Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Crowd management COVID-19

Liliana Durán-Polanco *, Mario Siller

Abstract

Crowds are a source of transmission in the COVID-19 spread. Contention and mitigation measures have focused on reducing people’s mass gathering. Such efforts have led to a drop in the economy. The application of a vaccine at a world level represents a grand challenge for humanity, and it is not likely to accomplish even within months. In the meantime, we still need tools to allow the people integration into their regular routines reducing the risk of infection. In this context, this paper presents a solution for crowd management. The aim is to monitor and manage crowd levels in interior places or point-of-interests (POI), particularly shopping centers or stores. The solution is based on a POI recommendation system that suggests the nearest safe options upon request of a particular POI to visit by the user. In this sense, it recommends places near the user location with the least estimated crowd. The recommendation algorithm uses a top-K approach and behavioral game theory to predict the user’s choice and estimate the crowd level for the requested POI. To evaluate the efficiency of this technological intervention in terms of the potential number of contacts of possible COVID-19 infections and the recommendation quality, we have developed an agent-based model (ABM). The adoption level of new technologies can be related to the end-user experience and trust in such technologies. As the end-user follows a recommendation that leads to uncrowded places, both the end-user experience and trust increased. We study and model this process using the OCEAN model of personality. The results from the studied scenarios showed that the proposed solution is widely adopted by the agents, as the trust factor increased from 0.5 (initial set value) to 0.76. In terms of crowd level, these are effectively managed and reduced on average by 40%. The mobility contacts were reduced by 40%, decreasing the risk of COVID-19 infection. An APP has been designed to support the described crowd management and contact tracing functionality. This APP is available on GitHub.

1. Introduction

In December 2019, in the city of Wuhan, was identified the SARS-CoV-2 virus for the first time. This virus produces COVID-19 disease. The virus spreads from person to person through droplets flying off from sick individuals (Chan et al., 2020). To avoid contagion and further consequences, the World Health Organization (WHO) took actions and published recommendations within a short time (WHO, 2020a), but despite those and the global state of alarm, on March 11, 2020, was declared a pandemic. As a prevention and containment measure, most countries have imposed a quarantine on the population. However, so far, the numbers from the WHO show a total of 216 countries, areas, or territories with active cases and more than 1 million deaths (WHO, 2020b). The massive outbreak of the disease has prompted the work of researchers and organizations to create vaccines, detection mechanisms, treatment strategies, containment, and mitigation measures, and decision-making tools (Chamola, Hassija, Gupta, & Guizani, 2020).

The spread of COVID-19 has caused a sharp drop in economic activity around the world; the impact of containment measures is not the same in all industries (Patterson, Raithatha, Schickling, Wieland, & Yeo, 2020). The level of impact is a function of how dependent an organization is to operate on face-to-face interactions. In sectors like retail, hospitality, and transportations is observed a high impact. As each country moves forward into the control of the pandemic, the containment measures have been relaxed in different ways by the health authorities. Even as restrictions on access to the previous sectors have eased, there will be those who may still avoid visiting these organizations. Currently, masks, antibacterial gel dispensers, disinfectant mats, and temperature controls, among others, have become part of our daily life; nevertheless, these measures do not exempt the danger of being in a crowd.

Control strategies for the evolution of the COVID-19 pandemic have been studied by Carli, Cavone, Epicoco, Scarabaggio, and Dotoli (2020) and Morato, Bastos, Caiudeiro, and Normey-Rico (2020), under...
the framework of Model Predictive Control, both studies investigated an optimal solution to control the pandemic using dynamic models. The first research work focuses on interventions in a multi-region scenario while the latter on social isolation measures. Both approaches emphasize regional control of the pandemic utilizing isolation measures such as telework, lockdown, or social distance. However, these models are not suitable for individual-level decision-making, which can help to study and prevent dangerous behaviors in society.

The early resilience observed in different places to some effects of the pandemic can be explained by the widespread access to digital technologies and the level of digital inclusion. Apps, for example, can be considered as an accessible technology and have been very successful and even institutionalized by governments. A large number of Apps related to the pandemic have been deployed, nevertheless, there are associated with ethical, social, and legal issues that still need to be addressed (Cattuto & Spina, 2020). Some of the COVID Apps include: Chatbots (Hispabot COVID-19, Victoria, etc.) (Agencia Digital de Innovación Pública, 2020; González, 2020), Self-assessment and remote diagnosis (MINECO, 2020a; MOHW, 2020; Secretaría de Salud MX, 2020; StarTimes, 2020), Contact tracing (Goggin, 2020; Health Canada, 2020; MINECO, 2020b; Ministry of Health, 2020), Geolocation (Government of Hong Kong (GovHK) and Office of the Government Chief Information Officer (OGCIO) and Hong Kong Special Administrative Region (HKSARG), 2020; MOHW, 2020), etc.

Recommendation systems provide suggestions that allow users with better decision-making. These systems are developed with different approaches and for varied application domains. In the current COVID-19 pandemic, knowledge-based recommendation systems have not been far explored, and they are particularly suitable in the context of personalized item selection. A knowledge-based recommendation system has the advantage of identifying the relationship between the user needs and the elements to be recommended. In contrast to collaborative systems, those based on knowledge do not require massive or inclusive private information from users. Consequently, they are considered non-invasive tools.

The WHO guidelines “Coronavirus disease (COVID-19) advice for the public” (WHO, 2020a), state that to reduce COVID-19 infection rates, the number of people per \( m^3 \) (cubic meter) must not exceed 1. The previous restriction directly affects the retail sector in terms of the number of customers that can be concurrently present in a given interior space. It also impacts the consumers as they need to plan and decide which place to visit for shopping or other activities. This consideration applies to a greater or lesser extent when visiting any interior space during the COVID-19 pandemic. In most cases, authorities rely both on the retailers to regulate the number of people (crowd management) in their places and the general public in deciding to visit only uncrowded or low-dense places. Under the current economic crisis, better crowd management regulations and implementations will allow a faster recovery of heavily affected face-to-face dependent industries and better management of the epidemic (lower infection rate, etc.). To address this hypothesis, we propose a knowledge-based recommendation system and an agent-based model. The recommendation system considers points-of-interest (POI) as recommendation items and suggests uncrowded safe places to visit. It is accessed through an App and intends to provide a non-invasive tool. Using a behavioral game theory approach allows to predict the user choice and estimate the crowd level at every POI. We study user behavior through an agent-based model considering individual aspects such as consciousness, agreeableness, openness, and trust. The model also accounts for retail attributes such as capacity and crowd density. The agent-based simulation has two objectives: be an evaluation tool for the recommender system and a means to measure the effectiveness of crowd management interventions at the community level during the COVID-19 pandemic.

The work divides into five sections. Section 2 reviews the state of the art of point of interest recommendation systems, as well as the apps developed in the context of the pandemic. Section 3 details the design of the proposal and the development of the agent-based model. Section 4 presents the results obtained from the simulation and analysis of the results. Section 5 contains a brief description of the App in development. Finally, Section 6 contains the conclusions.

2. State of the art

2.1. POI recommendation

A point-of-interest, according to (Papadopoulos, Popescu, & Kompaniatis, 2015), is defined as “an entity of interest with a well-defined location. Can range from famous landmarks (e.g. museums, churches, towers), natural attractions (e.g. bays, coasts, waterfalls) to commonplace spots (e.g. coffee shops, taverns)”. The subject of point of interest recommendation has been widely addressed in the literature. A large number of works focus on Location-Based Social Networks (LSBN). In Zhao, King, and Lyu (2016) three main challenges are identified for these types of systems: physical constraints (Liu & Xiong, 2013), complex relations between data (Baral & Li, 2017; Hoseini & Li, 2016), and heterogeneous information (Wang, Li, Liu, & Shao, 2021). Others focus on the context integration to improve the quality of recommendations as in Habayeb, Solanifar, Caglayan, and Bener (2016) and Macrì, Othman, Kobbane, Sabir, and Koubi (2016). While others address specific problems, for instance, in Xia, Li, Li, and Li (2017) neural networks are applied to generate a sequence of recommendations, or in Han and Yamana (2020) the authors seek to diversify the POI recommendations to include new places and those that are not often visited.

Usually, POI recommendation systems use hybrid, collaborative or top-K approaches (Chen, Yu, Tang, He, & Zeng, 2017). We are going to analyze six proposals. Each one contains an element relevant to ours.

2.1.1. Clustering approach and K-characteristics vector

The work presented in Mazumdar, Patra, and Babu (2020) consists of a recommendation system that uses a collaborative approach and is based on ratings. Cold-start is an issue in this approach since, in the beginning, there is not enough data to work with, to overcome this limitation, it crowdsources the ratings from different LSBN. The retrieved data requires a sentiment analysis by sentence to extract characteristics and ratings of every analyzed POI. Then a clustering algorithm is executed. At the end of this stage, similar POIs are grouped. A vector with K-characteristics is extracted from each cluster. Every user also has a characteristics vector. A similitude metric is then calculated using both vectors. The user is then associated with a cluster and the nearest POI within that cluster to the user location is obtained.

For our work, the relevant aspect is the use of a characteristics vector to represent every POI and the clustering approach. The crowdsourcing process can be seen as a way to obtain knowledge, in this sense, we do not have a collaborative recommender instead a knowledge-based recommender updated regularly to keep it up with the new user opinions. Our approach considers a knowledge-based recommendation system with a knowledge base updated through behavioral game theory predicting the user choice. A similarity metric for POI is not needed. In our work, we cluster the POIs according to their geographic distance, in this way, the clustering algorithm does not need to be executed regularly since the location is not likely to change.

2.1.2. Users’ interests

The model in Habayeb et al. (2016) associates the user interests (obtained from their interactions on the web) with information about places obtained through OSM (Open Street Map) and presents the user with recommendations based on his current location. The system has offline and online modules. The offline module is in charge of computationally demanding tasks such as obtaining POIs from OSM and classify them, as well as calculating the similarity between users and interests. The online module generates recommendations for places
according to the user distance and the similarity score between users with similar interests. The relevant aspects for our proposal are the assumed relationship between web interactions and interests, the offline calculation, and that it receives as input the current location of a user. In our work, we assume that the user interests are uncrowded and near places, for this reason, a similarity metric is not necessary. One of our objectives is to predict the user choice, which does not tackle this work.

2.1.3. Privacy preserving
User privacy is an utmost aspect in the design of systems. In Wang, Yang, and Lim (2018) we have a model that prioritizes it. In this model, the user and the central server do not exchange private data. In the beginning are characterized groups of users, what the server and the user exchange are group preferences. Knowledge is represented as a matrix of $m \times n$, where $m$ represents user groups and $n$ the POI. To establish the groups, it executes the k-means algorithm. At the user device, a preference vector is generated based on private data of previously visited places and ranks. An algorithm inside the device identifies the group to which the user belongs. From the server is retrieved a list with the best-scored POI within that group. The design based on maintaining the privacy of the users, the clustering technique, and the top-k approach represent the relevant aspects to our work. Even though this recommender system is not collaborative, it might suffer the cold-start issue because it needs existing data to characterize user groups. In our work, we have chosen to cluster POI according to a distance metric, and have assumed the user interest avoiding the cold-start issue.

2.1.4. Contextual information
POI recommendations in tourism have been improved by the use of Contextual information. In Braunhofer, Elahi, and Ricci (2014) we have a Context-Aware Recommender System (CARS). This kind of system provides adapted recommendations according to a contextual situation such as weather, season, or schedule. The design is addressed by a context-aware matrix factorization that pairs POI with contextual information and standardized parameters such as user rating. As with other recommender approaches, this has to deal with the cold-start issue for new users since they have not rated anything yet. To address this issue user ratings are predicted according to their personality. Using the Five-Item Personality Inventory in the registration process the system evaluates the personality and can predict user ratings to overcome the cold-start. In this system, the weather provides the context. During a pandemic, different factors can determine when it is appropriate to visit a POI. In the COVID-19 pandemic, we avoid crowded places. Our recommender system assumes the COVID-19 pandemic context and suggests uncrowded places. The prediction of user choice, in our work, is made through logit level-k avoiding the task, sometimes skipped, to answer a test for a user.

2.1.5. Crowdedness estimation
Following with CARS, the work in Mourchid et al. (2016) presents a recommender system that utilizes a Markov chain to predict contextual information and recommend places that can be visited at the next interval of time. In this case, contextual information refers to crowdedness. The Markov chain has three states: Not Crowded, Moderate Crowdedness, and Crowded. The recommendation process first infers user interest and estimates the crowdedness level from user check-in data. Then an algorithm called Learning-Based Random is applied to score places, the top-k best-scored places are given as recommendations. The crowd level estimation represents a relevant aspect for our work, but in the current pandemic context, the three states are not as representative as we need. The data is obtained from user check-in records which are not coherent with our privacy-preserving goal.

2.1.6. Utility theory
Utility theory can be useful to recommend POI due to the multivariate attributes. The work presented by Li, Xu, Chen, and Li (2015) is an example of this approach. The utility-based model proposed can learn aspects of the user’s preference to provide an improved recommendation. The recommendation system has offline and online modules. The offline module is in charge of learning aspects/attributes from reviews and ratings. With the learned aspects a utility function is calculated. The online module allows users to interact with the system providing the recommendation through a request. The retrieved POIs are the top-k with the highest utility. The applicable aspect in our work is the use of a utility function. Learning the relevant attributes for a POI is a demanding task. We do not need to learn these aspects to evaluate a POI because these are assumed due to the context of the COVID-19 pandemic. We as well use the request model to provide the recommendations. Finally, in our proposal, the execution of the logit level-k model helps to feed the attributes in our utility function, allowing us to supply better recommendations.

2.2. Apps developed during COVID-19 pandemic
A significant amount of apps have been developed during the COVID-19 pandemic, classifying them according to their purpose (Cascón-Katchadourian, 2020) we referenced a few examples:

- **Provide reliable information:** chatbots implemented by the WHO (WHO, 2020c, 2020d; WHO Regional Office for Europe, 2020) and by different countries such as Spain (González, 2020) and Mexico (Agencia Digital de Innovación Pública, 2020).

- **Self-assessment and remote diagnosis:** As a remote diagnosis example, “Self-quarantine & safety protection” (MOHW, 2020) from South Korea, allows practitioners to determine when to test and monitoring a quarantined patient. Self-assessment apps usually consist of a questionnaire that users carry out to determine if they present symptoms and provide recommendations. This type of application has been implemented in many countries (MINECO, 2020a; Secretaría de Salud MX, 2020; StarTimes, 2020).

- **Geolocation:** Apps made with this purpose handle sensitive data, for that reason, just a few examples can be found. Once that a patient has tested positive “Self-quarantine & safety protection” from South Korea (MOHW, 2020) allows to geolocate and monitor; Hong Kong’s “StayHomeSafe” (Government of Hong Kong (GovHK) and Office of the Government Chief Information Officer (OGCIO) and Hong Kong Special Administrative Region (HKSARG), 2020) works in conjunction with a bracelet, through geofencing it monitors those who have been quarantined.

- **Contact-tracing:** these type of applications have become popular despite privacy concerns. The first to emerge was “TraceTogether” in Singapore (Goggin, 2020), through Bluetooth it collects information from nearby devices. Following the same approach, other apps have emerged in different countries such as Australia (Goggin, 2020), Canada (Health Canada, 2020), and Spain (MINECO, 2020b). A different approach was taken by Israel with “HaMagen” (Ministry of Health, 2020), which uses geolocation. Check-in applications have emerged as well with this purpose as examples we can find “Territory Check-In” from Australia (Northern Territory of Australia, 2020), “NHS COVID-19” from UK (Department of Health and Social Care, 2020), and “NZ COVID Tracer” from New Zealand (Ministry of Health NZ, 2020). Still unfinished but with an ambitious goal is the PEPP-PT (Pan-European Privacy-Preserving Proximity Tracing) (PEPP–PT, 2020) project that seeks to track throughout the European Union while maintaining privacy.
3. Proposal

A crucial aspect in fighting the pandemic is about making better and informed decisions. A recommendation system allows people to make this kind of decision. The objective of our proposal is to monitor and manage crowd levels in interior places or point-of-interests. To address our goal, we have developed a model that integrates an agent-based simulation implemented in the Gama platform (Taillandier et al., 2018) and a recommendation system in Python, see “Fig. 1”. In this work, for practical purposes and taking into account people’s basic needs, we consider grocery stores as POIs. This represents an example of a COVID-19 affected industry.

The agents in the simulation model move around the city to make their purchases. Agents have an app that provides them with the least crowded places near them. The app is responsible for making requests to a server. The server was developed in Python and is responsible for calling the recommendation algorithm. The recommendation algorithm is knowledge-based since the product is known. In other words, grocery store locations are known and the user desired characteristics for such POIs: (1) closeness; and (2) no crowds. Privacy is preserved given that the system does not monitor users to estimate crowds. Rather, it uses behavioral game theory to assess the level of the crowd at each POI.

3.1. Agent-based model

The proposed ABM aims to evaluate the quality of recommendations and the intervention of the recommendation system as a decision-making tool under the COVID-19 pandemic. The model depicts the adoption process of the recommendation system in the form of an App. ABM has been used to study the adoption process of new technology in fields such as smart energy (Christensen, Ma, Varbak, Demazeau, & Jørgensen, 2020), health systems (Pardo & Coronado, 2016), business model (Basingab et al., 2017), and telecommunications (Jahng & Park, 2020). The adoption process in this work is based on the dynamic trust model of (Choi & Nazareth, 2014). Users develop trust in the App through experience. Agents gain experience as they follow the App recommendation. Trust is calculated following the indirect experience model from Jaffry and Treur (2013). The proposal is based on the following hypothesis: major trust implies that users will make informed decisions, which results in better crowd management, decreasing in this way the risk contacts. This hypothesis is represented in the causal network of “Fig. 2”. The proposed ABM is described next using the ODD protocol (Grimm et al., 2010).

3.1.1. Entities, state variables, and scales

The main entities used for this model are presented in this Section. The entities are divided into two groups: environmental and species.

Environmental entities represent the infrastructure of our model and these include:

- Street: A graph on which agents can travel during the simulation. It only contains two variables shape and location. A GIS file sets the variables.
- Commercial block: Polygons that represent parts of the city dedicated only for shopping purposes. It only contains two variables shape and location. A GIS file sets the variables.
- Residential block: Polygons that represent parts of the city where people live or work. It only contains two variables shape and location. A GIS file sets the variables.
- Connection manager: Agent who manages the responses of the recommendation system. It assigns every response to the respective agent. The attributes of this agent are related with the network connection: server (“localhost”), protocol (“udp_server”), port (0-65535).
- Store: Place where people can do shopping. It contains six attributes: location (coordinates), store (name of store), capacity (total number of people that can fit inside the place), people_allowed (allowed number of people inside the place), current_people (current number of people inside the place), crowd_percentage (percentage of the current number of people in relation to the total capacity).

Species entities represent agents that take decisions and interact with each other, these include:

- People: Parent species that describe individuals. It contains general attributes, personality, desires, intentions, and beliefs. Attributes: home (It represents a house or work location, is set randomly at any point in a residential block), need_supplies_time

![Fig. 1. System’s architecture.](image_url)

![Fig. 2. Hypothesis causal model.](image_url)
(A random number of cycles that decreases to trigger the need to go to a store), shopping_time (A random number of cycles that represents the spending time in a store, the number is generated using a Gaussian distribution with 48 min as mean and 10 as standard deviation), goal_place (Location to which an agent is heading), knowledge_base (List of the two nearest stores to the home location), wait_time_for_response (Number of cycles to wait for a response from the App, currently is 2), speed (Speed with which the agent moves, currently is 5 km/h, walking average speed), belief_congestion (Percentage referring to space capacity which agents/people consider as crowded, this attribute is set randomly, and it is different for every individual).

**Personality:** According to the OCEAN or Five Factor model (Wiggins, 1996), openness (departs the imaginative and creative), conscientiousness (related to organization and carefulness), extraversion (describes how sociable and unwind a person is), agreeableness (describes how well a person gets along with others), neurosisim (refers to the proneness to negative emotions). Each factor is bipolar (Goldberg, 1990) and address the agent personality in Agent-Based simulations.

**Beliefs:** uncongested (represents the belief that a place is not crowded), congested (represents the belief that a place is crowded), end_shopping (represents that an agent has found everything he needed).

**Desires:** need_supplies (represent the need to do the shopping).

**Intentions:** stay_home (Agent stays at home position), walking (Agent walks through the map), go_shopping (Agent walks to goal_place), go_home (Agent walks to home), find_near_store (Different for every child species of People) and it consists on selecting a place to do the shopping), shopping (Agent remains inside a store shopping), wait_response (Agent waits for App response). Emotion: fear (Emotion fired when an agent perceives a multitude in the store he is heading to, an agent can perceive a multitude in a 50 m radius).

- Person: A species that inherits from People and represents an agent that has no access to the recommendation system.
- User: A species that inherits from People and represents an agent that has access to the recommendation system through the App. This species contains particular attributes.
  - Attributes: app_trust (Indicates the level of trust in the App, it dynamically changes following the Dynamic Security-Trust framework as in Choi and Nazareth (2014) and Ngo-Ye, Choi, and Cummings (2018) and the indirect experience dynamic trust model from Jaffry and Treur (2013)), last_choice (Indicates the source of the last decision made by the user: App recommendation or User knowledge).
- App: Micro species of User which is in charge to request and show the response from the server to the User.
  - Attributes: recommended_places (Contains the list of places recommended by the recommendation system), coordinates (The location where a User is requesting a recommendation), server (IP address of the server to which it will make the recommendation request), protocol (Connection protocol to the server, “tcp_client”), port (Connection port, 0-65535).

### 3.1.2. Process overview

All People have an attribute called home, which represents their house or work location. Agents might start with one of two intentions walking or stay_home and remain there until the attribute need_supplies_time decreases to 0, which denotes that the agent has developed the desire need_supplies. The desire need_supplies triggers the intention find_near_store. This an individual and particular process for every child species of People. The intention find_near_store consists of selecting a place to do the shopping and setting the attribute goal_place. For the child species User, this selection depends on the attribute app_trust and fear emotion, see “Alg. 1”. This is referred to as User decision and indicates that a User-agent can choose between following the App recommendation or its knowledge. For the child species Person, the decision depends only on fear emotion because a Person-agent does not have access to the App and relies only on its knowledge, see “Alg. 2”. People’s knowledge is represented by their capacity to estimate the distance to different stores. The knowledge_base contains the two nearest stores to the home location, but if agents are far from there (which happens when fear emotion is triggered), they will choose a store based on a distance estimation to their current location. The select_nearby_store function finds the nearest store to the current location of the People agent, this function is called when the fear emotion has been triggered and allows the agent to find a new store to do the shopping when does not want to follow the recommendation or when does not have access to the App. Once goal_place is set the current intention changes to go_shopping, and the agent walks to goal_place. The fear emotion is activated depending on the perception of a multitude. When an agent is at a 50 meters distance away from its goal_place it can perceive a multitude. The attribute belief_congestion indicates the percentage in which an agent perceives a place as crowded. An agent queries the crowd_percentage attribute at a store, if crowd_percentage is greater than belief_congestion then the store is considered as crowded. The activation event of the fear emotion changes the current intention to find_near_store. If the agent does not perceive a multitude, then changes the intention to shopping, and it remains within the same intention until the attribute shopping_time decreases to 0. When shopping_time is 0, randomly, the agent selects between to find_near_store (with a 1% probability according to statistics from Renner, Baker, Cook, and Mellinger (2020)) or go_home. When the agent changes its intention to go_home, it then walks back to the home location and remains there until the simulation ends. The simulation ends when all agents are back home after they do their shopping. (see “Fig. 3”).

#### Algorithm 1 User decision

1: if flip(app_trust) then # Select app option
  2: selected ← one_of(app.recommended_places)
3: else
  4: if has_emotion(fear) then
    5: selected ← one_of(knowledge_base)
  6: else
    7: selected ← select_nearby_store
  8: end if
9: end if
10: goal_place ← selected

#### Algorithm 2 Person decision

1: if has_emotion(fear) then
  2: selected ← one_of(knowledge_base)
3: else
  4: selected ← select_nearby_store
5: end if
6: goal_place ← selected

### 3.1.3. Initialization

The Initialization of the model takes into account the following considerations: (i) the infrastructure entities such as the streets, residential and commercial blocks are created from GIS files (shapefiles); (ii) the store entities are loaded from a CSV file that provides location coordinates, store name, and capacity; (iii) a simulation parameter sets the populations of human agents; and (iv) the allowed crowd percentage is specified using a simulation parameter that can be changed at execution time.
3.1.4. Submodels
The influence diagram “Fig. 4” describes the agent’s decision-making process. The Choose Store box represents the decision that agents must make. Agents have two options that will lead them to a store: the App or their Knowledge. They select one according to their trust level in the App. The trust dynamic follows indirect experience model from Jaffry and Treur (2013). This model fuses one’s own experience and the influence of the experiences of others. Direct experience represents an individual evaluation of the recommendations of the App, the agent can evaluate a recommendation after having followed it. A good recommendation will lead the agent to a place perceived as uncrowded (the store has a crowd_percentage lower than the agent’s attribute belief_congestion). The overall experience is calculated by:}

\[
E_A(t) = \frac{\sum_{B=1, B\neq A}^N T_B(t)}{N - 1}
\]

where \( N \) represents the total number of agents, in this case of the User type, \( T_B \) the current trust of each User agent. In our model, the direct experience \( E^d \) for each agent is calculated by the number of good recommendation with respect to the total amount of followed recommendations:

\[
E^d_A(t) = \frac{\#\text{Good Recommendation}}{\#\text{Total Recommendations}}
\]

A recommendation is considered a Good Recommendation when it leads the agent to a location with a crowd_percentage lower than the belief_congestion attribute. The overall experience \( E \) to calculate the trust is computed by:

\[
E_A(t) = a_A \times E^d_A(t) + (1 - a_A) \times E^i_A(t)
\]

where \( a_A \) represents the willingness of the agent to be influenced by the experience of others. Bayel et al. (2020) suggests that behavior is influenced by what we think others approve. Our model considers indirect experience \( E^i_A \) as a measure of approval. According to Bègue et al. (2014) obedience is a way of social influence. We consider...
obedience as $a_k$ in the trust model. $E_k^j$ states the direct experience of the agent. Finally, the trust is computed by:

$$T_A(t + \Delta t) = T_A(t) + \gamma_A \times (E_A(t) - T_A(t)) \times \Delta t$$  

(4)

where $T_A$ represents the current trust of the agent, $E_A$ the overall experience, and $\gamma$ a personal characteristic of flexibility. The factor $\gamma$ in this model is associated with the openness aspect of personality since it is linked to our proneness to accept new experiences (Wiggins, 1996). The App gets recommendations based on the distance, and the estimated crowd level, a detailed explanation of this process is provided in the next section.

### 3.2. Recommendation algorithm

The authors in Han and Yamana (2020) state that a POI recommendation problem is equivalent to a k-POI selection problem. In this sense, a recommender system selects the first k POIs that match user preferences from a set of candidate POIs. Using this approach, we have developed a knowledge-based recommendation system.

We define the general characteristics based on the classification framework proposed by Bouraga, Jureta, Faulkner, and Hersens (2014) for knowledge-based recommendation systems, see “Table 1”.

The system takes the POI database and executes k-means as a clustering algorithm. POI clusters and mean points are saved in the database. The recommendation algorithm takes the mean points to classify the location of a user to a cluster. For every POI within a group, it obtains the coordinates, capacity, allowed percentage, and currently obtained the coordinates, capacity, allowed percentage, and currently.

#### Recommendation algorithm

For every POI within a group, the system calculates a payoff function $S$ based on the distance to the user location $d_u$ and the estimated multitude percentage $m_{per}$. The distance $d_u$ and estimated multitude percentage $m_{per}$ are normalized. A multitude_threshold must be set. The threshold represents the multitude percentage in which we want the POI multitude remains as much as possible. The payoff computing penalizes POI with a greater estimated multitude than the multitude_threshold. The penalization works as a bonus $b$ given to those POI with a minor estimated multitude than the multitude_threshold. The payoff $S$ is computed using three weights: distance weighing $W_d$, multitude weighing $W_m$, and bonus weighing $W_b$. The best POI to recommend is the one with the biggest payoff score.

$$b_j = \begin{cases} 
1 & m_{per_j} < \text{multitude_threshold} \\
0 & \text{otherwise} 
\end{cases}$$  

(5)

$$d_{u, \text{normalized}} = \frac{d_u}{d_{max}}$$  

(6)

$$m_{\text{normalized}_{i}} = \frac{m_{per_{i}}}{m_{per_{max}}}$$  

(7)

$$S = W_d \times d_{u, \text{normalized}} + W_m \times m_{\text{normalized}_{i}} + W_b \times b_j$$  

(8)

Once the payoff is calculated, and the k-best POI selected, logit level-k calculates the probability of a user to choose each place, updating the estimated multitude at the POI with the higher probability. The logit level-k model is computed using the following equation (Kochenderfer, 2015):

$$P(a_i) \propto e^{\lambda (b_i (s_{<1} - a_i) - 5)}$$  

(9)

where $s_{<1}$ denotes the assumed strategy profile of other agents, $k$ represents the rationality depth parameter, $\lambda$ the precision parameter that controls the liability to utility differences, $a_i$ represents the selected recommendation and $U_i$ references to a utility function. We consider a depth $k = 1$ because it has proven to be a representative level for human agents (Breitmoser, Tan, & Zizzo, 2014) and has been applied in similar situations (Jones, 2012). $a_i$ represents a utility value given when selecting a recommendation and it is defined as follows:

$$a_i = \begin{cases} 
S_i + 5 & a_i \text{ is not crowded} \\
S_i - 5 & a_i \text{ is crowded} 
\end{cases}$$  

(10)

where $S_i$ represents the payoff associated to the selected recommendation, $s_{<1} \text{ represents the assumed strategy profile of other agents. The utility function takes into account a ranking the benefits of the action: select a crowded POI or select an uncrowded POI. The } +5 \text{ and } -5 \text{ penalizations are the benefits obtained from the action. We choose such values to differentiate the results in the logit level-k computing. However, if these values are changed while the relationship is maintained, there will be no difference in the calculation effect. Note that in logit level-k the agents assume that other agents have a } k - 1 \text{ level. In this model:}

$$s_{<1} = a_{<i}$$  

(11)

which represents the choice of the opponent’s assuming that his level is 0, which means that it is chosen randomly. Finally the utility is
computed by:

\[ U_i(a_i, s_{-i}) = a_i + s_{-i} \]  \hspace{1cm} (12)

The recommendation algorithm is shown in “Fig. 5”.

4. Experimental work

4.1. Assumptions

- The simulation runs in one day, for example, Saturday, which is the preferred day for shopping.
- The population of agents represents the people who make purchases during a day.
- The stores do not have a monitoring mechanism at the entrance because we are exploring how the tool by itself can prevent multitudes.
- Trust evolves following the indirect experience model from Jaffry and Treur (2013). Agents gain experience by using app recommendations.

- The quality of recommendations is the factor by which the trust is increased. A good recommendation is one that leads to a
place perceived as uncrowded, otherwise it is considered a bad recommendation (place perceived as crowded).

- Personality plays a role in the decision-making process and trust evolution. Openness is linked to our readiness to have new experiences (equivalent to $\gamma$ in the trust model). Agreeableness and Conscientiousness are linked to obedience (Bègue et al., 2014), we considered obedience as how others influence us (equivalent to $\alpha$ in the trust model).

- Trust is a measure within the interval $[0,1]$. Initial trust is 0.5, this comes from the fact that we assumed that our App is institutionalized, which gives more credibility to the tool.

- A track of mobility contacts is made as agents move through the map. A mobility contact is made when two agents violate social distance guideline of 1.5 m.

4.2. Setting

- Our experiments were executed using a zone from Guadalajara city in Mexico, the area around the Galerías Mall. The selected zone is highly commercial and surrounded by different residential areas (see "Fig. 14").

- The selected stores are the biggest in the zone (Galerías, Walmart, SAMS, Costco, Comercial, and Chedraui).

- A population of 1000 agents was set for exploration purposes.

- A 25% of the total capacity is allowed at each store. However, the stores do not have a monitoring mechanism at the entrance because we are exploring how the tool by itself can prevent multitudes, as a result the multitude percentage at the simulation can exceed this quantity. This percentage sets the parameter $\text{multitude\_threshold} = 0.25$ in the payoff calculation.

- The parameters for payoff computing are: $W_d = 0.2$, $W_a = 0.6$, $W_b = 0.2$. 
4.3. Experimental scenarios and results

1. **Scenario 1**: 100% of the population do not have access to the recommendation system. The simulation lasted 12 h with 37 min (758 min), in which a store reached more than 100% of its capacity. See "Fig. 6".

2. **Scenario 2**: 50% of the population do not have access to the recommendation system, and the other 50% do have it. The simulation lasted 12 h and 16 min (737 min), in which some stores almost reached 60% of their capacity (see "Fig. 7"). The percentage of good recommendations is high, 99.1% (see "Fig. 8"). The trust evolution in the application allowed users to opt for the suggestions given. At first, the decisions (following the App recommendation or the own knowledge) were even but as the simulation unfolds, the App recommendations were preferred (see "Fig. 9"). The average trust in the application increased from 0.5 to 0.77.

3. **Scenario 3**: 100% of the population have access to the Recommendation system. The simulation lasted 12 h and 33 min (754 min), in which some stores reached more than 50% of their capacity (see "Fig. 10"). The percentage of good recommendations is high, 90.7% (see "Fig. 11"). At the end of the simulation, the recommendations were preferred (see "Fig. 12"). The average trust in the application increased from 0.5 to 0.76.

4.4. Analysis and discussions

The observed track of mobility contacts changed in every scenario. We can appreciate that scenario 1 reached 42,322 contacts, while the others presented an improvement reducing the contacts to 34,484 and 25,124, respectively (see "Fig. 13"). For a summary of the presented peak values and the number of times that the allowed capacity was passed, see "Table 2"). It is possible to appreciate how in every scenario the peak distribution is more balanced allowing lower peak values. The balanced distribution affects the number of times that the allowed capacity was surpassed, this value increases as more people have access to the App. The results suggest that the App intervention distributes the users reducing the crowds level in all the available POIs. For the intervention to succeed, users should follow the App recommendations. In the simulations, they developed an average trust of 0.76, and by the end users prefer to follow the Apps recommendations. The previous fact suggests that, with time, most of the users will follow the suggestions. The distribution of the crowd affected the mobility contacts, it is observed that a more balanced distribution led to fewer contacts. The results also show that some stores remain more crowded than others, this can be due to the geographic location.
5. Crowd management and contact tracing App. Design and implementation

“Crowd Management and Contact Tracing App” was conceived as a comprehensive mitigation tool. It aims to serve as a crowd manager in POIs, a monitoring and also contact-tracing tool. This objective must be met under the premise of preserving privacy and not posing a threat due to its pervasiveness. To address this, we started from the fact that, currently, crowd management measures commonly include control strategies such as temperature measurement, the use of disinfectant mats and hands sanitizer. Every public or private space that allows the gathering of people must carry out these preventive measures, and from now on, we refer to such places as establishments. Our App works in conjunction with this protocol. It does not monitor the location of a user, but keeps records of user check-ins at the establishments. These check-ins are implemented using QR codes. We have considered two types of users: Establishment and Common User. Each user has access to different modules. The establishment module allows owners and employees to keep track of the crowd level. The Common User module integrates contact-tracing and a decision-support engine to provide users a way to reincorporate to normal activities. Here we list the main features for a Common User:

- Anonymous registration. To use the App, the user register anonymously. A hash code is generated at the beginning and no more personal data is required.
- Record of temperature measures. Temperature measurement is a control strategy in the current pandemic. The App can keep a track of all the temperature measures made at every visited place.
- Notify infection. When a user is infected he chooses to share the history of all visited places, then the App sends an alarm to all the users that could had been in contact with the infected user.
- Heatmap to show crowded places. Through the heatmap, the user can visualize which places are more convenient to visit.
- Recommend convenient places. Besides the heatmap, the App can suggest near and uncrowded places.

Key features for establishment:

- Crowd level monitoring. Keep track of all the check-in/out to the establishment, allowing continuous monitoring to not incur in a fault to mitigation policies imposed by health authorities.
- Check-in. The user registration process returns a QR code. The code is unique and should be scanned at every establishment to register the check-in and maintain the temperature records.
- Notification when allowed capacity is reached.

The App code is available on GitHub. New functionality is being currently added and access can be granted upon request to the authors. Some screenshots are presented in figures: “Figs. 15”, “16” and “17”.

6. Conclusions

In this work, we have presented a technological intervention in the context of the COVID-19 pandemic. Results show that such an intervention can be useful even when not all the population has access to the proposed recommendation system. It is observed that in all cases trust is increased and good recommendations never descended from 90%. The crowd management in interiors was not as successful as we thought, however, we can observe improvements in the peak levels reached in every scenario. For instance, scenarios with the intervention
never reach 60% of capacity. In every simulation we can observe that some stores remained more crowded than others, this can be due to the geographic location. In terms of mobility contacts, it varies in every scenario. It can be seen that the higher the access to the recommendation system, the lower the number of contacts produced. The proposed model can be complemented with SIR, SIS, or SEIR models to further study the spread of the coronavirus during a contact situation. Finally, it is possible to estimate and manage the crowd level at POIs using little information and maintaining user privacy.

**Declaration of competing interest**

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

**Acknowledgments**

We thank the development team integrated by CINVESTAV Guadalajara MSc Students from the Software Engineering course: Abel David García Nevarez, Carlos Cárdenas Ruiz, David Alejandro Mares Rodríguez, David Alejandro Mares Rodríguez, Juan Bernardo Camacho Tirado, Julia Alejandra Rodríguez Abud and Marisol Vera Arellano for their work with the Crowd Management and Contact-Tracing App. Also we thank to David Ramirez and Luis Villela for their role as stakeholders.

This work is supported by Consejo Nacional de Ciencia y Tecnología (CONACYT) via the Master scholarship program, Mexico (CVU-1001334).
Towards real-time demand-aware sequential POI recommendation. *Information Sciences*, 547, 482-497. http://dx.doi.org/10.1016/j.ins.2020.08.088.

Wang, X., Yang, H., & Lim, K. (2018). Privacy-preserving POI recommendation using nonnegative matrix factorization. In *2018 IEEE symposium on privacy-aware computing (PAC)*. IEEE, http://dx.doi.org/10.1109/pac.2018.00018.

WHO (2020a). Advice for the public. URL https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public.

WHO (2020b). WHO Coronavirus disease (COVID-19) dashboard | WHO coronavirus disease (COVID-19) dashboard. URL https://covid19.who.int.

WHO (2020c). WHO launches a chatbot on Facebook Messenger to combat COVID-19 misinformation. URL https://www.who.int/news-room/feature-stories/detail/who-launches-a-chatbot-powered-facebook-messenger-to-combat-covid-19-misinformation.

WHO (2020d). WHO Health Alert brings COVID-19 facts to billions via WhatsApp. URL https://www.who.int/news-room/feature-stories/detail/who-health-alert-brings-covid-19-facts-to-billions-via-whatsapp.

WHO Regional Office for Europe HealthBuddy: a new chatbot to engage with communities in Europe and central Asia on COVID-19. URL https://www.euro.who.int/en/health-topics/health-emergencies/coronavirus-covid-19/news/news/2020/5/healthbuddy-a-new-chatbot-to-engage-with-communities-in-europe-and-central-asia-on-covid-19.

Wiggins, J. S. (1996). The five-factor model of personality.

Xin, B., Li, Y., Li, Q., & Li, T. (2017). Attention-based recurrent neural network for location recommendation. In *2017 12th international conference on intelligent systems and knowledge engineering (ISKE)*. IEEE, http://dx.doi.org/10.1109/iske.2017.8258747.

Zhao, S., King, I., & Lyu, M. R. (2016). A survey of point-of-interest recommendation in location-based social networks. CoRR, abs/1607.00647, arXiv:1607.00647 URL http://arxiv.org/abs/1607.00647.