Analyzing the Effectiveness of Stereotypes

Nanda Kishore Sreenivas and Robin Cohen
Cheriton School of Computer Science, University of Waterloo, Waterloo, ON, Canada

Keywords: Multiagent Systems, Trust Modeling, Stereotypes, e-Marketplaces, Emotional Intelligence.

Abstract: In this paper, we present an agent-based model to understand the effectiveness of stereotpyes in supporting decision making of users in online settings such as e-marketplaces. We define four different agents types that combine information from stereotypes and past interactions in different ways. Through simulation, we find that agents that use their memory primarily and stereotypes as a last resort perform the best. We discuss how this work relates to that of trust modeling in multiagent systems and reflect briefly on how the emotional makeup of a user may influence what is best for decisions about stereotype use.

1 INTRODUCTION

A stereotype is a generalized belief about a group or category of people (Hinton, 2013). Psychologists and social scientists have learned that the use of stereotypes lowers our cognitive effort and helps us to remember and recall information, especially in situations that demand a quick response (Fiske, 1992). On the other hand, the overgeneralization makes many stereotypes inaccurate (Mullen and Johnson, 1990) and the cultural aspect of stereotypes also explains why we hold on to stereotypes even when presented evidence to the contrary (Hinton, 2013). And yet, many studies have shown them to be quite accurate (Jussim et al., 2015). While stereotypes can be explicitly acquired (with the user fully admitting to and aware of their usage), automatic inferencing methods may be needed. In our model, we focus on these implicitly learned stereotypes.

In this paper, we develop an agent-based model (ABM) to understand the effectiveness of stereotypes in supporting decision making with high reward, under various scenarios. ABMs are a great tool to model and analyze such processes and phenomena, and they have been used for decades in social science for that reason (Epstein and Axtell, 1996). We model a marketplace setting where agents must pick a seller offering the highest quality product or service. This setting is inspired by the trust and reputation literature (Zhang and Cohen, 2008; Liu et al., 2013). However, the agent may not always have enough information or past experience to calculate the expected quality of a seller. The agent can fill this missing information in several ways — guessing, stereotypes, reputation (second-hand information), etc.

We focus on stereotypes in this work, and the agents form stereotypes based on the sellers' observable traits (feature). In contrast to researchers focused on arguing that stereotypes are an effective avenue for boostrapping decision making in multiagent systems (Burnett et al., 2013; Taylor et al., 2017), we explore a more comprehensive analysis of the conditions under which the use of stereotypes may or may not be helpful. We use the setting of electronic marketplaces in order to illustrate this decision making process more concretely. We construct four different types of agents with various stereotyping mechanisms that cover the spectrum from highly rational to irrational. For example, agents can rely on stereotypical knowledge only to fill in missing information or recklessly as mental shortcut. The stereotyping is modeled as a regression model where the seller's feature is the independent variable, and the quality is the dependent variable to be predicted. We comment on how effective stereotypes are challenged when seller features and quality are not correlated; we also experiment with forgetting as an exponential decay of memories.

User modeling researchers have proposed the use of stereotypes to assist users in reasoning about peers when there is a lack of current knowledge about the other agent (Kass and Finin, 1988). In this paper, we explain how that use of stereotypical inference can be viewed as a decision under the user's control, clarifying through simulations where reckless use of stereotypes may introduce challenges and where greater re-

Sreenivas, N. and Cohen, R.
Analyzing the Effectiveness of Stereotypes.
DOI: 10.5220/0011623300003393
In *Proceedings of the 15th International Conference on Agents and Artificial Intelligence (ICAART 2023) - Volume 1*, pages 181-188
ISBN: 978-989-758-623-1; ISSN: 2184-433X

liance on the user's own memory of past interactions with other agents may lead to better rewards. We conclude with a brief reflection on how emotional intelligence of users may be modeled in order to refine the use of stereotypes.

2 MODEL

To analyze the effect of stereotypes, we build an agent-based model (ABM) with two different types of agents — sellers and agents. The number of sellers and agents are denoted by N_S and N, respectively. Sellers are denoted by s_i, where $0 \leq i \leq N_S$, and agents are denoted by a_j where $0 \leq j \leq N$.

This is an effort to simulate a marketplace where the sellers offer products and services at varying levels of quality, and the agents try to choose the best ones[1]. In the rest of this section, we discuss the interaction model, sellers, the various types of agents, their parameters, and their behaviors.

2.1 Interaction Model

The system is simulated for T timesteps. In each timestep, the sellers are partitioned into N non-overlapping subsets. The partitions are fully random, and they change every timestep. Each of these subsets is assigned to an agent, and the agent picks one seller from this group. This is done to mimic the dynamic environment of an online marketplace where new and unknown sellers crop up often. The agent and the seller interact, i.e., the agent receives a product of some quality (between 0 and 1), which is the agent's reward. The agent updates its total reward and logs the interaction in its memory.

2.2 Seller

A seller is defined by a 3-tuple $< i, f, q >$. i is simply an identifier, and f is the feature vector represented by an 8-bit binary vector. Each bit indicates the presence (or absence) of some observable trait. This work is an abstract model, so we do not define these traits. In the real world, these could be seller ratings, age on the platform, number of products, etc. We obtain the feature vector by taking the 8-bit binary representation of $i \mod 256$.

Finally, q denotes the quality of a product or service offered by the seller. In this work, we assume the seller offers only one product, and therefore, q can be seen as the quality of the seller. In our model, $0 \leq q \leq 1$ where 0 corresponds to poor quality, and 1 corresponds to perfection. To understand the effect of stereotypes, we would like to model a mapping between a seller's feature (f) and its quality (q). It is important to note that all traits may not have an equal impact on performance (Zell and Lesick, 2021). Sometimes, we humans place weights on traits that have no causal relationship with performance. We define a global variable called the quality influence vector (Q), a list of traits ordered by their relative impact on quality. So, the first trait in Q has more weight on quality than the second and so on. Varying the length of Q from 0 to 8, we can control the relationship between the seller's traits and quality. Q is a simulation parameter, and for a given simulation, the same Q applies to all sellers.

Algorithm 1: Mapping function.

```
1  Function Θ(f, Q):
2      f' ← ∅;
3      l ← 0;
4      if Q = ∅ then
5          q ← U(0,1)
6      else
7          for i ← Q do
8              Append(f', f[i]);
9              l ← l + 1;
10         end
11         temp ← ConvToDecimal(f');
12         q ← temp / pow(2,l) − 1;
13     end
14     return q;
15 end
```

We define a mapping function Θ that takes two arguments — f and Q and returns quality q. This function is outlined in Algorithm 1. If Q is empty, there is no relationship between a seller's feature and their quality. So, the quality is sampled from a uniform distribution between 0 and 1 ($U(0,1)$). If Q is not empty, f is transformed to f' by taking the relevant traits and arranging them in the priority order defined by Q (lines 7–10). Now, f' and Q are of the same length. The binary string f' is then converted to a decimal number (line 11). As noted above, we want the weights to be related to the position of the trait in Q. The binary to decimal transformation ensures that the order of priority defined by Q is implemented because the most significant bits get a higher weight. Finally, we scale it to the range $[0,1]$ by dividing it by the highest possible number (given the binary string's length) to get the quality (a normalization step).

The *feature quality relationship* (FQR) of sellers depends on the length of Q, denoted by $|Q|$. If Q is empty, there is no relationship between the observed traits and quality, which is an interesting scenario for experimentation, and we denote this scenario as

[1]While a marketplace setting (similar to what is used in prior work (Burnett et al., 2013; Taylor et al., 2017)) helps to ground our discussion in this paper, our model is general enough to apply to other contexts.

FQR-0. In this case, the seller's quality is sampled from $U(0,1)$, a uniform distribution between 0 and 1. When $|Q| > 0$, there exists a relationship between the traits and seller quality, and we denote this scenario as FQR-1. Even as $|Q|$ varies, the distribution of quality in a seller population remains symmetrical, with a mean of 0.5 [2]. The sellers are non-strategic, and they do not try to adapt or learn about the agents in the system.

2.3 Basic Agent

A basic agent is defined by a 4-tuple $< j, R, \Sigma, \phi >$. As defined above, j is an integer identifier for each agent. The total reward of the agent is denoted by R, where $0 \leq R \leq T$ because the maximum reward in a single interaction is 1. The agent's memory (Σ) stores its experiences with different sellers.

The stereotyping part of the agent is through a learning model (ϕ) that tries to predict a seller's quality based on its feature. The agent continuously learns the relationship and fine-tunes the model by updating it after every interaction. We implement the learning model as a simple linear regression because the mapping function Θ is a linear combination of the traits.

When an agent a_j is assigned a subset of sellers $(s^t_{j1}, s^t_{j2}, \ldots s^t_{jk})$ at any timestep t, it has to pick the seller offering the highest quality. Essentially, the agent needs to compute the expected quality of each seller. This computation for one seller is outlined in Algorithm 2. If a_j has past experiences with a seller, it can retrieve past levels of quality from memory and average them (line 4). Otherwise, a_j has no reliable information to compute the expected quality. Instead, a_j uses stereotypes to fill this information gap by computing expected quality using ϕ (line 6).

After computing the expected qualities of all agents, the agent picks the seller with the highest value. Then, the interaction takes place, and the agent receives the reward as defined in Section 2.1. The agent also adds interaction details such as seller identifier, feature, and observed quality, to its memory.

Algorithm 2: Expected Quality function.

```
1 Function expQuality(a, s_k):
2     res ← search(a.Σ, s_k.i) ;
3     if res ≠ ∅ then
4         eq ← average(res);
5     else
6         eq ← predict(a.φ, s_k.f) ;
7     end
8     return eq;
9 end
```

[2]The details of this theorem are omitted due to space. The basic idea is to consider the number of sellers N_S to be of the form 2^X

We implement the agent's memory (Σ) as an array of episodic traces, with a forgetting rate λ. The trace is defined as a 4-tuple: $< i, f, q, \sigma >$, where i, f, and q denote the seller's identifier, feature, and quality, respectively. The last element σ represents the strength of the trace. All traces are created with an initial strength of σ_0, and with every passing timestep, the strength σ decays exponentially as: $\sigma_{t+1} = \sigma_t e^{-\lambda}$. When the strength of a trace drops below a global threshold σ_{min}, the trace is forgotten forever and deleted from the array. The retention period (d_Σ), i.e., the number of timesteps a trace remains in memory is given by: $d_\Sigma = \left\lceil \frac{1}{\lambda} \times \ln \frac{\sigma_0}{\sigma_{min}} \right\rceil$. An agent can search its memory by a seller identifier and retrieve all available matching traces. This is implemented by the `search` method in Algorithm 2.

3 AGENT TYPES

We can modify the basic agent described in Section 2.3 in several ways to generate different agent types. A core facet of agent behavior is estimating the quality of an unknown seller. The basic agent uses stereotypes to fill the information gap, but we explore some other approaches in this section. These agent types are summarized in Table 1. We borrow the dual-process model terminology of System-1 and System-2 to differentiate the agent types (Kahneman, 2011). Most stereotypes are automatic and at a subconscious level (Greenwald and Banaji, 1995), and therefore they belong to System-1 thinking. Looking up the agent's memory for past interactions and calculating the average is a cognitively demanding task, much like System-2. The Type0 and Type1 agents use System-2 first and resort to System-1 only when it fails, which is in line with the traditional thinking that emotions are irrational. However, it is becoming more apparent that humans do not think that way (Zhu and Thagard, 2002). Instead, the intuitive, mental shortcuts-based System-1 is the typical pathway, and System-2 is rarely invoked. We model Type2 and Type3 agents to follow this paradigm.

Type0 agents are the same as basic agents. They try to be *rational* first by looking up their own memory for past interactions with a particular seller. The memory may not have useful information, either because the seller is completely new or the agent has forgotten the interaction. In that case, the Type0 agent falls back on the stereotypes, which could be seen as irrational as they are gross generalizations.

Type1 agent extends the basic agent and is defined as a 5-tuple: $< j, R, \Sigma, \phi, eq_1 >$. Type1 agents are also rational because they do the heavy work of looking up

the memory to form an opinion about the seller. But, they do not use stereotypes when the desired information is unavailable. Instead, they model the quality of all unknown sellers as a constant eq_1, where $0 \leq eq_1 \leq 1$. We use this to model risk aversion in agents. *Risk aversion* is the tendency to prefer the option with low uncertainty (known seller) even if the riskier option may offer a larger expected reward (unknown seller) (Kahneman and Tversky, 1984). If eq_1 is closer to 1, the agent is more likely to choose a unknown seller. Such behavior can be seen as risk-seeking, and when eq_1 is on the lower end, the agent is risk-averse. The default value of eq_1 for Type1 is 0.5.

The *Type2* agent does not use its memory and completely relies on the mental shortcut of stereotypes. So, this is an implementation of pure System-1 thinking. Type2 agents do not differentiate between known and unknown sellers; instead, they simply use the stereotype to predict the expected quality. This approach is much faster in terms of computation because it avoids querying the memory.

Type3 agent extends the basic agent and is defined as a 5-tuple: $< j, R, \Sigma, \phi, p_3 >$. Type3 is a slight variation of Type2 in that it probabilistically uses System-2 thinking. System-1 thinking is still the primary mechanism, but with a probability of p_3, it scans the memory for past interactions and uses that information. We use a default value of p_3 as 0.1 in our experiments.

Table 1: Summary of Agent Types.

Agent Type	Description
Type0	Stereotypes used only when needed
Type1	No stereotypes, unknown sellers' quality assumed to be eq_1
Type2	Always use stereotypes
Type3	Probabilistically use stereotypes

4 EXPERIMENTS & RESULTS

We configure the system with 250 sellers ($N_S = 250$) and 10 agents ($N = 10$), and simulate it for 500 timesteps ($T = 500$). We implement the ABM in Python and use the stochastic gradient descent regressor (SGDRegressor) module in the scikit-learn library (Pedregosa et al., 2011) for the stereotype learning model of an agent. We executed all simulations on a laptop with an 8-core Intel® Core™ i5-10210U CPU and 16GB RAM, running Ubuntu 20.04.3.

We vary the agent types outlined in the previous section, some agent parameters like rate of forgetting and the feature-quality relationship (FQR) in sellers (described in Section II-B) to observe the effects on agents' performance. The metric to compare different simulations is the average total reward, i.e., we compute the total reward (after 500 timesteps) of each agent and average them. As a baseline, we consider an *oracle*, which is an agent that can see all of the seller's quality levels. We create two scenarios with almost all configurations — four traits impact the sellers' quality, and no trait impacts quality. We refer to these two scenarios as FQR-1 and FQR-0, respectively. We chose four traits as a representative scenario for a relationship between the seller's features and quality.

4.1 Baseline

First, we look at the scenario (FQR-1) where a feature-quality relationship exists, *i.e.*, a seller's quality depends on their feature (see Figure 1). The highest possible total reward for an agent is 500, but that is not attainable even for an all-seeing agent because none of the sellers in the assigned subset may be perfect sometimes. So, we see 458 for the oracle and 432 for Type0. Even with stereotyping, agents still lose some performance, probably because of poor interactions early when the stereotype has not formed yet. We will explore this in a later experiment. In the FQR-0 scenario (where there is no relationship between the feature and quality for sellers), Type0 agents' reward drops by about 12%. This result is expected because the sellers' qualities are fully noisy here, and the stereotypes cannot learn anything.

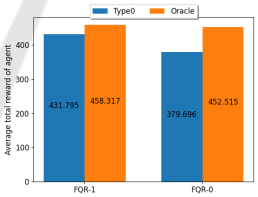
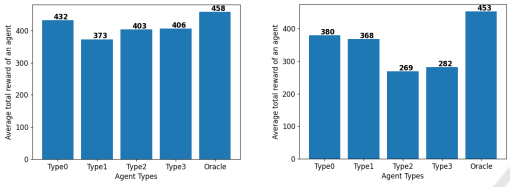


Figure 1: Performance of a Type0 agent in the presence (FQR-1) and absence (FQR-0) of the feature-quality relationship in sellers.

4.2 Comparing Agent Types

In this experiment, we run multiple simulations focusing on a different agent type in each run. These agent types and their behaviors are discussed in Section 3. We include the oracle for comparison, and we once again consider both scenarios. First, we look at the FQR-1 scenario (where sellers' qualities and features are related) in Figure 2. The Type0 agent performs the best, and the Type1 agent is the worst. The two other primarily-stereotyping approaches (Type2 and Type3) perform moderately well.

All agent types except Type1 perform significantly worse in FQR-0 (no relationship between a seller's quality and its feature), which is fairly obvious because there is no 'good' method to fill the missing information when there is a new seller. Type1 is not affected by the FQR simply because it does not rely on stereotypes. Whereas, Type2 and Type3 agents' performance crash because they rely heavily on stereotypes, and in FQR-0, stereotypes are typically meaningless. Type0 also takes a hit because it can not use stereotypes to fill the missing information. Therefore, its performance is close to Type1, which is not surprising as both essentially use random values to compute the expected quality of unknown sellers.



(a) FQR-1: Seller's feature and quality are related (b) FQR-0: Seller's feature and quality are not related

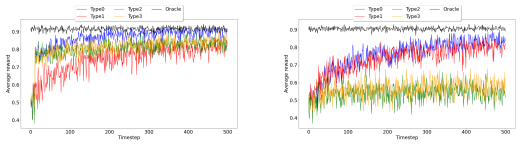
Figure 2: Comparing Agent Types.

4.3 Rewards over Time

We have been looking at the average total rewards in these experiments. But, to gain a better understanding of the agent's learning capabilities, we look at how the agents' rewards progress with time for each agent type (see Figure 3). The oracle's reward trajectory serves as the benchmark for other agent types to emulate.

In the scenario where a feature-quality relationship exists (FQR-1), the Type0 agent performs poorly early, and it quickly improves within 20-25 timesteps and gets gradually closer to the oracle's curve. It shows that, although we saw a significant difference in the average total reward between Type0 and oracle in the first experiment, most of that loss is in the early stages, and Type0 performs very similarly in the long run. The stereotype-focused approaches — Type2 and Type3 also follow a similar trajectory, with quick stereotype formation and sustained good rewards later. On the other hand, the memory-based Type1 has a much slower upward trend because it relies on building experience with sellers through interactions.

In the FQR-0 scenario, Type2 and Type3 agents that rely heavily on stereotypes obtain a reward of 0.5 or thereabouts throughout the simulation. Type1 follows a trajectory as in the FQR-1 scenario, slowly improving with more experience. Interestingly, the Type0 curve also looks similar to Type1 because the



(a) FQR-1: Seller's feature and quality are related (b) FQR-0: Seller's feature and quality are not related

Figure 3: Reward progression with time.

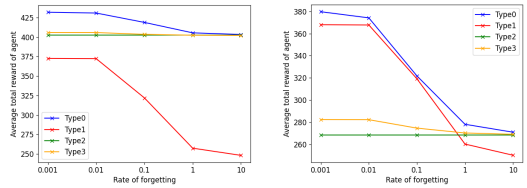
stereotypes are meaningless, and therefore Type0 essentially devolves to the memory-dependent Type1.

4.4 Other Results

In all of the experiments so far, we assumed perfect memory for the agents, i.e., $\lambda = 0$. We vary the **rate of forgetting** in this experiment and observe its effect on the performance of all four agent types. We set $\sigma_0 = 50$ and $\sigma_{min} = 1$, so the rate of forgetting is the only other tunable parameter in an agent's memory. We want to consider the two extremes of memory — perfect recall and immediate forgetting, and everything in between. Therefore, we consider values of λ from 0.001 (retention $\approx$ 3900 timesteps) to 10 (retention $<$ 1 timestep). These results are shown in Figure 4. Across both scenarios, Type2 is unaffected by the rate of forgetting because it does not use its memory at all. Type3 is also similar but occasionally depends on memory (with a probability of 0.1). So, there is a slight downward trend with increased forgetting.

Type0 and Type1 experience a sheer drop in rewards when the rate of forgetting rises above 0.01, i.e., retention goes below 390 timesteps. Experience with a particular seller is already scarce (average of 2 interactions with a specific seller in 500 timesteps), so forgetting those valuable interactions is not ideal for these agent types. One critical difference between the two is that Type0 always has a stereotype to fall back on. So, its reward is lower bounded by Type2 in a way. However, Type1 suffers more without its memory and essentially chooses an average seller every time (indicated by the average reward of $\approx$ 250, which is $500 * 1/2$).

Type1 agents calculated the expected quality of an unknown seller by simply assuming it to be some constant, eq_1, which is a parameter of the agent. If eq_1 is on the lower end towards zero, the agent avoids unknown sellers and picks from the known sellers. We vary eq_1 from 0 (**risk-averse**) to 1 (**risk-seeking**) in steps of 0.1 to see its effect on performance. The results (see Figure 5a) are unexpected because we see peak performance when the agent is moderately risk-seeking (eq_1 = 0.7). The quality among the seller



(a) FQR-1: Seller's feature and quality are related

(b) FQR-0: Seller's feature and quality are not related

Figure 4: Effect of rate of forgetting (λ).

population is symmetrically distributed with a mean of 0.5. So, one would expect moderate risk-seeking behavior ($eq_1 = 0.5$) would be optimal, but it is not so. We repeated the experiment with multiple random seeds (hence the multiple curves) to ensure it is a true effect, and it is indeed so.

We started with the assumption that sellers' features are fully observable, which is not realistic. Certain traits in the real world are either hidden or subject to misinterpretation. To allow for that, we introduce a bit-level error rate (*i.e.*, **noise**) in this experiment. When agents observe a seller's feature, each bit has a probability of being flipped. We simulate with an error rate of 0.125, and since there are 8 bits in the feature vector, one bit is flipped on average. These results are shown in Figure 5b. All stereotyping agents see a performance decline, but there is a steep drop for Type2 and Type3 agents (roughly 20%). On the other hand, Type0 suffers only an 8% drop and still performs better than all alternatives. It shows that Type0 agents are robust to noisy feature observations because they primarily use their memory, whereas purely stereotyping agents do not fare well in such realistic settings.

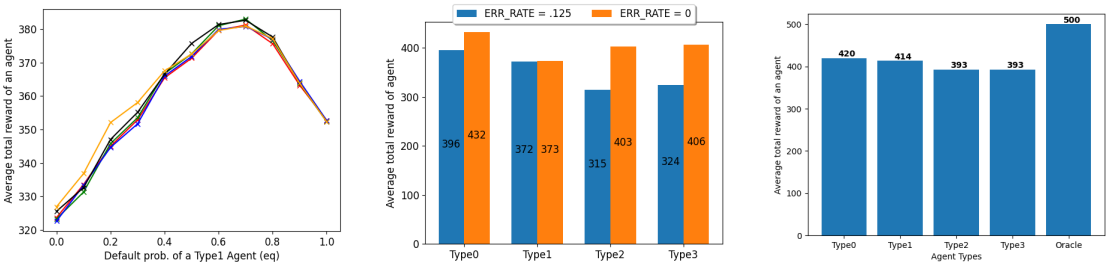
As a final experiment, we explored the use of a **real dataset**, that of Yelp[3], where users review businesses (sellers) and provide a star rating (out of 5). With this real-world Yelp dataset, quality is not constant (as assumed in our simulations); it is captured by the user's star rating, which we normalize to be in [0,1] for consistency. We use the seller's average star rating and the number of reviews from the dataset to model the seller's features. Empirical research (Jadil et al., 2022) suggests "online sellers' perceived size and perceived reputation" are important factors in users' trust and decision-making in e-commerce settings. We select 145 users from the dataset who had reviewed more than 500 businesses, and we run similar simulations with 500 rounds and users having to pick from 10 sellers in each round. The score for the agent in each round is the user's normalized rating of the chosen seller. The results are shown in Figure 5c, and we see a trend similar to what appeared in Figure 2: the Type 0 agent (that uses stereotypes carefully) performs the best compared to other agent types that rely more on stereotypes. This provides additional validation for our model and confirms the findings based on simulations.

[3]https://www.yelp.com/dataset

5 RELATED WORK

While the concept of stereotypes has been examined by user modeling researchers for a long time now (Kass and Finin, 1988), more relevant to our discussion are models for reasoning about trust and stereotypes, developed by multiagent system researchers. Included in that work is the research of Burnett *et al.* (Burnett et al., 2013) where simulations show that reasoning probabilistically together with the use of stereotypes will improve performance in trust modeling, especially where hidden feature-behaviour correlations exist. Taylor *et al.* (Taylor et al., 2017) discuss how witnesses can provide stereotype-reputation assessments to assist with subjective differences in trust (an issue that was introduced to the trust modeling community by Regan *et al.* (Regan et al., 2006)). They compare the result of reasoning with and without stereotypes, again leading to the conclusion that including stereotypes can improve accuracy. Liu *et al.* (Liu et al., 2013) also examine stereotypes in e-commerce settings and suggest using historical information to arrive at the trustworthiness. Sensoy *et al.* (Şensoy et al., 2016) present a more practical implementation: they elicit complex patterns exhibited by malicious agents and then use those as stereotypes to judge a new and unknown agent.

While many researchers seek to demonstrate how using stereotypes may be beneficial, we choose to examine more carefully the circumstances under which a more cautious approach to the use of stereotypes is advisable, helping to quantify as well where reckless use of stereotypes may introduce difficulties. We make these comparisons in different settings, where a feature-behavior relationship may or may not exist, and feature observations may be noisy. In this sense, we provide a broader landscape of the conditions under which stereotypical knowledge may be best introduced, for decision making in contexts like e-commerce where trust modeling is key. In our exploration we consider as well forgetting and decay, more explicitly, and we have initial insights into the role that the emotional makeup of a user may play, when deciding whether to rely on stereotypical inferences as part of the reasoning.



(a) Risk Aversion in Type1

(b) Effect of Noise in Feature Observations

(c) Yelp Dataset

Figure 5: Results for Risk Aversion, Noise, and Realworld Data.

6 CONCLUSION & FUTURE WORK

We present a model of decision-making based on stereotypes in an environment designed to simulate a marketplace. The agents are equipped with memories where they store past interactions. We design four types of agents with different approaches — (0) use stereotypes when there is no relevant information in memory, (1) guess at random, (2) use stereotypes always even if there are experiences in memory, and (3) use stereotypes mostly but use the memory with a small probability. The latter approaches always go for the subconscious mental shortcut first and invoke the cognitively demanding system rarely, if at all.

We compare all four agent types, and Type0 agents that use stereotypes cautiously perform the best. Our experiments also show that Type0 agents' performance is only about 5.8% less than the maximum possible reward. Considering most of the difference in reward occurs early when the stereotype is still forming, Type0 agents' performance is excellent in this setting. When the seller's features and quality are not related, the performance of all agent types drops because there is no efficient stereotype in this scenario. We also relax the fully observable feature assumption and introduce a bit-level error rate in feature observations. We experiment with an error rate of 0.125, and we see a minor drop in performance with Type0 agents. However, the Type2 and Type3 agents drop by over 20%. This result shows that in the real world, where some traits are hidden or can be easily misinterpreted, Type0 agents are better suited.

Varying the forgetfulness, we observe a downward trend in performance with an increasing rate of forgetting. This effect is less pronounced, if at all, in Type2 and Type3 agents because they hardly depend on memory. We also observe that Type1 agents that are risk-seeking perform better than risk-averse agents. Our detailed empirical study provides important insights on the use of stereotypes for agent decision making, advancing from what multiagent trust modeling has proposed.

While our model is grounded in reality, with implicit stereotypes and forgetting both confirmed by social science literature, we know that building a perfect model of human decision-making that captures all the intricacies is a nearly impossible endeavour. One direction for future work is to extend the proposed memory model. For example, it has been shown that events recalled often tend to stay longer in memory than those that are not (De Oliveira Alvares et al., 2013); this suggests considering not just recency of interactions with sellers but also frequency. It would also be valuable to explore in greater detail the impact of emotion on memory and decision making. The work of (Levine and Pizarro, 2004) indicates that events with higher emotional valence remain in the memory longer. Motivated by this research, prior agent-based approaches have looked at various effects of emotion and mood on decision-making (Sreenivas and Rao, 2020). We could extend such user models to better understand the effects of emotion and stereotyping on a user's performance in e-marketplaces.

For our initial study, we focused on cases where agents make decisions based on what is known in the multiagent trust modeling literature as direct experience (Tran, 2010; Falcone and Castelfranchi, 2001). This contrasts with other work (Burnett et al., 2013; Taylor et al., 2017) which explores scenarios where peer advice is also guiding the decision making (known in the trust modeling literature as advisor-based trust or reputation). For future work, we will expand our experiments to integrate reasoning about these witness reports as well. This would align us with a wealth of effort today in multiagent trust modeling that focuses on reports from advisors (Zhang and Cohen, 2008; Wang et al., 2010).

We anticipate conducting other extensions for our experiments. We are encouraged by efforts such as (Zhou et al., 2015). This work advocates for a deeper exploration of context, in order to reason about the conditions under which stereotypical knowledge is reliable. The paper also points out the need to address deliberate deception from agents and the value of adopting a more refined kind of regression analysis. For the future, we can also expand our exploration or Yelp or move on to consider other realworld datasets.

REFERENCES

Burnett, C., Norman, T. J., and Sycara, K. (2013). Stereotypical trust and bias in dynamic multiagent systems. *ACM Trans. Intell. Syst. Technol.*, 4(2):1–22.

De Oliveira Alvares, L., Crestani, A., Cassini, L., Haubrich, J., Santana, F., and Quillfeldt, J. (2013). Reactivation enables memory updating, precision-keeping and strengthening: Exploring the possible biological roles of reconsolidation. *Neuroscience*, 244:42–48.

Şensoy, M., Yilmaz, B., and Norman, T. J. (2016). Stage: Stereotypical Trust Assessment Through Graph Extraction. *Computational Intelligence*, 32(1):72–101.

Epstein, J. M. and Axtell, R. (1996). *Growing artificial societies: social science from the bottom up*. MIT Press.

Falcone, R. and Castelfranchi, C. (2001). Social trust: A cognitive approach. In *Trust and Deception in Virtual Societies*, pages 55–90. Springer Netherlands.

Fiske, S. T. (1992). Thinking is for doing: Portraits of social cognition from Daguerreotype to laserphoto. *Journal of Personality and Social Psychology*, 63(6):877–889.

Greenwald, A. G. and Banaji, M. R. (1995). Implicit social cognition: Attitudes, self-esteem, and stereotypes. *Psychological Review*, pages 4–27.

Hinton, D. P. R. (2013). *Stereotypes, Cognition and Culture*. Psychology Press, London.

Jadil, Y., Rana, N. P., and Dwivedi, Y. K. (2022). Understanding the drivers of online trust and intention to buy on a website: An emerging market perspective. *International Journal of Information Management Data Insights*, 2(1):100065.

Jussim, L., Crawford, J. T., and Rubinstein, R. S. (2015). Stereotype (In)Accuracy in Perceptions of Groups and Individuals:. *Current Directions in Psychological Science*, 24(6):490–497.

Kahneman, D. (2011). *Thinking, fast and slow*. Thinking, fast and slow. Farrar, Straus and Giroux, NY, US.

Kahneman, D. and Tversky, A. (1984). Choices, values, and frames. *American Psychologist*, 39(4):341–350.

Kass, R. and Finin, T. (1988). Modeling the user in natural language systems. *Computational Linguistics*, 14(3):5–22.

Levine, L. J. and Pizarro, D. A. (2004). Emotion and memory research: A grumpy overview. *Social Cognition*, 22(5):530–554.

Liu, X., Datta, A., and Rzadca, K. (2013). Trust beyond reputation: A computational trust model based on stereotypes. *Electron. Commer. Rec. Appl.*, 12(1):24–39.

Mullen, B. and Johnson, C. (1990). Distinctiveness-based illusory correlations and stereotyping: A meta-analytic integration*. *British Journal of Social Psychology*, 29(1):11–28.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Regan, K., Poupart, P., and Cohen, R. (2006). Bayesian reputation modeling in E-marketplaces sensitive to subjectivity, deception and change. In *Proceedings of AAAI 2016*, pages 1206–1212.

Sreenivas, N. K. and Rao, S. (2020). Analyzing the effects of memory biases and mood disorders on social performance. *Scientific Reports*, 10(1):20895.

Taylor, P., Griffiths, N., Barakat, L., and Miles, S. (2017). Stereotype Reputation with Limited Observability. In *Proceedings of the 19th International Workshop on Trust in Agent Societies (Trust@AAMAS 2017)*.

Tran, T. (2010). Protecting buying agents in e-marketplaces by direct experience trust modelling. *Knowledge and Information Systems*, 22(1):65–100.

Wang, Y., Zhang, J., and Vassileva, J. (2010). Effective web service selection via communities formed by super-agents. In *Proceedings of WI-IAT 2010*, pages 549–556.

Zell, E. and Lesick, T. L. (2021). Big five personality traits and performance: A quantitative synthesis of 50+ meta-analyses. *Journal of Personality*.

Zhang, J. and Cohen, R. (2008). Evaluating the trustworthiness of advice about seller agents in e-marketplaces: A personalized approach. *Electron. Commer. Rec. Appl.*, 7(3):330–340.

Zhou, P., Gu, X., Zhang, J., and Fei, M. (2015). A priori trust inference with context-aware stereotypical deep learning. *Knowledge-Based Systems*, 88:97–106.

Zhu, J. and Thagard, P. (2002). Emotion and action. *Philosophical Psychology*, 15(1):19–36.