs

James Houghton

May 28, 2021

Contents

S1 Simulation	S2
S1.1 Simulation code	S2
S1.2 Motivating the “Knowledge Graph” representation of belief interaction	S6
S1.3 Sensitivity to parameters	S8
S2 Analysis of Observational Data	S9
S2.1 Description of Datasets	S9
S2.1.1 KPTimes	S9
S2.1.2 Web of Science	S9
S2.1.3 Pew Political Typology Survey	S9
S2.1.4 World Values Survey	S10
S2.2 Alternate clustering analysis	S10
S3 Experiment	S11
S3.1 Design considerations and experiment parameters	S11
S3.2 Design of participant interface	S13
S3.2.1 Consent	S13
S3.2.2 Training	S13
S3.2.3 Game introduction	S14
S3.2.4 Playing the game: Exchanging clues	S14
S3.2.5 Post-game survey: Making the case	S17
S3.3 Experimental manipulation/stimulus	S18
S3.3.1 Constructing the bank of clue concepts	S18
S3.3.2 Assembling sets of clues for use in games	S24
S3.4 Data collection and measurements	S25
S3.4.1 Recording player actions	S25
S3.4.2 Choice of summary statistics	S26
S3.4.3 Handling missing data	S26
S3.5 Preregistration	S26
S3.6 Outcomes	S27
S3.6.1 Participants’ experience	S27
S3.6.2 Detailed Results	S28
S4 Resources for replication, extension and reanalysis	S29
S4.1 Conditions of validity	S29
S4.2 Replicating these results	S29
S4.3 Extending this research	S29
S4.4 Opportunities for reuse of the data generated by this experiment	S29

S1 Simulation

S1.1 Simulation code

The simulations presented in this paper can be replicated with the following code. This code is optimized for clarity over speed. This code was originally written using:

- Python 3.7.1
- NetworkX 2.3
- Pandas 0.24.2
- Scikit-learn 0.20.1

In this code, `g` is the social network (a networkx graph object) with `n` agents. Each node in `g` is an agent numbered $0 \dots n - 1$, and has an attribute `M` which stores the agent's current belief set as a knowledge graph (another networkx graph object for each agent). In the independent condition, agents also have an attribute `S` representing their susceptibility to beliefs, also formatted as a knowledge graph. When beliefs are passed between functions, it is either as an ordered tuple representing the edge in a knowledge graph `(2, 7)` or as a numpy array of tuples `np.array([(2, 7), (2, 16), ...])`

To import this module into another python file or jupyter notebook, call:

```
from example_code import *
```

To run a matched simulation of interdependent and independent diffusion, call:

```
result = run()
```

To run a number of simulations and average their output, call:

```
n_sims = 10
df = pd.concat([run() for i in range(n_sims)])
result = df.groupby(level=0).aggregate('mean')
```

This code and supporting materials are available at: <https://github.com/JamesPHoughton/interdependent-diffusion>.

```
1  """
2  example_code.py
3  James Houghton
4  houghton@mit.edu
5  """
6  import networkx as nx
7  import numpy as np
8  import itertools
9  import pandas as pd
10 import copy
11 from sklearn.decomposition import PCA
12
13
14 def susceptible(g, agent, belief):
15     """Assess whether an agent is susceptible to a given belief"""
16     if 'S' in g.node[agent]: # has exogenous susceptibility defined (independent case)
17         return g.node[agent]['S'].has_edge(*belief)
18     else: # interdependent case
19         try:
20             return nx.shortest_path_length(g.node[agent]['M'], *belief) <= 2
21             # current holders are also susceptible
22         except (nx.NetworkXNoPath, nx.NodeNotFound):
23             return False # no path exists between the nodes
24
```

```

25
26 def adopt(g, agent, belief):
27     """Assess whether an agent will adopt a given belief"""
28     suscep = susceptible(g, agent, belief)
29     exposed = any([belief in g.node[nbr]['M'].edges() for nbr in g[agent]])
30     return suscep and exposed # both susceptibility and exposure required to adopt
31
32
33 def measure(g, beliefs, initial_susceptible=None, initial_adopted=None):
34     """Take measurements of the state of the system (for creating figures)"""
35     res = {} # dictionary to collect measurements
36
37     # Fig 2A: Susceptible and adopting populations
38     # -----
39     # build a matrix of who (rows) is susceptible to what beliefs (columns)
40     suscep = pd.DataFrame(index=g.nodes(), columns=[tuple(b) for b in beliefs])
41     for agent in g:
42         for belief in suscep.columns:
43             suscep.at[agent, belief] = susceptible(g, agent, belief)
44     # return average susceptible fraction across all beliefs
45     res['% susceptible'] = suscep.mean().mean()
46
47     # build a matrix of who (rows) holds what beliefs (columns)
48     adopt = pd.DataFrame(index=g.nodes(), columns=[tuple(b) for b in beliefs])
49     for agent in g:
50         for belief in adopt.columns:
51             adopt.at[agent, belief] = g.node[agent]['M'].has_edge(*belief)
52     # return average adopting fraction across all beliefs
53     res['% adopted'] = adopt.mean().mean()
54
55     # Fig 2B: correlation between predicted new adoption and actual new adoption
56     # -----
57     if initial_adopted is not None and initial_susceptible is not None: # t>0
58         res['initial prediction correlation'] = np.corrcoef(
59             adopt.sum(axis=0) - initial_adopted,
60             initial_susceptible - initial_adopted
61         )[1, 0] # select an off-diagonal term
62     else: # first time => establish baseline
63         initial_adopted = adopt.sum(axis=0)
64         initial_susceptible = suscep.sum(axis=0)
65         res['initial prediction correlation'] = np.nan # measure has no meaning at t0
66
67     # Fig 2C: correlation between a belief and it's most popular neighbor
68     # -----
69     adopt_counts = pd.DataFrame()
70     adopt_counts['self'] = adopt.sum(axis=0)
71     adopt_counts['leading neighbor'] = 0
72     for c1 in adopt.columns:
73         # search for the leading neighbor's popularity
74         leading_value = 0
75         for c2 in adopt.columns:
76             if len((set(c1) | set(c2))) == 3: # three nodes total => c1 and c2 are neighbors
77                 leading_value = max(leading_value, adopt_counts.loc[[c2], 'self'].values[0])
78             adopt_counts.at[[c1], 'leading neighbor'] = leading_value
79     res['leading neighbor correlation'] = adopt_counts.corr().loc['self', 'leading neighbor']
80
81     # Fig 2D: clustering coefficient of 10% most popular beliefs

```

```

82 # -----
83 # shuffle within sorted value so that when 10% falls within a level of popularity
84 # we don't add spurious clustering by selecting sequential beliefs
85 adopt_counts['shuffle'] = np.random.rand(len(adopt_counts))
86 adopt_counts.sort_values(by=['self', 'shuffle'], inplace=True, ascending=False)
87 leaders = adopt_counts.iloc[:int(len(adopt_counts) * 0.1)] # take leading 10% of beliefs
88 # construct knowledge graph from leading beliefs
89 popular_graph = nx.from_edgelist(list(leaders.index))
90 res['popular belief clustering'] = nx.average_clustering(popular_graph)
91
92 # Fig 3A: similarity btw 5% and 95% most similar pairs
93 # -----
94 n_agents = len(adopt.index)
95 trimask = np.tri(n_agents, n_agents, 0, dtype='bool') # mask the diagonal and below
96 corrs = adopt.astype(float).T.corr().mask(trimask).stack()
97 res['95% similarity'] = np.percentile(corrs, 95)
98 res['5% similarity'] = np.percentile(corrs, 5)
99
100 # Fig 3B: PC1 percent variance
101 # -----
102 pca = PCA(n_components=1)
103 pca.fit(adopt)
104 res['PC1 percent of variance'] = pca.explained_variance_ratio_[0] * 100
105
106 return res, initial_susceptible, initial_adopted
107
108
109 def simulate(g, n_steps=10):
110     """Conduct a single run of the simulation with a given network"""
111     # capture a list of all the beliefs in the population
112     beliefs = np.unique([tuple(sorted(belief)) for agent in g
113                          for belief in g.node[agent]['M'].edges()], axis=0)
114
115     # measure initial conditions
116     m0, initial_susceptible, initial_adopted = measure(g, beliefs)
117     # initialize list to collect measurements at each time step
118     m = [m0]
119
120     # perform the simulation
121     for step in range(n_steps):
122         # cycle through agents in random order
123         for ego in np.random.permutation(g):
124             # cycle through all possible beliefs in random order
125             for edge in np.random.permutation(beliefs):
126                 # check whether the selected agent adopts the selected belief
127                 if adopt(g, ego, edge):
128                     # add the belief to the agent's knowledge graph
129                     g.node[ego]['M'].add_edges_from([edge])
130             # ignore returned init suscep and adopt
131             m.append(measure(g, beliefs, initial_susceptible, initial_adopted)[0])
132
133     return pd.DataFrame(m) # format as pandas DataFrame
134
135
136 def run(n_agents=60, deg=3, n_concepts=25, n_beliefs=25, t_match_susceptibility=0, n_steps=10):
137     """
138     Run a matched pair of simulations (inter/independent) from the same initial condition

```

```

139
140 Parameters
141 -----
142 n_agents: (integer) - Number of agents in the population
143 deg: (integer) - How many neighbors each agent has (on average)
144 n_concepts: (integer) - How many nodes are in the complete semantic
145 network that beliefs are drawn from
146 n_beliefs: (integer) - Exact number of beliefs (semantic net edges)
147 each agent is initialized with
148 t_match_susceptibility: (integer) - time step at which the interdependent
149 susceptibility will be matched
150 (must be less than n_steps)
151 n_steps: (integer) - Number of time steps in the simulation
152 """
153
154 # Shared Initial Setup
155 # -----
156 # create a random connected social network g0
157 connected = False
158 while not connected:
159     g0 = nx.gnm_random_graph(n=n_agents, m=int(n_agents * deg / 2))
160     connected = nx.is_connected(g0)
161
162 # give agents their initial beliefs
163 nx.set_node_attributes(
164     g0,
165     name='M', # set node attribute 'M' (for 'mind')
166     # create a knowledge graph with a different random set of beliefs
167     # for each agent, and assign to nodes in the social network
168     values={agent: nx.gnm_random_graph(n_concepts, n_beliefs) for agent in g0}
169 )
170
171 # Interdependent simulation
172 # -----
173 g1 = copy.deepcopy(g0) # create copy, to preserve initial conditions for other case
174 res1 = simulate(g1, n_steps)
175
176 # Independent simulation
177 # -----
178 g2 = copy.deepcopy(g0) # copy from original starting conditions
179
180 # calculate the population likelihood of being susceptible to a given (non-held) belief
181 p = ((res1.loc[t_match_susceptibility, '% susceptible'] - res1.loc[0, '% adopted']) /
182      (1 - res1.loc[0, '% adopted']))
183
184 # choose a set of beliefs for each agent to be susceptible to
185 new_sus = {}
186 for agent in g2:
187     # potentially susceptible to any belief
188     gc = nx.complete_graph(n_concepts)
189     # temporarily remove existing beliefs
190     gc.remove_edges_from(g2.node[agent]['M'].edges())
191     # from remainder, randomly select a subset of beliefs to be susceptible to
192     edges = list(itertools.compress(
193         list(gc.edges()), # selection candidates
194         np.random.binomial(n=1, p=p, size=len(gc.edges())) == 1 # selection mask
195     ))

```

```

196     # add susceptibility to existing beliefs
197     edges += list(g2.node[agent]['M'].edges())
198     # create networkx graph of susceptibilities
199     new_sus[agent] = nx.from_edgelist(edges)
200
201     # assign new susceptibilities to positions in social network
202     nx.set_node_attributes(g2, name='S', values=new_sus)
203
204     # perform independent simulation
205     res2 = simulate(g2, n_steps)
206
207     return pd.merge(res1, res2, left_index=True, right_index=True,
208                    suffixes=(' (inter)', ' (indep)')) # format as single DataFrame

```

S1.2 Motivating the “Knowledge Graph” representation of belief interaction

There are two primary ways that we could model the interaction between beliefs. One method used by Friedkin et al. [5] is to create a matrix of the compatibility between candidate beliefs (rows) and preexisting beliefs (columns). In this model, weights in the matrix determine the influence of (unstructured) existing beliefs on the probability of an individual adopting each candidate belief. This formulation assumes external ‘logic’ under which certain beliefs naturally go together.

A second method is the knowledge graph representation suggested by Goldberg and Stein [7] and by Schilling [11], in which existing beliefs have no independent effect on an agent’s susceptibility to a candidate belief. Instead, they influence future adoption decisions through the structure they create in the individual’s knowledge graph. In this representation, all beliefs are potentially compatible with one another, depending on the arrangement of other beliefs that the individual holds.

When we study the effect of diffusion on the emergence of worldviews and polarization, it’s important that we choose a representation of belief interaction that does not foreordain particular outcomes. If we can predict patterns of belief clustering and polarization from the decision logic alone, then the simulation will be unable to identify whether clustering is shaped by social contagion, or is merely a result of the assumed decision logic.

Unfortunately, an interdependence matrix which maps the presence of belief A to the likelihood of adopting belief B will always be informative of the final configuration of beliefs (or trivial). We can demonstrate this outcome with an extremely simple deterministic model with no diffusion at all. This model is in the style of Friedkin et al. [5], but omits social influence, stochasticity, and degrees of belief. These simplifications let us see intuitively why the interdependence matrix is problematic for our purpose. When stochasticity, social influence, etc. are present, the problem doesn’t go away, it is just masked by the complexity of the model. In this simplified interdependence-matrix belief adoption model:

1. Each agent is exposed to all beliefs in every timestep (making this an individual learning model, not a social learning model).
2. The presence of belief ‘A’ in an agent’s existing belief set either contributes (+) to an agent’s likelihood of adopting candidate belief ‘B’, or takes away from it (−).
3. The absence of belief ‘A’ from an agent’s belief set has no direct influence on an agent’s likelihood of adopting candidate belief ‘B’.
4. Belief adoption is binary and permanent.
5. Agents adopt a candidate belief if the majority of their current beliefs support doing so.¹

To show that the matrix of influence is informative of the outcomes of contagion, we simulate a population of 64 individuals, each with a unique combination of six beliefs, denoted A-F. The top center chart in Fig. S1 illustrates this starting condition. Each column represents an individual agent (0-63), each row represents a belief (A-F). I darken the corresponding square to show that an agent has adopted a particular belief. To the right-hand side of the adoption plot, a histogram shows the total number of individuals adopting each belief. In the initial condition, all beliefs have been adopted by 32 individuals.

¹Other thresholds would lead us to the same insights that we will develop with the “majority rule”.

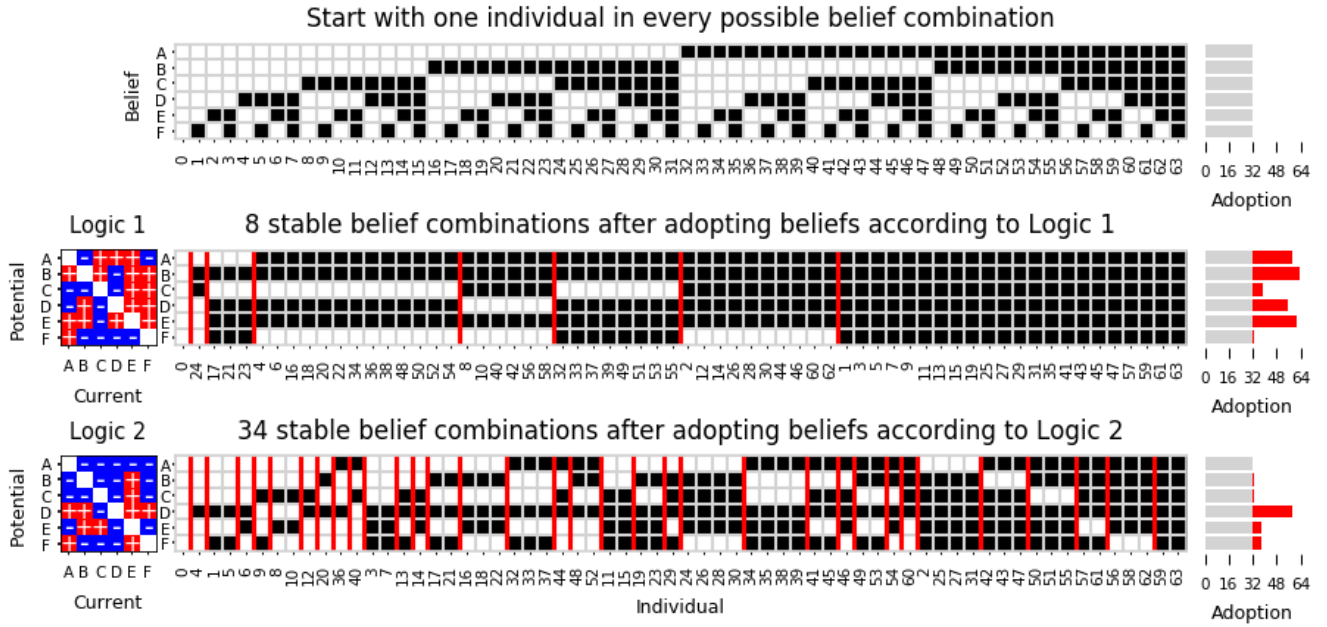


Figure S1: With an interdependence-matrix style belief structure, polarization and belief clustering can be explained by the assumed compatibility relationships. But where does this logic come from?

The subsequent two plots in Fig. S1 show the final adoption of beliefs under two influence matrices (Logic 1 and Logic 2) in the style of Friedkin et al.[5]. Logic 1 and Logic 2 are randomly generated influences between beliefs, shown as + (red) and - (blue). In each matrix, if an individual holds A, then the influence of that belief on the adoption of belief B is found in Column A, Row B. In Logic 1, the influence of A on B is positive, and in Logic 2, this influence is negative.

For each individual, we calculate the net influence of existing beliefs on adoption of each candidate belief. If the candidate belief gets a majority of + votes then it is adopted. We repeat the process until all agents have converged on a stable combination of beliefs.

Any model of social learning with binary and permanent belief adoption and a finite number of beliefs will show some form of belief consolidation, and an increase in mean similarity between individuals. We expect to see that from our maximally-differentiated initial conditions, some groups of stable belief combinations should emerge. In the adoption plots in Fig. S1, I group individuals according to their stable sets of beliefs, indicating the groups with a red divider.

Despite being drawn from the same pool of possible logics, the two Friedkin-style logics yield very different stable combinations of beliefs. In Logic 1, there are 8 different stable combinations, and in Logic 2, there are 34. In a diffusion model, we would expect to see a lot more consolidation of beliefs, and clustering of individuals, using Logic 1 rather than Logic 2, regardless of social contagion. Moreover, the two different logics strongly influence which beliefs we should expect to be widely adopted in the population. Logic 1 suggests strong new adoption (shown in red) of beliefs A, B, D, and E. Logic 2 shows that only belief D is widely adopted, with belief A having no new diffusion at all.

In the real world, it may well be that a natural logic ties some beliefs together, and promotes the diffusion of some beliefs at the expense of others. In our simulation, however, this influence makes it difficult to identify clustering and amplification of beliefs that is due to social contagion. In this paper, I have suggested an alternative formulation, in which beliefs are formalized as edges in a knowledge graph, and the adoption rule is that beliefs are adopted if they close a triangle in an individual's knowledge graph.

In Fig S2, I repeat the above analysis with this new formulation. Again there are six beliefs, representing all of the possible edges in a knowledge graph with concepts P, Q, R and S. Again, each of 64 individuals is initialized with a unique combination of beliefs, and has access to all six beliefs. Each individual adopts new beliefs that are consistent with the triangle-closing decision rule, and I plot the stable belief combinations after each individual has individually converged. As there is only one decision rule, there can be only one outcome for grouping the sets of stable beliefs.

With this formalization, we know exactly how the decision rule for adoption influences the number of possible

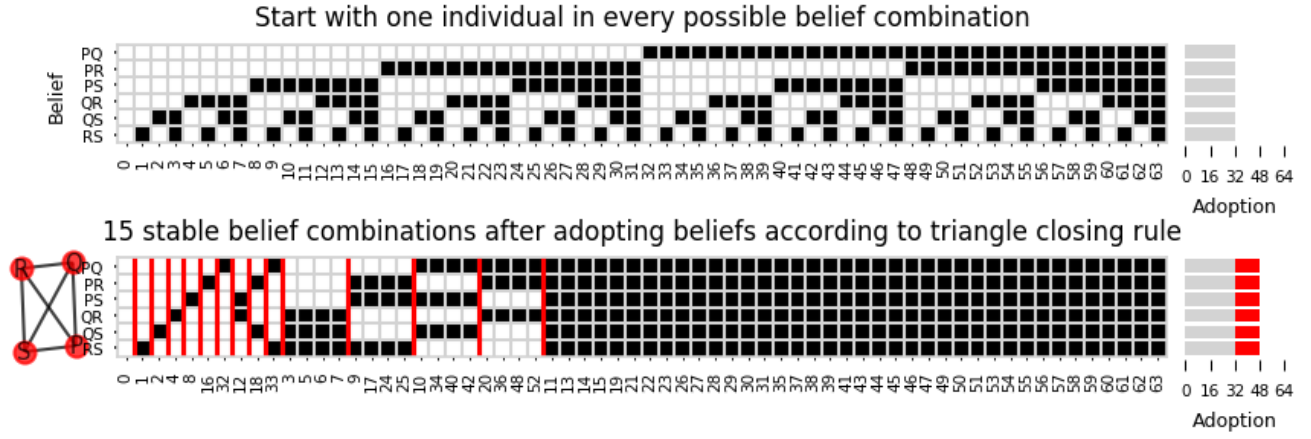


Figure S2: With the knowledge graph representation of belief structure, we have confidence that any variance in adoption patterns, or differences in social clustering, are due to the interaction of the belief structure with the diffusion process.

stable states; and most importantly, these states are symmetric with respect to the individual beliefs. The histogram to the right of the second row of S2 shows that each belief has an equal number of new adoptions, and that we should not expect any beliefs to preferentially diffuse as a result of the decision rule itself.

In this paper I have shown that some beliefs do indeed diffuse much more widely than others, and that the population clusters into a subset of the possible stable states. Because the decision rule does not influence which of the beliefs diffuse most widely, or suggest variation in the number of social clusters we can expect, we have confidence that the results are genuinely due to the interactions between beliefs as they diffuse.

S1.3 Sensitivity to parameters

Figure 2D in the main body of the paper makes a comparison between the clustering coefficient amongst the most popular beliefs, defined as the 10% most widely adopted. Fig. S3 in this supplement shows that the qualitative result is insensitive to this particular choice of threshold between approximately 5% and 40%.

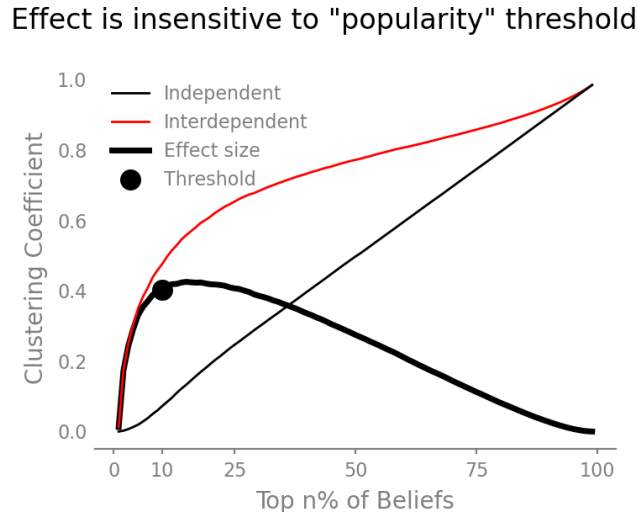


Figure S3: The effect of interdependence on the clustering of the most popular beliefs is robust to a wide range of thresholds for “popularity”. In the simulations presented in this experiment, I use a conservative 10% threshold.

In Fig. 3A, 5B and 5C of the paper, I use 5th and 95th percentile values to characterize the level of similarity across and within ideological camps. These thresholds are arbitrary, in that we could have plausibly used a wide range of other values. Fig. S4 shows that the qualitative effect we are interested in is insensitive to the precise choice

of threshold.

Effect is insensitive to within/across camp thresholds

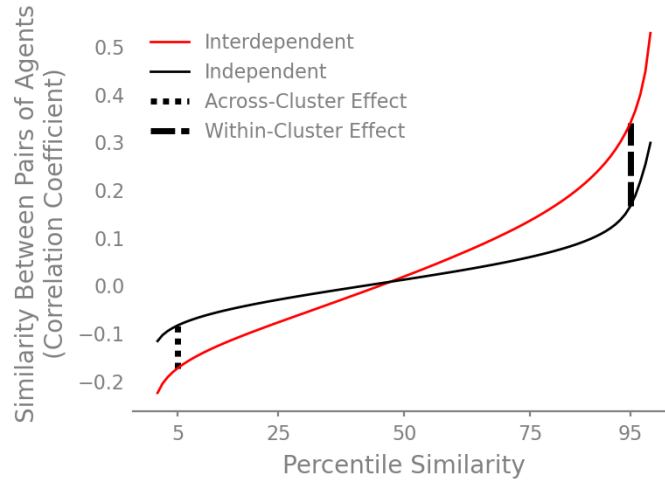


Figure S4: The effect of interdependence on within-camp and across-camp similarity is robust to a wide range of thresholds for assessing whether relationships are within or across camp. In the simulation and experiment, I use conservative 5th and 95th percentile thresholds.

S2 Analysis of Observational Data

S2.1 Description of Datasets

All code needed to generate Fig. 4 in the main body is available in the [/observational](#) folder of the code repository.

S2.1.1 KPTimes

The KPTimes dataset [6] contains New York Times articles between January 1, 2006 and June 25, 2019. The analysis in Fig. 4 of the main body included 288,545 documents from 27 categories. These categories were selected on the criteria that they include ≥ 3000 documents. When a document was labeled with multiple categories, it was assigned to the smallest category with ≥ 3000 documents. The categories included in the analysis are: (parentheses indicate the number of documents per category)

africa (5203), americas (4801), asia (16218), baseball (12754), basketball (7023), business (33686), dealbook (3815), economy (3186), europe (14630), football (8209), global (5762), golf (3471), hockey (4323), media (9427), middleeast (12338), national (3513), ncaabasketball (4011), ncaafootball (3180), nyregion (43221), politics (15094), science (7419), soccer (5254), sports (15059), technology (9634), tennis (3520), us (29420), world (4374)

S2.1.2 Web of Science

The Web of Science dataset [10] was originally collected by Kowsari et al. [9] for training text classification algorithms, and was repurposed for this analysis. The dataset contains 46,985 documents in 7 Major fields: (parentheses indicate the number of documents per category)

Computer Science (6514), Civil Engineering (4237), Electrical Engineering (5483), Mechanical Engineering (3297), Medical (14625), Psychology (7142), Biochemistry (5687)

S2.1.3 Pew Political Typology Survey

The Pew Political Typology Survey interviewed 10,013 US adults in three approximately equal waves from January 23, 2014 through March 16, 2014. Each wave was split into half land-line and half cellphone interviews. The analysis in Fig. 4 of the main body included 58 questions about political opinions, chosen prior to data observation to focus

only on political opinions (i.e. to exclude questions about demographics and other ascribed characteristics) and to consolidate redundant questions. Not all questions were included in each wave. The question codes included are: q25a, q25b, q25c, q25d, q25f, q25g, q25h, q25i, q25j, q25k, q25l, q25m, q25n, q25o, q25p, q50q, q50r, q50s, q50t, q50u, q50v, q50w, q50y, q50z, q50aa, q50bb, q50dd, q50ee, q50ff, q50gg, q50hh, q51ii, q51jj, q51kk, q51ll, q51mm, q51oo, q51pp, q53, qb26, qc26, qb106, qb107, qb109, qb110, qc115, qc116, q121, q122, q123, q124, q125, q126, qc127, qc128, qc135, qb56, qb108

S2.1.4 World Values Survey

The 2020 World Values Survey [8] contains 70,867 responses collected from 2017-2020. The analysis in Fig. 4 of the main body used 144 questions relating to apolitical social values. The questions to be used were chosen prior to data observation to focus only on social opinions (i.e. to exclude questions about demographics and other ascribed characteristics, and to exclude explicitly political questions) and to consolidate redundant questions. The question codes included are:

Q1, Q2, Q3, Q4, Q5, Q6, Q7, Q8, Q9, Q10, Q11, Q12, Q13, Q14, Q15, Q16, Q17, Q18, Q19, Q20, Q21, Q22, Q23, Q24, Q25, Q26, Q27, Q28, Q29, Q30, Q31, Q32, Q33, Q34, Q35, Q36, Q37, Q38, Q39, Q40, Q41, Q43, Q44, Q58, Q59, Q60, Q61, Q62, Q63, Q64, Q65, Q66, Q67, Q68, Q69, Q70, Q71, Q72, Q73, Q74, Q75, Q76, Q77, Q78, Q79, Q80, Q81, Q82, Q83, Q84, Q85, Q86, Q87, Q88, Q89, Q106, Q107, Q108, Q109, Q110, Q121, Q122, Q123, Q124, Q125, Q126, Q127, Q128, Q129, Q130, Q158, Q159, Q160, Q161, Q162, Q163, Q164, Q165, Q166, Q167, Q168, Q169, Q170, Q171, Q172, Q173, Q174, Q176, Q177, Q178, Q179, Q180, Q181, Q182, Q183, Q184, Q185, Q186, Q187, Q188, Q189, Q190, Q191, Q192, Q193, Q194, Q195, Q196, Q197, Q198, Q235, Q236, Q237, Q238, Q239, Q241, Q242, Q243, Q244, Q245, Q246, Q247, Q248, Q249

The analysis in Fig. 4 of the main body includes a curve for each of 49 different country codes: (parentheses show number of respondents)

AND (1004), ARG (1003), AUS (1813), BGD (1200), BOL (2067), BRA (1762), CHL (1000), CHN (3036), COL (1520), CYP (1000), DEU (1528), ECU (1200), EGY (1200), ETH (1230), GRC (1200), GTM (1203), HKG (2075), IDN (3200), IRN (1499), IRQ (1200), JOR (1203), JPN (1353), KAZ (1276), KGZ (1200), KOR (1245), LBN (1200), MAC (1023), MEX (1739), MMR (1200), MYS (1313), NGA (1237), NIC (1200), NZL (1057), PAK (1995), PER (1400), PHL (1200), PRI (1127), ROU (1257), RUS (1810), SRB (1046), THA (1500), TJK (1200), TUN (1208), TUR (2415), TWN (1223), UKR (1289), USA (2596), VNM (1200), ZWE (1215)

S2.2 Alternate clustering analysis

The analysis presented in Fig. 4A of the main body included connections between all keywords attached to each newspaper article or academic paper. An alternate method of producing these graphs is to only include connections between *pairs* of keywords within each article, randomly selected without replacement. While this results in a significantly sparser dataset, it guarantees that none of the clustering observed is due to clustering within documents.

With this method, the overall clustering is reduced across all thresholds, and in particular thresholds that include keywords found in only a few documents. However, the qualitative prediction of increased clustering among the most popular connections between keywords remains, as shown in Fig. S5.

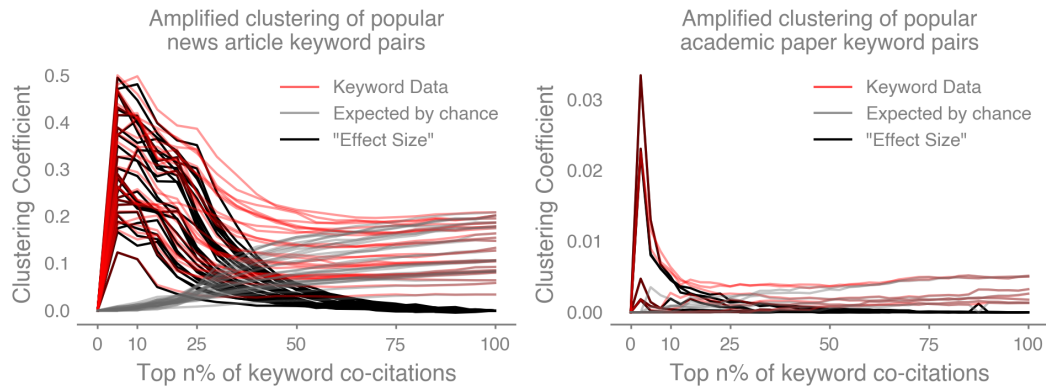


Figure S5: The predicted amplification of clustering among popular keyword pairs is preserved when the data is restricted to disallow any contribution to clustering from the attachment of more than two keywords to the same article.

S3 Experiment

The experimental design had two major objectives. 1) Allow for the manipulation of interdependence between diffusants, and 2) minimize the spillover of real-world polarization into the artificial social networks of the experiment. Beyond those considerations, I wanted to use a set of diffusants that individuals could quickly understand and would be interested in. A detective case solving a burglary was sufficiently relatable and abstract from the real world to serve as a stimulus and manipulation. Section S3.1 details the various experiment design choices that were not specific to the manipulation. Section S3.2 details the participant’s experience, and Section S3.3 details the design of the mystery’s clues, which formed the experimental manipulation. Section S3.4 describes how data was collected and handled. Section S3.5 describes the experiment’s preregistration and deviations between the preregistration and the realized experiment. Section S3.6 lists outcomes of the experiment that have not already been discussed.

S3.1 Design considerations and experiment parameters

There are many ways that the game interface and experimental parameters could have been implemented. Each decision took into account the feasibility of implementation, the constraints of the online lab context, the desire to create a naturalistic information and task environment, and the desire to cleanly isolate the mechanisms being tested. Below I justify the major decision points.

Information was presented to each participant all at once in the form of Detectives’ Notebooks shared by their neighbors. This was an intuitive structure for the detective-game task; it allowed players to quickly understand how their categorization decisions would be shared and how the information was presented to them. There is a slight cost in real-world representativeness with this design choice, as it is different from how information is typically presented in social media. However, the alternative “scrolling feed” type information display has recency and primacy effects, and opens questions about how we should aggregate social information from multiple players. Showing all information at once, in the same order that it is sorted by the neighbor, eliminates the effect of alternate ordering sequences.

Individuals had three social network neighbors. Given the chosen display, the number of neighbors is limited by the size of the screen and an individual’s ability to process information. The minimum number of neighbors for a non-trivial social network is 3, and is also a reasonable number for managing the cognitive load in the game.

Individuals began with four clues in their detectives’ notebooks. Fewer starting clues are preferred for minimizing individuals’ cognitive load. With three neighbors, individuals began the game having to process 16 clues. The next increment (5 starting clues each) would have given 20 items for an individual to process at game start, which (in “friends and family” beta tests) proved to be cognitively overwhelming.

The social network contained 20 players. Larger numbers of players are better for generalizability and seeing an effect size. On the other hand, smaller networks allow more replications and are easier to recruit and coordinate. There needed to be enough players that the mean shortest-path-length was greater than two, to realistically represent multi-stage diffusion. Pilot tests showed that we could reasonably expect to fill four 20-player games at once, and simulation suggested that 20-player networks should be sufficient to detect an effect size.

The social networks were shaped as a dodecahedron and a regular connected caveman ($k=5$) network. I evaluated eleven symmetric candidate social networks ($n=20$, $\text{degree}=3$) shown in Fig. S6. Of this set, the dodecahedral network minimizes the average shortest path between individuals with no network clustering, and represents a social network in which *a priori* we should expect to see low polarization. A regular connected caveman network maximizes the characteristic path length and exhibits strong clustering, and so we expect to see more polarization in this network. Edgelists for each of the eleven social networks are available in the [/experiment/setup](#) folder of the code repository.

Each game contained 78 unique clues. From an information diversity perspective, more clues is better. With 4 starting clues and 20 players, we can have up to 80 unique clues in the game. A complete 13 node clue network (knowledge graph) has 78 clues, and the two spots remaining were filled with the (given) link between the crime scene and the stolen object.

Each clue was represented in exactly 1 starting notebook. In order to make sure that the initial frequency of information in the network did not bias the network to certain outcomes, each clue was present exactly once in the clues initially assigned to players. Other than this constraint, clues were assigned randomly. The clue linking the stolen object to the crime scene was included 2 additional times to fill out the 80 slots (20 players * 4 starting clues) available.

The game was played for 8 minutes. Games needed to be long enough that participants had a chance to sort clues and make sense of the mystery, but short enough that they remained engaged and did not drop out of the

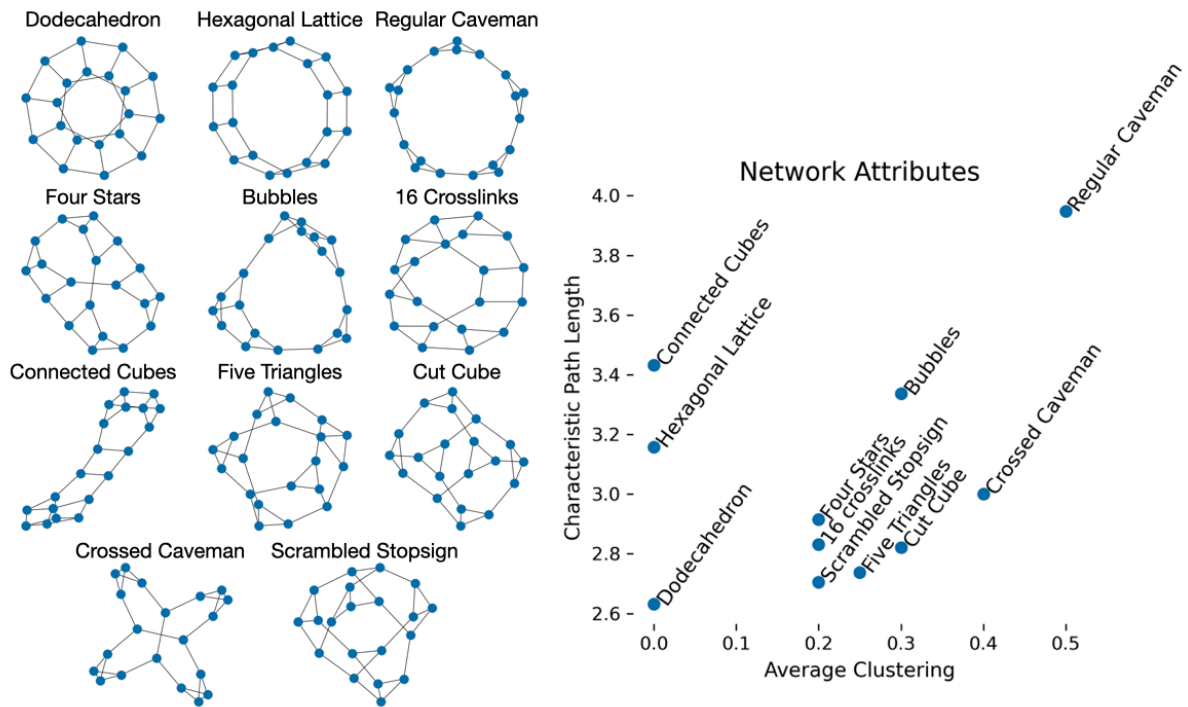


Figure S6: Various possible social networks were evaluated, and two were selected (dodecahedron and regular cave-man) to maximize the difference in average clustering and characteristic path length.

experiment. Varying the length of games during pilot trials suggested that 8 minutes was a good balance between these constraints. Fig. S7 shows that the average rate of categorization activity built over the first minute as players started to develop an understanding of the mystery, and then declined as individuals settled in on a solution.

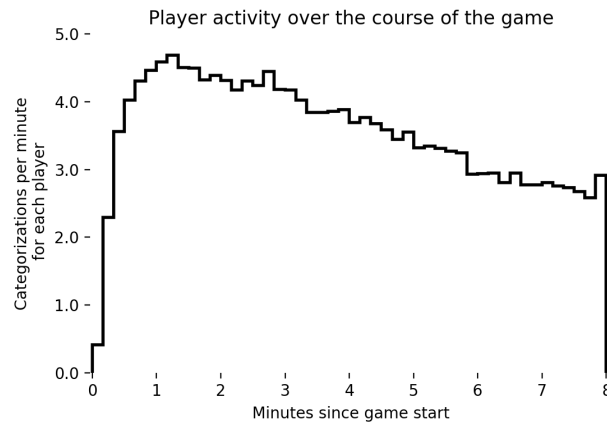


Figure S7: Average rate of activity for all players in experiment runs (excluding pilots).

Post-game opinions were elicited as likelihood of participation for each element in the mystery. Rather than force individuals to make a discrete choice between suspects/vehicles etc., individuals assessed how likely each element of the mystery was to have been involved in the crime. While this is different from how a jury might assess the guilt of a suspect in a courtroom, it gave more resolution on the strength of individuals' opinions.

Each condition was sampled 30 times. Each sample of the four conditions in this experiment cost about \$400. As there is no precedent for this type of multi-player, multi-diffusant experiment, and I did not have enough pilot data to know how effect sizes predicted in simulation would translate to experiment, it was difficult to assess the marginal benefit of additional samples relative to their cost. Based on the technical success of the pilots, I was able to secure \$12,000 to compensate participants, allowing for 30 sets of data-points.

S3.2 Design of participant interface

The experiment was implemented using the Empirica [1] framework, a platform for realtime multiparticipant web-based experiments.

S3.2.1 Consent

Upon arrival, participants were shown a consent screen (Fig. S8) telling them about the study and what they would be expected to do, including information that the games were oversubscribed and that they may not be able to complete the whole game. Those who gave their consent to participate continued to training.

About this study

This study takes 20-30 minutes.

You may play a collaborative game with other Mechanical Turk workers.

The game is over-subscribed to shorten waiting times.

You will be paid for training, even if you don't get to play.

What you need to do:

1. Please use a computer, not a mobile device.
2. When the game starts, please give it your full attention.
3. If you don't play actively, your team members may lose part of their bonus.

Consent to participate

This HIT is part of a MIT scientific research project. Your decision to complete this HIT is voluntary. There is no way for us to identify you. The only information we will have, in addition to your responses, is the time at which you completed the survey. Anonymous behavioral data from this study may be shared in public forums. The results of the research may be presented at scientific meetings or published in scientific journals. Clicking on the 'SUBMIT' button on the bottom of this page indicates that you are at least 18 years of age and agree to complete this HIT voluntarily.

Problems? Email detective@mit.edu.

✓ I AGREE

Figure S8: Participants indicate consent before proceeding.

S3.2.2 Training

The first training screen (Fig. S9) instructed participants in how to interact with the Detective Game interface. They were asked to sort clues into “Promising Leads” and “Dead Ends” by dragging them into labeled sections of their “Detective’s Notebook”. Having done so, they were shown clues that two artificial “collaborators” had categorized as promising leads. Each participant then had to correctly sort the collaborators’ clues before they could continue to the next training screen.

The second training screen (Fig. S10) told participants how they would be rewarded for their performance. Individuals were told that they would receive \$0.10 for each clue correctly categorized as a promising lead, and would be penalized \$0.10 for each clue categorized as a promising lead that was actually false. They were also told that they would be rewarded for their team’s average performance, receiving the average of all players’ individual bonuses as a Team Bonus. These incentives encourage individuals to carefully sort each clue according to their best estimate of its veracity, and to share clues with their neighbors that they believe will improve the team’s collective sense-making ability. Setting the reward for success to be equal to the penalty for mistakes encourages participants to accurately assess each statement, rather than ‘hedge’ by keeping too many or too few clues. Participants completed a comprehension check to ensure that they understood their incentives before they could proceed to the game. Participants were compensated \$1 for training.

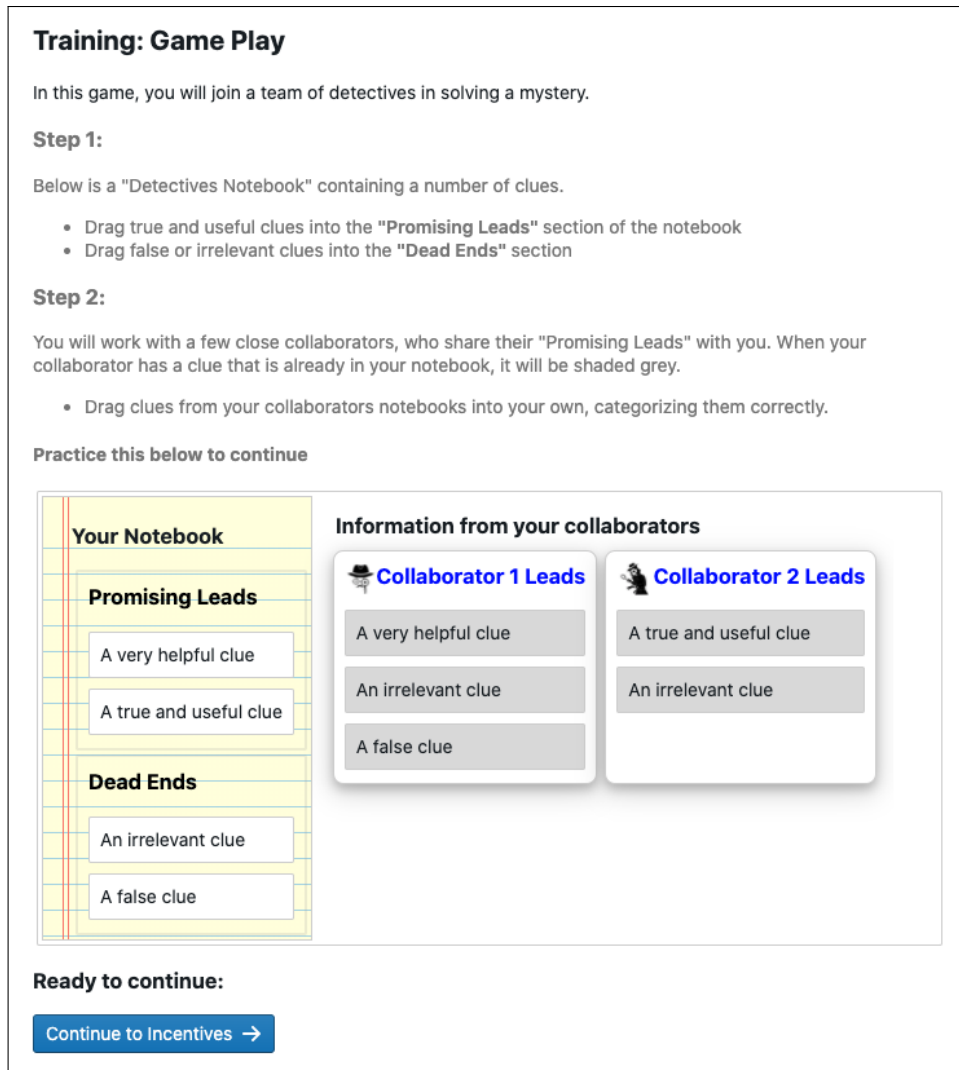


Figure S9: Training screen 1: How to use the game interface

S3.2.3 Game introduction

After completing training (taking between 2 and 4 minutes), participants entered a waiting room until enough players had completed training and were ready to play.² The training was oversubscribed so that if some participants were unable to complete the training the game could still launch.

When the game filled, players were assigned to locations in their social network. Each individual was given a Detective's Notebook with four randomly assigned clues in the Promising Leads section. They were shown a "Police Bulletin" (Fig. S11) which gave them background information about the mystery and reminded them of their task. They had 60 seconds to view this information and orient to the task.

S3.2.4 Playing the game: Exchanging clues

When the game launched, the police bulletin was replaced with the promising leads sections of three collaborators' notebooks, showing the participants a total of 16 unique clues at the start of the game. Individuals at corresponding positions in each social network were given clues that were as similar as possible while allowing for the intervention. These are shown for players in the treatment and control conditions in Figs. S12 and S13 respectively.

The game was played in real-time over 8 minutes. When a participant made any change to their promising leads, their neighbors immediately saw the change on their own screen. The starting clues of every individual were recorded,

²Participants in each block were randomized into two sets of 40, each of which was then split again upon launch. This was due to the history of how the game was developed, and how the implementation of the game and the conditions of the experiment evolved over time. In future implementations, it would be simpler to have one condition per group, although the results would be identical.

Training: Incentives

Your payment is calculated as follows:

Participation

Completing training:	+\$1.00
Playing the Game:	+\$1.00

Individual Bonus

Correct "Leads":	+\$0.10 each
Incorrect "Leads":	-\$0.10 each
Correct "Dead Ends":	\$0.00 each
Incorrect "Dead Ends":	\$0.00 each
	(Up to +\$1.00 total)

Team Bonus

Average of individual bonuses:	Up to +\$1.00
Total:	Up to +\$4.00

Comprehension Check (1 of 2)

Jane has 1 correct lead, 1 correct dead end, and 2 incorrect dead ends.

Jane's Notebook

Promising Leads

A true and useful clue ☒

Dead Ends

A very helpful clue ☐

This clue is probably irrelevant ☒

An important lead ☐

What is Jane's Individual Bonus?

☐ \$0.00
 ☐ +\$0.10
 ☐ +\$0.20
 ☐ +\$0.40

[← Back to Game Play](#)
[Continue to game lobby →](#)

Figure S10: Training screen 2: Incentives and comprehension check

The game will begin in:

0:49

Your Notebook

Promising Leads

Mills has a tattoo of a flag.

Bennet was seen at Kensington House.

Law enforcement is seeking a handsome man.

A witness thought they saw the diamond in a silver BMW.

Dead Ends

POLICE BULLETIN

Theft of DIAMOND from KENSINGTON HOUSE

FRI JUL 03 2020

A burglary at Kensington House includes the loss of a priceless diamond.

Detectives with Promising Leads are asked to compare notes with three collaborators.

Determine which leads are true and useful, and which are dead ends: lies, mistakes, or just irrelevant.

Don't trust any clue. They could all be true, but then they could all be lies.

You will have 8 minutes to determine who did the burglary and how. Then I'll expect your answer.

p.s. This is a high-profile theft, and your bonuses are on the line. Remember, you get paid for leads you get right, but I'll dock your pay for mistakes.

Figure S11: Exposition: Players were introduced to the case and reminded of their task when the game launched.

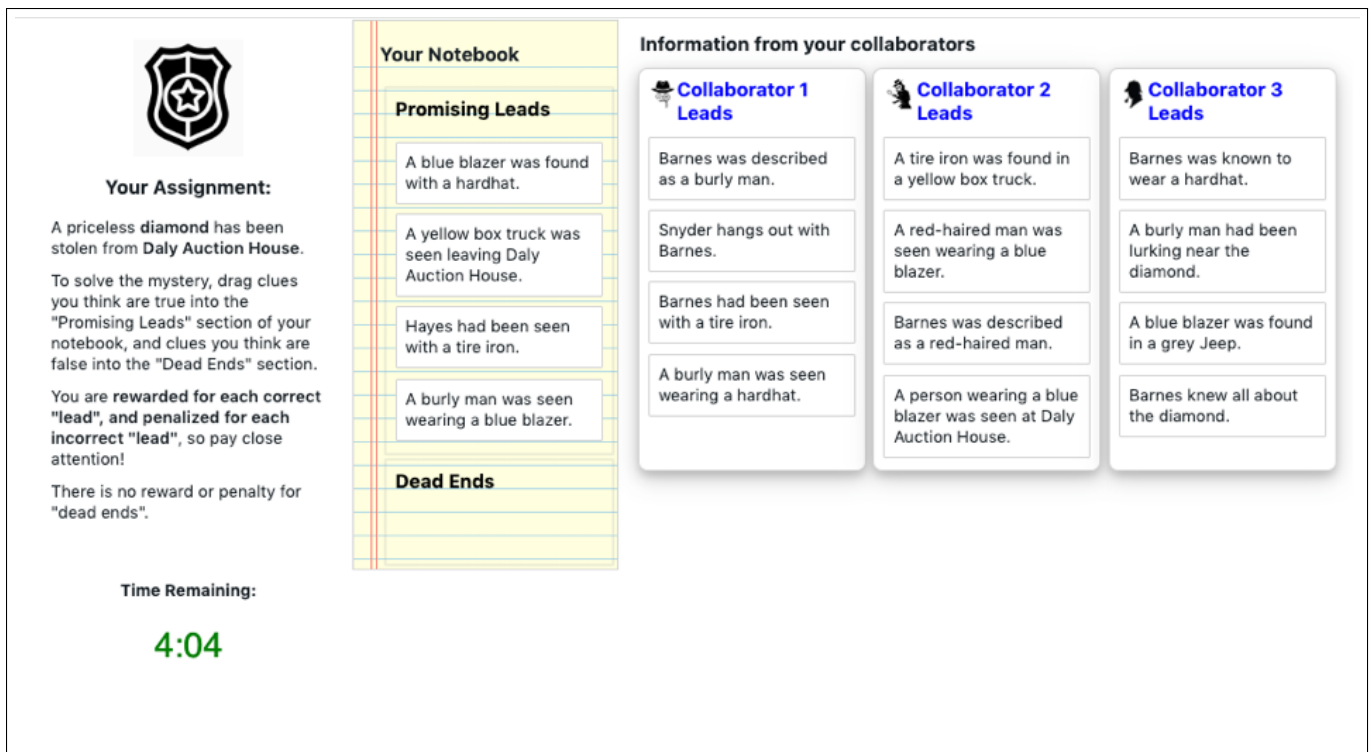


Figure S12: Player interface with interdependent clue set

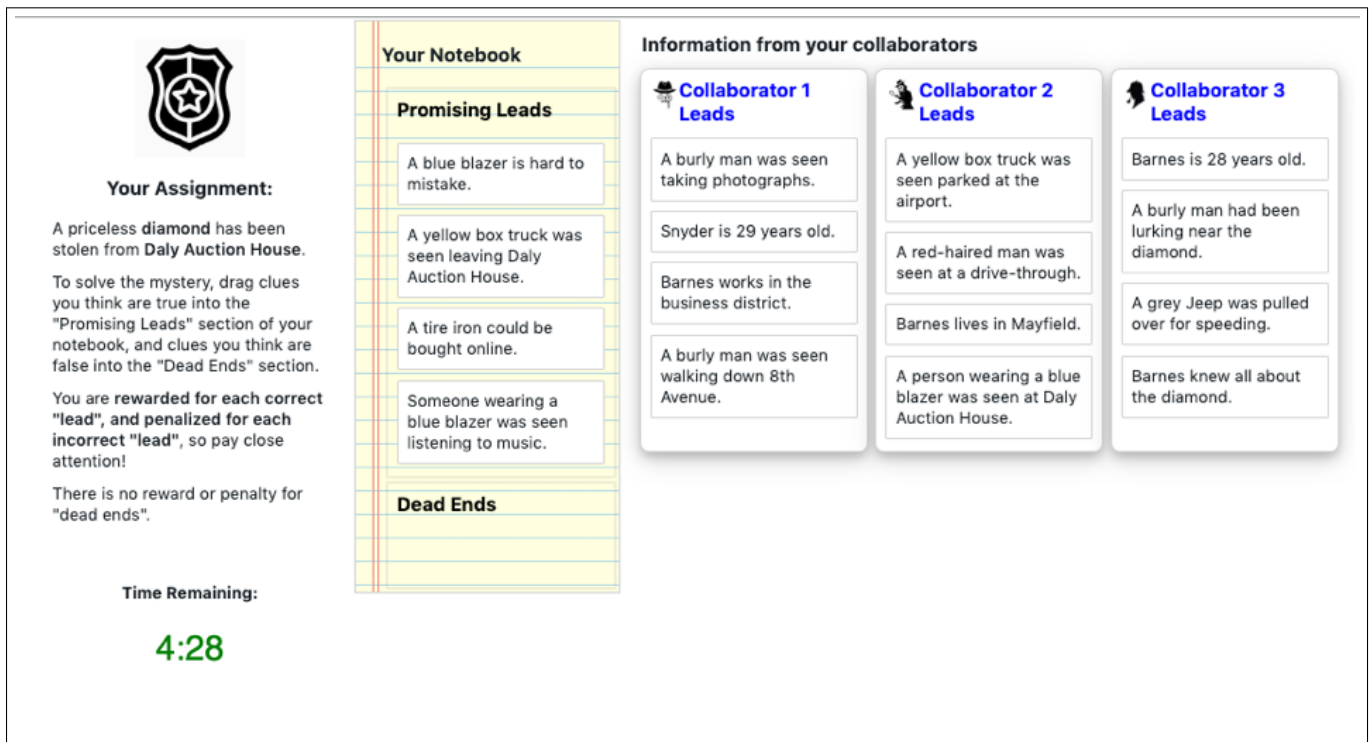


Figure S13: Player interface with matching independent clue set

and every change to every player's detective notebook was logged, such that the state of every player's notebook can be reconstructed at each moment in the game. Participants were compensated \$1.00 for playing the game.

S3.2.5 Post-game survey: Making the case

Following the game, participants were asked to assess (using a slider) how likely it was that certain individuals referenced in the game were the burglar, and how likely it was that they used various tools, vehicles, and disguises in the task. The first few of these questions are shown in Fig. S14. Sliders were labeled from Extremely Unlikely to Extremely Likely, and responses were recorded on a scale from 0 to 100. Participants were also asked to assess their confidence in their solution, and to estimate the level of consensus among their team, as shown in Fig. S15.

Your Notebook

Promising Leads

Fox owns a white Lexus.

The case for the bracelet might have been opened using a plasma cuttler.

Snyder was described as a man with a large adam's apple.

Warren knew all about the bracelet.

Dead Ends

Simmons was known to wear a blue blazer.

A witness thought they saw the bracelet in a blue Toyota Yaris.

Phillips was described as a man with buck teeth.

A pipe cutter was found with a pneumatic drill.

Make the case

Your team has narrowed down the clues to just a few suspects and burglary methods.

How likely do you think it is that the burglar had the following last names?

Extremely unlikely

Somewhat unlikely

Neither likely nor unlikely

Somewhat likely

Extremely likely

Bennet

Campbell

Warren

How likely do you think it is that the burglar fits the following descriptions?

Extremely unlikely

Somewhat unlikely

Neither likely nor unlikely

Somewhat likely

Extremely likely

a hollow-cheeked man

a teenage man

Figure S14: Post-game survey: Make the case for who committed the burglary

Your experience

How confident are you in your solution to the mystery as a whole? (Click the slider below to enter a value. 0% implies no confidence, 100% implies complete confidence.)

0%

25%

50%

75%

100%

What fraction of your team do you think shares your solution? (Click the slider below to enter a value. 0% implies that no other members of your team agree, 100% implies that all other members of your team agree with you.)

0%

25%

50%

75%

100%

Submit

Figure S15: Post-game survey: Assess your confidence and estimate consensus

After “Making the Case”, individuals were told that they were part of an experimental condition in which none

of the clues were false, and they were rewarded \$0.10 for each clue in the promising leads section of their notebook, along with \$0.10 for each of the average number of clues their teammates had categorized as a promising lead. Participants were given a completion code to collect their bonuses, and an opportunity to report any problems with the game or describe their strategy. Many reported that they had enjoyed the game, and would like to play again.

S3.3 Experimental manipulation/stimulus

Clues were constructed from a bank of concepts (11 stolen objects, 11 crime scenes, 15 suspect names, 10 descriptions, 10 articles of clothing, 10 tools, and 10 vehicles) and a set of relationships (e.g. “{Name} owns a {vehicle}”, “A witness thought they saw {stolen object} in {vehicle}”) that formed a complete network between all concepts. This pool is sufficient to generate $11^2 * \binom{15}{3} * \binom{10}{2}^4 \approx 2.25 \times 10^{11}$ different mysteries.

S3.3.1 Constructing the bank of clue concepts

The bank of concepts was constructed by starting with a pool of 403 candidate concepts including names, clothing, vehicles, etc. A pretest survey was conducted in which Amazon Mechanical Turk workers rated how likely each candidate concept was to have been used in a generic burglary. Individuals saw a subset of the concepts and were asked to give their gut reactions using a slider from Extremely Unlikely to Extremely Likely, as illustrated in Fig S16. In total, 139 participants rated each of the 403 candidate concepts between 20 and 30 times. Participants in the pretest were paid \$1.25 for a task which took each participant an average of about 4 minutes.

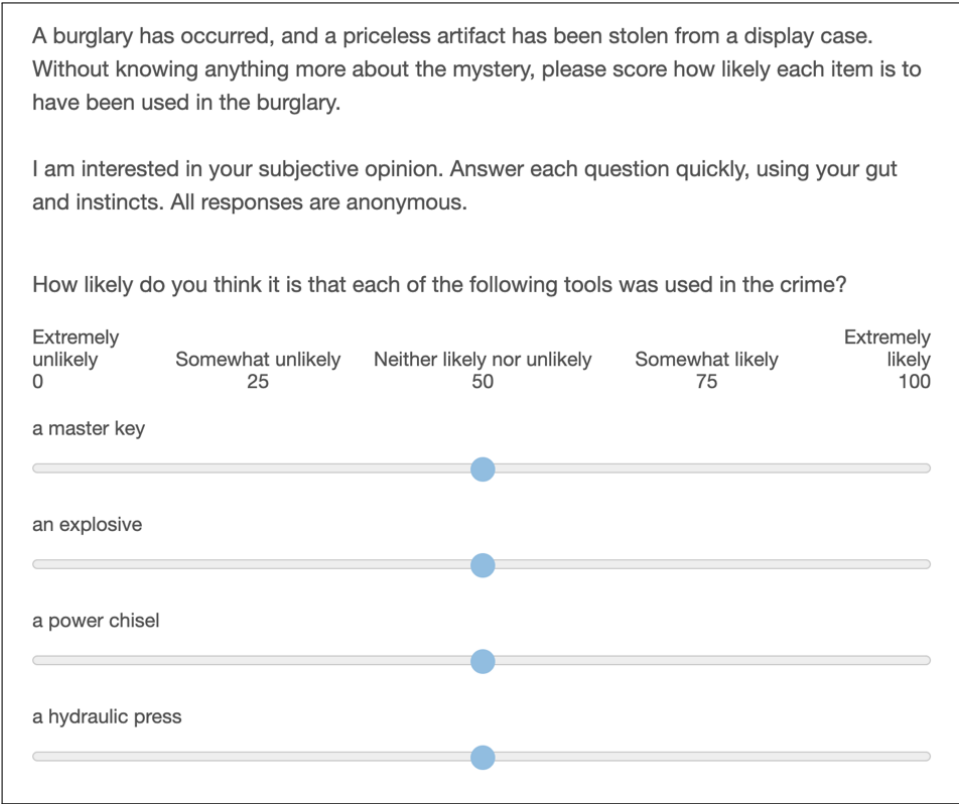


Figure S16: Pretesting the perceived likelihood of each concept being involved in a crime

The pool of candidate names in the pretest represents the subset of the 200 most popular last names in the United States with a racial composition of between 50% and 80% ‘White’, as recorded in the 2000 US census. This selection is made to minimize the possibility of racial biases in the results. Additionally, names which are also common first names were excluded (e.g. “Stewart” or “Ross”) as were names which also serve as descriptors or adjectives in other clues (e.g. “Green”, “White”, or “Young”).

The remaining candidate concepts were written such that they would be as independent from one another as possible (e.g. I do not include both “a fat man” and “an overweight man” as these are synonymous, nor both “an old man” and “a man with grey hair” as these are perceived to go together.)

From the pretest results, I selected a subset of concepts that were perceived to be as likely as one another to be used in a burglary. (This helps to ensure that we do not see games in which all participants adopt “a set of lock picks” as a tool in the burglary, and reject “craft scissors”, just because lock picks are easier to imagine being used in a burglary.) The final selection was made by taking the subset of beliefs that minimized the difference in mean value of pretest survey responses when responses are normalized for each individual, and cross-checking against the means of the raw responses.

A similar pretest survey was conducted to select ‘spur’ clue concepts from a pool of candidates.

Table S1: Clue concept bank. Each element is perceived to be equally likely to have been involved in a burglary.						
CrimeScene	StolenObject	Suspect	Clothing	Appearance	Tool	Vehicle
the art museum	the painting	Collins	a pair of overalls	a long-haired man	a hacksaw	a yellow box truck
the Pine Street Gallery	the statue	Hawkins	a wool hat	a pot-bellied man	a serrated knife	a blue Chevrolet Corvette
Kensington House	the relic	Mills	a blue denim jacket	a partially-bald man	a set of hex keys	a green Mazda 3
the Asper Casino	the bracelet	Cooper	a tracksuit	a grey-haired man	a masonry drill	a silver BMW
the Danforth Hotel	the antique	Moore	a pair of skinny-jeans	a short man	a circular saw	a silver VW Jetta
Knight Secure Storage	the necklace	Bennet	a black scarf	a well-groomed man	a blowtorch	a black Hummer
the Daly Auction House	the watch	Mitchell	a motorcycle helmet	a man with sideburns	an impact wrench	a white Ford Fusion
the Kentwood Mansion	the diamond	Stevens	a black leather jacket	a heavily-scarred man	a tire iron	a blue Toyota Yaris
the Dalhoff Estate	the opal	Wagner	a pair of ripped jeans	a blonde-haired man	a pipe cutter	a white Toyota Avalon
the Darrowby Country Club	the crystal	Edwards	a blue long sleeve shirt	a handsome man	a sledgehammer	a blue Honda Fit
DeRolfe Jewelers	the jewel	Rice				
		Roberts				
		Daniels				
		Warren				
		Sullivan				

Table S2: Interdependent clue connections together create a complete knowledge graph.

{StolenObject.1} was kept in a case at {CrimeScene.1}	{Suspect.2} had been seen with {Tool.2}
{Suspect.1} was seen at {CrimeScene.1}	{Suspect.2} owns {Vehicle.1}
{Suspect.2} was seen at {CrimeScene.1}	{Suspect.2} owns {Vehicle.2}
{Suspect.3} was seen at {CrimeScene.1}	{Suspect.3} was known to wear {Clothing.1}
A person wearing {Clothing.1} was seen at {CrimeScene.1}	{Suspect.3} was known to wear {Clothing.2}
A person wearing {Clothing.2} was seen at {CrimeScene.1}	{Suspect.3} was described as {Appearance.1}
{Appearance.1} was seen at {CrimeScene.1}	{Suspect.3} was described as {Appearance.2}
{Appearance.2} was seen at {CrimeScene.1}	{Suspect.3} had been seen with {Tool.1}
Evidence at {CrimeScene.1} indicates the use of {Tool.1}	{Suspect.3} had been seen with {Tool.2}
Evidence at {CrimeScene.1} indicates the use of {Tool.2}	{Suspect.3} owns {Vehicle.1}
{Vehicle.1} was seen leaving {CrimeScene.1}	{Suspect.3} owns {Vehicle.2}
{Vehicle.2} was seen leaving {CrimeScene.1}	{Clothing.1} was found with {Clothing.2}
{Suspect.1} knew all about {StolenObject.1}	{Appearance.1} was seen wearing {Clothing.1}
{Suspect.2} knew all about {StolenObject.1}	{Appearance.2} was seen wearing {Clothing.1}
{Suspect.3} knew all about {StolenObject.1}	{Clothing.1} was found with {Tool.1}
A person wearing {Clothing.1} had been seen lurking near {StolenObject.1}	{Clothing.1} was found with {Tool.2}
A person wearing {Clothing.2} had been seen lurking near {StolenObject.1}	{Clothing.1} was found in {Vehicle.1}
{Appearance.1} had been lurking near {StolenObject.1}	{Clothing.1} was found in {Vehicle.2}
{Appearance.2} had been lurking near {StolenObject.1}	{Appearance.1} was seen wearing {Clothing.2}
The case for {StolenObject.1} might have been opened using {Tool.1}	{Appearance.2} was seen wearing {Clothing.2}
The case for {StolenObject.1} might have been opened using {Tool.2}	{Clothing.2} was found with {Tool.1}
A witness thought they saw {StolenObject.1} in {Vehicle.1}	{Clothing.2} was found with {Tool.2}
A witness thought they saw {StolenObject.1} in {Vehicle.2}	{Clothing.2} was found in {Vehicle.1}
{Suspect.2} hangs out with {Suspect.1}	{Clothing.2} was found in {Vehicle.2}
{Suspect.1} hangs out with {Suspect.3}	{Appearance.2} was seen with {Appearance.1}
{Suspect.1} was known to wear {Clothing.1}	{Appearance.1} was seen with {Tool.1}
{Suspect.1} was known to wear {Clothing.2}	{Appearance.1} was seen with {Tool.2}
{Suspect.1} was described as {Appearance.1}	{Appearance.1} was seen driving {Vehicle.1}
{Suspect.1} was described as {Appearance.2}	{Appearance.1} was seen driving {Vehicle.2}
{Suspect.1} had been seen with {Tool.1}	{Appearance.2} was seen with {Tool.1}
{Suspect.1} had been seen with {Tool.2}	{Appearance.2} was seen with {Tool.2}
{Suspect.1} owns {Vehicle.1}	{Appearance.2} was seen driving {Vehicle.1}
{Suspect.1} owns {Vehicle.2}	{Appearance.2} was seen driving {Vehicle.2}
{Suspect.3} hangs out with {Suspect.2}	{Tool.1} was found with {Tool.2}
{Suspect.2} was known to wear {Clothing.1}	{Tool.1} was found in {Vehicle.1}
{Suspect.2} was known to wear {Clothing.2}	{Tool.1} was found in {Vehicle.2}
{Suspect.2} was described as {Appearance.1}	{Tool.2} was found in {Vehicle.1}
{Suspect.2} was described as {Appearance.2}	{Tool.2} was found in {Vehicle.2}
{Suspect.2} had been seen with {Tool.1}	{Vehicle.2} was found near {Vehicle.1}

Table S3: Filler element bank. Each element equally influences the perception that an associated concept was involved in a burglary.

suspectAge	suspectConviction	suspectMeans	suspectMotive	suspectTattoo
33 years old	drug possession	was trained as a welder	has paid hush-money to a former lover	a compass
37 years old	fraud	installs security systems	has family connections to organized crime	a bear
in their early 30's	drug distribution	was trained as a goldsmith	is deep in payday-loan debt	a heart
in their mid 20's	running a Ponzi scam	has worked as an automotive repossession agent	has an expensive drug habit	a flower
in their late 20's	embezzlement	has worked as a security guard	has a gambling addiction	a clock
in their late 30's	identity theft	worked at a pawn shop	wrote a revolutionary manifesto	a star
in their early 20's	perjury	has worked as an armored car driver	had been involved in gang activity	a dog
29 years old	shoplifting	has worked for an import/export company	has large gambling debts	a crown
36 years old	arson	worked for an alarm company	has a heroin addiction	a flag

appearanceInjury	appearanceRemoved	appearanceReported	appearanceStreet	appearanceWanted
a broken arm	a museum	waiting in a dark alley	Maple Avenue	Law enforcement is seeking
a fractured kneecap	a public library	shouting at 11pm	Lincoln Boulevard	Officers are asking questions about
a concussion	a bar	sitting in a tree	Chestnut Street	Private security companies have been warned to...
a fractured rib	a restaurant	painting graffiti	Church Street	Airport security has been asked to look out for
minor burns	a residence	vandalizing a vending machine	Hill Street	Police are interviewing witnesses about
a drug overdose	a party	carrying a large bag	Ninth Avenue	Information is wanted about

carBehavior	carBuy	carDamage	carEnterprise	carTicketed
driving after midnight	in a wholesale auction	a broken headlight	a club	an expired registration
with someone sleeping in the back seat	at a police auction	a broken grill	a massage parlor	parking in a loading zone
with darkly tinted windows	from a classified ad	damaged suspension	a strip mall	driving without headlights
parked in a lot for multiple nights	at an estate sale	a missing wing mirror	a laundromat	a broken tail light
taking the back streets	from a used-car salesmen	the airbags deployed	a delicatessen	illegal parking
with its hood up on the roadside	from a junk-yard	a broken axle	a hotel	running a stop sign

clothingActivity	clothingDamage	clothingDiscoverer	clothingFootage	clothingWith
pacing back and forth	cut into pieces	a gym owner emptying abandoned lockers	at a bus stop	a list of tools
entering a machine room	with tire marks on it	a dockworker moving shipping pallets	in the woods	a home-made electronic device
pulling an object out of a gutter	with frayed edges	a store worker breaking down cardboard boxes	in the park	a pair of rubber gloves
looking through binoculars at night	burned in a fire	a journalist uncovering a story	at a campsite	an inter-city train schedule
getting into a taxi	discolored with bleach	a postal worker emptying a mailbox	on a bridge	the stub of a bus ticket
climbing on a bridge	caked in mud	theater staff cleaning up after a movie	on the golf course	an envelope containing GPS coordinates

toolDamage	toolFound	toolUse	toolWith	toolrandom
showing signs of misuse	buried in debris	access maintenance crawlspaces	with a pair of work gloves	could be used by one person
with minor damage	in an abandoned house	open an upper-story window	with safety features removed	has been used in prior burglaries
with burn marks	in a trash compactor	bypass an alarm system	that had been painted black	could be concealed in a backpack
disassembled into pieces	in a garage	deactivate a motion sensor	with gunpowder residue	leaves distinctive marks if used carelessly
covered in sawdust	in a creek	disassemble an alarm panel	wrapped in newspaper	is often used by thieves
that had been damaged	beside a road	circumvent a lock	wrapped in tape to make it quieter	was shown in news coverage of another burglary
falling from a height				

Table S4: Independent clue connections break relationships between analysis clues

{StolenObject_1} was kept in a case at {CrimeScene_1}	A constable noticed {Appearance_1} on {appearanceStreet_1}
{Suspect_1} was seen at {CrimeScene_1}	A constable noticed {Appearance_2} on {appearanceStreet_2}
{Suspect_2} was seen at {CrimeScene_1}	{Tool_1} was found {toolWith_1}
{Suspect_3} was seen at {CrimeScene_1}	{Tool_2} was found {toolWith_2}
A person wearing {Clothing_1} was seen at {CrimeScene_1}	{Vehicle_1} was recently purchased {carBuy_1}
A person wearing {Clothing_2} was seen at {CrimeScene_1}	{Vehicle_2} was recently purchased {carBuy_2}
{Appearance_1} was seen at {CrimeScene_1}	{Suspect_1} is {suspectAge_1}
{Appearance_2} was seen at {CrimeScene_1}	{Suspect_2} is {suspectAge_2}
Evidence at {CrimeScene_1} indicates the use of {Tool_1}	{Suspect_3} is {suspectAge_3}
Evidence at {CrimeScene_1} indicates the use of {Tool_2}	{Clothing_1} was discovered by {clothingDiscoverer_1}
{Vehicle_1} was seen leaving {CrimeScene_1}	{Clothing_2} was discovered by {clothingDiscoverer_2}
{Vehicle_2} was seen leaving {CrimeScene_1}	{Appearance_1} was reported {appearanceReported_1}
{Suspect_1} knew all about {StolenObject_1}	{Appearance_2} was reported {appearanceReported_2}
{Suspect_2} knew all about {StolenObject_1}	{Tool_1} could be used to {toolUse_1}
{Suspect_3} knew all about {StolenObject_1}	{Tool_2} could be used to {toolUse_2}
A person wearing {Clothing_1} had been seen lurking near {StolenObject_1}	{Vehicle_1} was ticketed for {carTicketed_1}
A person wearing {Clothing_2} had been seen lurking near {StolenObject_1}	{Vehicle_2} was ticketed for {carTicketed_2}
{Appearance_1} had been lurking near {StolenObject_1}	{Suspect_1} {suspectMeans_1}
{Appearance_2} had been lurking near {StolenObject_1}	{Suspect_2} {suspectMeans_2}
The case for {StolenObject_1} might have been opened using {Tool_1}	{Suspect_3} {suspectMeans_3}
The case for {StolenObject_1} might have been opened using {Tool_2}	{Clothing_1} was found with {clothingWith_1}
A witness thought they saw {StolenObject_1} in {Vehicle_1}	{Clothing_2} was found with {clothingWith_2}
A witness thought they saw {StolenObject_1} in {Vehicle_2}	{Appearance_1} was treated for {appearanceInjury_1}
{Suspect_1} has a tattoo of {suspectTattoo_1}	{Appearance_2} was treated for {appearanceInjury_2}
{Suspect_2} has a tattoo of {suspectTattoo_2}	An FBI agent found {Tool_1} {toolFound_1}
{Suspect_3} has a tattoo of {suspectTattoo_3}	An FBI agent found {Tool_2} {toolFound_2}
A policeman saw someone in {Clothing_1} {clothingActivity_1}	An officer identified {Vehicle_1} at {carEnterprise_1}
A policeman saw someone in {Clothing_2} {clothingActivity_2}	An officer identified {Vehicle_2} at {carEnterprise_2}
{appearanceWanted_1} {Appearance_1}	{Suspect_1} has a prior conviction for {suspectConviction_1}
{appearanceWanted_2} {Appearance_2}	{Suspect_2} has a prior conviction for {suspectConviction_2}
{Tool_1} {toolrandom_1}	{Suspect_3} has a prior conviction for {suspectConviction_3}
{Tool_2} {toolrandom_2}	Forensics identified {Clothing_1} {clothingDamage_1}
{Vehicle_1} was reported {carBehavior_1}	Forensics identified {Clothing_2} {clothingDamage_2}
{Vehicle_2} was reported {carBehavior_2}	{Appearance_1} was forcibly removed from {appearanceRemoved_1}
{Suspect_1} {suspectMotive_1}	{Appearance_2} was forcibly removed from {appearanceRemoved_2}
{Suspect_2} {suspectMotive_2}	A forensics report contained {Tool_1} {toolDamage_1}
{Suspect_3} {suspectMotive_3}	A forensics report contained {Tool_2} {toolDamage_2}
Someone wearing {Clothing_1} was seen on security footage {clothingFootage_1}	{Vehicle_1} was found with {carDamage_1}
Someone wearing {Clothing_2} was seen on security footage {clothingFootage_2}	{Vehicle_2} was found with {carDamage_2}

S3.3.2 Assembling sets of clues for use in games

This experiment manipulates the structure of clues within the detective game, to create an “interdependent” condition in which the clues interact strongly with each other, and an “independent” condition that limits those interactions while preserving as much similarity with the treatment condition as possible.

Clues were constructed in three waves. The first wave was identical for both conditions, and is illustrated in Fig. S17. Clues were created that link ‘hub’ concepts (including a crime scene and a stolen object) to ‘rim’ concepts (including three suspects, two articles of clothing, two physical descriptions, two tools, and two vehicles). For example “**Hayes** was seen at the **Daly Auction House**” or “The case for the **diamond** might have been opened using a **circular saw**”. “Spoke” clues were independent of one another, as they could only interact via association with the crime scene and stolen object – items that were known in advance to be relevant to the mystery. There were 11 rim concepts and 2 hub concepts, and so 22 spoke clues.

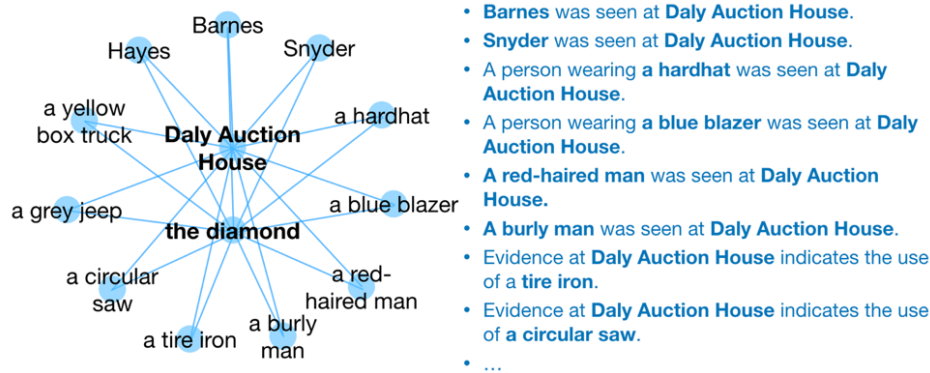


Figure S17: Wave 1 of clue generation created “spoke” or analysis clues used in both treatment and control games. “Spoke” clues connected rim concepts to hub concepts.

In the interdependent condition, the second wave of clue construction created “cross-link” clues, which connected each of the spoke clues to one another (e.g. “**Hayes** owns a **circular saw**”). These cross-link clues create interdependence between the spoke clues, and allow for clues to logically support one another (e.g. if I believe that “A burly man was seen at the Daly Auction House” and that “Barnes is a burly man”, then I am more receptive to the idea that “Barnes was seen at the Daly Auction House”). A cross-link clue connected each of the 11 rim concepts to the other rim concepts, for a total of 55 unique cross-link clues, as shown in Fig. S18.

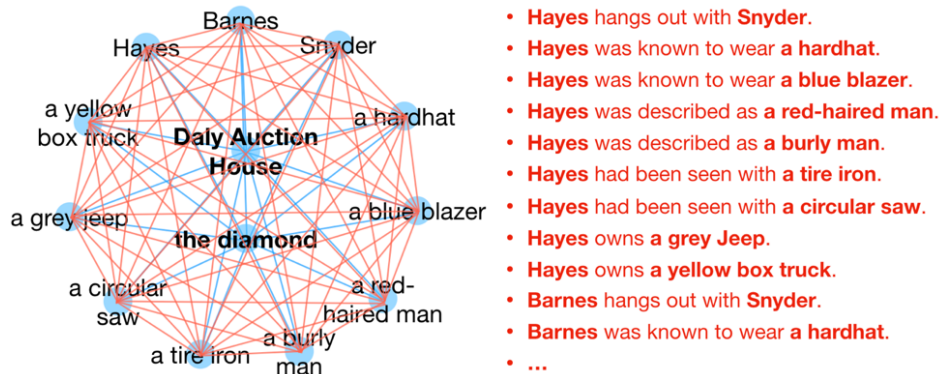


Figure S18: Wave 2 of clue generation created “cross-link” clues for the interdependent condition. Cross-link clues connect rim concepts to one another.

In the independent condition, the second wave of clue construction created “spur” clues that connected to the rim concepts, but did not connect to other clues (Fig S19). There were the same number of spur clues in the independent condition as cross-link clues in the interdependent condition: 55. By connecting to the rim concepts (rather than being disconnected altogether) these clues help separate the effect of interdependence manifest as logical relationships between clues from the effect of the frequency of each rim concept in the set of clues. The content of the spur clues was selected in pretest to have a uniform impact on participants judgement of the rim element to which they connect.

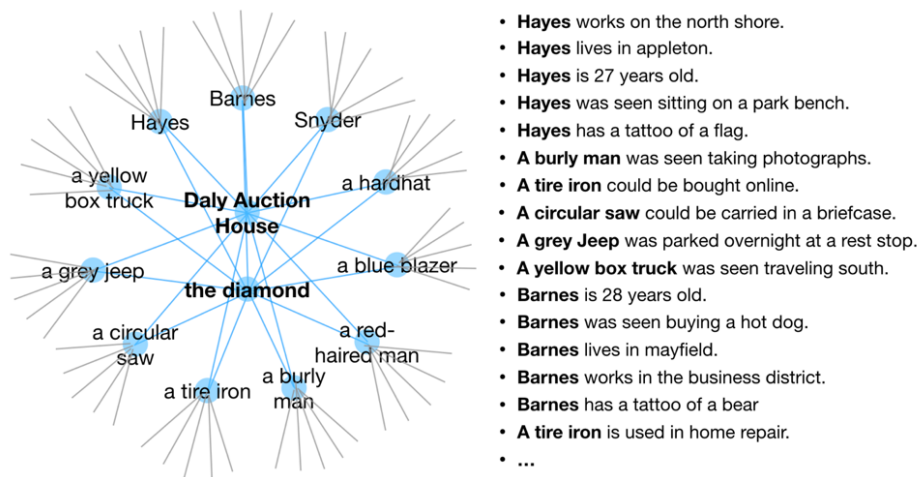


Figure S19: Wave 2 of clue generation created “spur” clues for the independent condition. Spur clues filled the place of cross-link clues without creating links between rim concepts. Spur clues still allowing for multiple exposures to rim concepts.

The first and second waves of clue construction created 77 unique clues. As there were 20 individuals in each treatment within each game, 80 clues were needed to give each individual 4 starting clues. The third wave of clue construction filled the 3 remaining spaces with the clue connecting the crime scene to the stolen object (e.g. The **diamond** was stolen from the **Daly Auction House**.) This was redundant information, as all participants were told this at the start of the game.

S3.4 Data collection and measurements

Sufficient data was collected to replicate the state of the game at any point during game play, and to observe every action taken by every player.

S3.4.1 Recording player actions

In addition to the social network structure and the initial assignment of clues to positions within the social network, the following information was recorded:

1. Each drag event that resulted in a change in a player’s notebook (i.e. addition of a clue to a notebook section OR change of order within a notebook section) was logged. Logging information includes the ID of the clue being dropped, the source for the drag event (i.e. the exposing player or notebook the belief came from), the destination for the drag event (i.e. the notebook the clue is being dragged into), the position within the destination notebook that the clue moved into (i.e. its index in the list) and the time at which the drop event occurred.
2. The final state of all detectives’ notebooks was recorded. Together with the initial state, this provides a check that all events were logged properly.
3. Each individual provided a self-report of the degree to which they believed each of the 11 “rim” concepts to be connected to the crime, collected using an empty slider from “Extremely Unlikely” to “Extremely Likely”. Slider positions were captured as an integer value between 0 and 100.
4. Individuals reported their confidence in their solution on a scale from 0 to 100 using a blank slider.
5. Individuals reported the fraction of their team they thought shared their solution on a scale from 0 to 100 percent, using a blank slider.
6. Individuals reported their Age, Education, Gender, and feedback on the game.

S3.4.2 Choice of summary statistics

The **similarity between individuals’ self-reported beliefs** was measured using Pearson’s correlation coefficient. As others have identified, [7] correlation is a natural measure when we have a fixed number of continuous measures of each subject. It is readily interpretable, and the fixed range (-1,1) maps to intuitive understandings of similarity and difference.

The **similarity between individuals’ behavior** was measured using the Phi coefficient. This measure corresponds to Pearson correlation when values are binary, and has the same interpretable (-1,1) range. The phi coefficient is appropriate for a universe with a finite number of beliefs, but would be less appropriate as the number of adopted beliefs becomes a very small fraction of the total number of possible beliefs.

The **alignment of the population along a “left-right axis”** was measured as the percent of variance present in the first principal component, using singular value decomposition. This measure corresponds to the notion of “constraint” articulated by Dimaggio et al. ([4]). In their paper they describe Chronbach’s alpha and the PCA measure both providing similar measures of constraint. I have chosen the PCA measure here as more interpretable, as it maps better to our intuitive understanding of a political spectrum.

The **within-camp and across-camp similarities** were measured as the 5th and 95th percentile similarities according to the similarity measures above. There are a number of different measures in the literature that try to capture the notion that with polarization, the most similar individuals become more self-similar, and the least similar individuals move further away from one another. The fact that no single measure has emerged as the leader hints at problems with each. When the identities of camps are already known the difference of means between groups can be used ([3],[4]). Variance (see [2],[4]) captures heterogeneity between individuals, but not clustering into camps. Kurtosis ([2],[4]) is predicated on a bimodal distribution. The “gap” statistic ([7]) is one of dozens of ways of assessing the quality of a machine learning clustering algorithm.

As I do not need to identify the groups themselves, or compare to external datasets, it is sufficient to merely report what each of these other measures is trying to approximate: the similarity that is found within groups, and that which is found across groups. As I am only interested in the relative differences between conditions (or changes over time) then I can arbitrarily designate a threshold for which comparisons will be considered “within-group” or “across groups”. This provides a much more intuitive demonstration of increasing polarization than the measures found in above literature.

The closer the chosen thresholds are to the tails of the distribution, the more conservative the claim that the comparisons beyond this threshold are appropriately “within” or “across” groups. At the same time, we need enough samples included in the set to minimize noise due to the finite number of comparisons. In this 20-participant social network, the 95th and 5th percentiles correspond to 10 comparisons between individuals. The sensitivity of results to the choice of these thresholds is explored in section S1.3 of this supplement.

For behavioral measures of within and across-camp similarity (i.e. those based upon clues identified as promising leads at the end of the game) these percentiles are sensitive not only to interdependence and network structure, but also to the average level of diffusion of clues. As the average level of diffusion can vary between games due to differences in players’ baseline activity levels, this additional source of noise increases the sample size we would require to see an effect. To compensate for this noise, we can compare the percentiles to what would be expected due to chance, given the same extent of diffusion. To calculate the effect of interdependence and network structure, I first remove some of the noise by subtracting the 5th and 95th percentile values from a shuffled data-set that keeps the number of adopters of each clue and the number of clues adopted by each participant fixed. This makes for an apples-to-apples comparison of the effects of interdependence or network structure. This correction was designed in simulation and included in the preregistration for the experiment.

S3.4.3 Handling missing data

As the game was played in real-time, the effect of a participant ‘dropping out’ during game-play was equivalent to them holding their beliefs fixed for the remainder of the game. As it is impossible to distinguish these two behaviors, I identified a drop-out as any player failing to submit the post-game survey.

When an individual failed to complete the post-game survey, aggregate results for their condition were calculated based upon the remaining players. Aggregate results for the paired comparison conditions were calculated as the average of all same-sized subsets of players in each comparison condition.

S3.5 Preregistration

The preregistration for the experiment included all the code necessary to run and analyze the experiment presented in this paper, such that a direct replication can be conducted from the preregistered materials. The preregistration

is available at <https://osf.io/239ns>. In addition to the analyses presented in this paper, the preregistration also included a secondary analysis of mediating and moderating effects of interdependent diffusion that will be reported in a second paper.

The experiment differed from the preregistered procedure in three minor ways: expanding the recruitment pool, correcting a coding mistake, and formalizing an analysis that was preregistered with only a text description.

1. The preregistered strategy of recruiting individuals with a “sign-up” HIT immediately prior to each game did not scale well to multiple games per day. Instead, I drew on a pool of Mechanical Turk workers from the US and Canada that had been recruited for a previous experiment. As the locations of the panelists were not recorded, this required me to relax the preregistered participant qualifications to include Canadian workers. I felt that this addition would not significantly influence the likelihood of success of the experiment or its generalizability. The phenomenon under study is not expected to behave differently in different populations, and the study does not require any special outside knowledge.
2. The analysis code for comparing end-of-game measures was designed originally to account for dropouts in a (treatment/control) game by averaging over same-sized subsets of the matching (control/treatment) game. However, the code as preregistered did not account for the fact that comparisons should be made between four experimental conditions instead of two. The code was revised to make the correct comparisons.
3. While the interaction analysis was included in the text of the preregistration, the code for computing it was mistakenly omitted. This additional code is included in the experiment repository.

S3.6 Outcomes

S3.6.1 Participants’ experience

Participants were Amazon Mechanical Turk workers who lived in the US or Canada and were over 18 years of age. Workers must have completed at least 100 HITs and have a 90% or better approval rating. Recruitment and compensation were handled using TurkPrime (www.cloudresearch.com).

For blocks 0-2, recruitment took place in the hour preceding a game launch via a timed “sign-up” HIT. This recruitment strategy had been shown to be effective during pilots, but did not scale well when multiple blocks were run during the same day. For blocks 3-29, recruitment took advantage of a panel of workers who had previously indicated willingness to be notified of upcoming games. This panel was expanded through ongoing paid and unpaid recruitment HITs during the period the experiments were run. Panel members were notified via email at the beginning of each day when experiments would take place, and again 10 minutes prior to the launch of a game. At the scheduled game time, HITs were made available to any worker meeting the qualifications, whether they were in the original panel or not.

Participants were compensated \$0.10 for accepting the game HIT, in addition to \$1 for training, \$1 for playing the game, and up to \$2 in bonuses. Participants who trained but were unable to play were eligible to attempt to play again. Those who completed training and entered the game were blocked from participating in future games via an exclusion qualification.

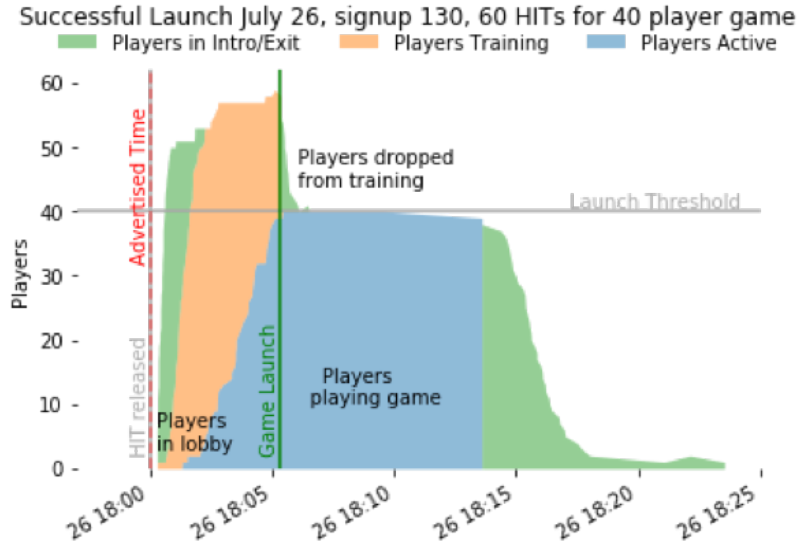


Figure S20: Players active by stage and time. (Data from pilot test.)

The game took about 20 minutes to play, including training, waiting room, and follow-up. The average payout was approximately \$4.00, for an hourly rate of approximately \$12.00/hr. Participants who trained but were unable to play spent about 5 minutes in the platform before they were bumped. They earned \$1.10, for an approximate hourly rate of \$13.20. Fig. S20 shows the number of participants active in different parts of the game at different times for a 2-condition pilot test. There were necessarily some individuals who were dropped from training when the game launched, as the number of individuals who showed up for a game was unpredictable.

S3.6.2 Detailed Results

Over eight minutes of gameplay, participants on average made 28.4 classifications and adopted 16.2 clues as promising leads. Despite the symmetry of the setup, and the fact that there was by design no solution to the mystery, participants came to strongly-held beliefs about which suspect was guilty and how they performed the crime. On average, participants felt that the most likely suspect was almost 50% more likely to be involved in the crime than the least likely suspect, and over half of participants reported confidence in at least one part of their solution of 95% or greater. Table S5 reports the effect sizes for all comparisons made in the experiment.

Table S5: Effect and Interaction Summary

		Effect of Interdependence			Effect of Network Structure		
		Effect Size	p value	90% CI	Effect Size	p-value	90% CI
Within-Camp	- Behavioral	0.026	0.015	(0.0083, 0.045)	.16	1.17e-9	(.13, .19)
Similarity	- Self-Report	-0.0039	0.41	(-0.036, 0.025)	.021	.12	(.0099, .050)
Across-Camp	- Behavioral	0.010	0.22	(-0.012, 0.032)	-.034	.004	(-.054, -.016)
Similarity	- Self-Report	-0.034	0.058	(-0.067, 0.0015)	-.089	6.3e-5	(-.12, -.057)
Alignment w/	- Behavioral	2.13%	0.016	(0.61%, 3.72%)	13.7%	2.1e-11	(.12, .16)
Left-Right Axis	- Self-Report	2.81%	0.022	(0.57%, 5.03%)	7.2%	2.3e-5	(4.8%, 9.8%)

Baseline for effect: Independent clues and non-polarizing social network.

One-sided pairwise T-tests; n=30 pairs; unadjusted p-values

The presented theory does not predict how individuals will feel about their team's performance. Interdependence did not change the perceived consensus among the team by any measurable amount, despite the increase in polarization. However, participants in the interdependent condition did report feeling more confident in their solution (+2.17% p=.045). This is not a particularly large effect, as the results were measured on a 100 point scale.

S4 Resources for replication, extension and reanalysis

S4.1 Conditions of validity

The simulations and effects presented here are only valid when susceptibility to a belief can vary over the same timescale as the diffusion process. This primarily occurs when there are multiple beliefs diffusing in the same social network over the same timescales. If the population is broadly susceptible to a belief before it becomes available for adoption (for example, if international relations are strained, news of war might propagate quickly, as individuals are already susceptible to this idea) then diffusion of that belief will proceed much like the spread of a viral infection. Other beliefs would certainly be spreading at the same time, but they would not be necessary for the adoption of the focal belief.

Conversely, if there are not enough facilitating beliefs in a population to make adoption likely, then the reciprocal facilitation mechanism is unlikely to activate, and diffusion (if it occurs at all) will be merely among those who are initially susceptible.

We can see both of these limiting conditions in simulation by changing the number of beliefs that an individual starts with. Too few, and diffusion stalls in both independent and interdependent simulations. Too many, and susceptibility is a foregone conclusion, and in both independent and interdependent conditions adoption is universal. However, for a broad range of values in between, reciprocal facilitation is the dominant factor in a particular belief's level of adoption.

The mechanisms presented in this paper are also likely to be less important when information spreads from a central source to all individuals, as the agreement cascade mechanism depends upon individuals throughout the network adopting beliefs from one another.

S4.2 Replicating these results

All of the code required to conduct this experiment and all of the data generated by the experiment is available open-source at <https://github.com/JamesPHoughton/interdependent-diffusion>. An exact replication of these results can be run without writing any code. Slight changes to the replication - such as creating new sets of clues, or using a different social network - can be accomplished by changing the experiment's configuration file.

Resources that will help a researcher replicate this analysis include the Empirica documentation (<https://empirica.ly/>) and introductory paper [1], and the Empirica code repository: <https://github.com/empirically/meteor-empirica-core>.

I am also happy to advise replication efforts and answer questions about the code or implementation at the github repository for this experiment: [Repository Issues](#).

S4.3 Extending this research

There are a number of obvious opportunities for extending this research. In this simulation and experiment, individuals do not verify their beliefs against ground-truth. We should hope that in the real world, this occurs at least occasionally, and that when it does, it forms a corrective force against the drivers of polarization. An interesting extension to this experiment would be to test the effects of belief verification on macro-scale outcomes. An experimenter could manipulate the cost to verify information, and the correlation of this cost between clues, or between members of the population, and examine the effects of specialization or general knowledge on collective problem solving.

Another opportunity for extension would be to vary the way information is presented, to mirror that of various social media websites, and to explore which factors tend to amplify the polarizing influence of agreement cascades and reciprocal facilitation.

The detective game experiment allows researchers to study the simultaneous contagion of multiple diffusants on a level playing field; the game could be adapted to many contexts with this requirement.

Resources for extending this research include the Meteor (<https://www.meteor.com/>) and React (<https://reactjs.org/>) documentation, and the above-listed Empirica documentation. Additionally, several third-party developers have experience developing experiments using Empirica and can be engaged to implement modifications.

S4.4 Opportunities for reuse of the data generated by this experiment

This experiment recorded timestamped data on 68,229 adoption decisions made by 2400 individuals, along with instantaneous and historical information participants used to make those decisions. The information originally shown to participants is randomized, and diverse. As a result, there are many opportunities to reuse this data to answer questions about the micro-level processes involved in the adoption of new beliefs.

References

- [1] Abdullah Almaatouq, Joshua Becker, James P Houghton, Nicolas Paton, Duncan J Watts, and Mark E Whiting. Empirica: a virtual lab for high-throughput macro-level experiments. *Behavior Research Methods*, pages 1–14, 2021.
- [2] Delia Baldassarri and Peter Bearman. Dynamics of political polarization. *American sociological review*, 72(5):784–811, 2007.
- [3] Joshua Becker, Ethan Porter, and Damon Centola. The wisdom of partisan crowds. *Proceedings of the National Academy of Sciences*, 116(22):10717–10722, 2019.
- [4] Paul DiMaggio, John Evans, and Bethany Bryson. Have american’s social attitudes become more polarized? *American journal of Sociology*, 102(3):690–755, 1996.
- [5] Noah E Friedkin, Anton V Proskurnikov, Roberto Tempo, and Sergey E Parsegov. Network science on belief system dynamics under logic constraints. *Science*, 354(6310):321–326, 2016.
- [6] Ygor Gallina, Florian Boudin, and Beatrice Daille. KPTime: A large-scale dataset for keyphrase generation on news documents. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 130–135, Tokyo, Japan, October–November 2019. Association for Computational Linguistics.
- [7] Amir Goldberg and Sarah K Stein. Beyond social contagion: Associative diffusion and the emergence of cultural variation. *American Sociological Review*, 83(5):897–932, 2018.
- [8] JD Systems Institute and WWSA. European values study and world values survey: Joint evs/wvs 2017-2021 dataset (joint evs/wvs). dataset version 1.0.0. 2020.
- [9] Kamran Kowsari, Donald E Brown, Mojtaba Heidarysafa, Kiana Jafari Meimandi, , Matthew S Gerber, and Laura E Barnes. Hdltext: Hierarchical deep learning for text classification. In *Machine Learning and Applications (ICMLA), 2017 16th IEEE International Conference on*. IEEE, 2017.
- [10] Kamran Kowsari, Donald E Brown, Mojtaba Heidarysafa, Kiana Jafari Meimandi, , Matthew S Gerber, and Laura E Barnes. Web of science dataset v6. *Mendeley Data*, 2018.
- [11] Melissa A Schilling. A” small-world” network model of cognitive insight. *Creativity Research Journal*, 17(2-3):131–154, 2005.