

Predicting Customer Goals in Financial Institution Services: A Data-Driven LSTM Approach

Andrew Estornell*,¹ Stylianos Loukas Vasileiou*,¹ William Yeoh,¹ Daniel Borrajo,² Rui Silva²

¹ Washington University in St. Louis
{aestornell,vstylianos, wyeoh}@wustl.edu

² J.P. Morgan AI Research
{daniel.borrajo,rui.silva}@jpmchase.com

Abstract

In today’s competitive financial landscape, understanding and anticipating customer goals is crucial for institutions to deliver a personalized and optimized user experience. This has given rise to the problem of accurately predicting customer goals and actions. Focusing on that problem, we use historical customer traces generated by a realistic simulator and present two simple models for predicting customer goals and future actions – an LSTM model and an LSTM model enhanced with state-space graph embeddings. Our results demonstrate the effectiveness of these models when it comes to predicting customer goals and actions.

Introduction

The financial industry has experienced significant transformation in recent years, driven by rapid technological advancements, evolving customer expectations, and increased competition. As customers demand more personalized and convenient services, financial institutions are under pressure to develop a deeper understanding of their clients’ needs and preferences. This has led to a growing interest in leveraging data-driven approaches to gain insights into customer behavior and predict future actions. Accurate goal prediction can help financial institutions provide targeted incentives, improve customer satisfaction, and ultimately foster loyalty and retention in an increasingly competitive landscape.

Planning-based approaches have been widely used for goal prediction, as they focus on modeling an agent’s decision-making process and finding optimal sequences of actions to achieve specific objectives (Ramírez and Geffner 2010; Sohrabi, Riabov, and Udrea 2016; Keren, Gal, and Karpas 2019; Wayllace et al. 2016; Vered and Kaminka 2017). For example, Ramírez and Geffner (2010) propose a probabilistic plan recognition approach that utilizes classical planners to predict goals based on observed actions. Keren, Gal, and Karpas (2019) introduced the concept of goal recognition design, which aims to optimize the planning domain to facilitate the goal prediction process.

In this context, the work of Borrajo and Veloso (2020) presents an approach to address the challenges associated with predicting goals in complex real-world domains. In particular, they introduced a domain-independent simulator for

generating synthetic customer behavior datasets, which can be used to model and analyze customer interactions with a financial institution, such as ATM or mobile app usage. By leveraging this simulator, they are able to create datasets that accurately represent the intricacies and dynamics of customer traces in a banking environment, providing a valuable foundation for the development and evaluation of goal prediction models.

Inspired by the work of Borrajo and Veloso (2020), we build upon their domain-independent simulator to generate synthetic customer behavior datasets. We then use this data to train two models – an LSTM model and an LSTM model enhanced with state-space graph embeddings. The LSTM-based models capitalize on the sequential nature of customer traces, capturing the intricate patterns present in customer interactions over time. By incorporating state-space graph embeddings into the LSTM model, we further enrich the model’s understanding of the relationships and dependencies among various features within the dataset, which may lead to improved performance. This combination of LSTM models and state graph embeddings offers a more scalable and efficient solution in predicting customer goals and actions, while maintaining a high level of accuracy and robustness in the face of real-world complexities. Our results demonstrate the effectiveness of LSTM models and state graph embeddings in addressing the challenges faced by financial institutions when it comes to predicting customer goals and actions.

Related Work

Goal prediction (known as goal recognition in the planning literature) refers to the process of identifying an agent’s objective among several possibilities, based on the agent’s behavior, a model of the environment, and a sequence of observations. The first approach that explicitly addresses goal prediction is that of Baker, Saxe, and Tenenbaum (2009), who framed the problem as “inverse” planning. Ramírez and Geffner (2010) followed up on the work by proposing a generative approach that uses classical planning algorithms for inferring probability distributions over a set of possible goals given some observations. Recent research has extended the original problem formulation of recognizing plans in various domain models, such as continuous domain models and epistemic planning problems (Vered and Kaminka 2017;

*Equal contribution.

Sohrabi, Riabov, and Udrea 2016; Kaminka, Vered, and Agmon 2018; Shvo et al. 2020). In similar spirit, Keren, Gal, and Karpas (2019) provided an alternate view of goal recognition that focuses on modifying the domain model such that the goal recognition can be achieved with as few observations as possible. Extensions of the goal recognition design problem have been developed over various types of domain models and settings (Shvo and McIlraith 2020; Wayllace et al. 2016, 2020; Wayllace and Yeoh 2022).

However, planning-based approaches often struggle to scale with the size and complexity of real-world domains, due to factors such as large state spaces, partial observability, and dynamic environments. In contrast, learning-based approaches may perform better under these circumstances, especially when real-time or near-real-time predictions are required. Most notably, Borrajo, Gopalakrishnan, and Potluru (2020) illustrated the trade-offs between planning-based and learning-based approaches for goal recognition in different (planning) domains. Their results highlight that planning-based approaches perform better when there is a partial order of actions in plans, whereas learning techniques, such as LSTMs, can better capture the relationship between actions and goals if there is a relation not directly associated with goal achievement. While in this paper we also use a learning-based approach (e.g., LSTMs) for goal prediction, note that we focus on the specific task of predicting customer goals and actions in the context of financial institutions. Finally, there has been an interest in integrating planning-based approach with data-driven approaches. For example, Wilken et al. (2023) proposed a hybrid method that captures statistical relations between certain states of the environment and goals learned from past observations.

While in this paper we focus on goal prediction problems, i.e., predicting user goals and actions, we explore data-driven methodologies and focus on a problem related to financial industry. In that context, a number of works that tackle related problems have been investigated. Baeza-Yates et al. (2015) presented a method for predicting the next mobile app a user is going to open based on their usage history and the “wisdom” of the crowd, while several works have addressed the problem of customer churn in banking with various machine learning techniques (Yaghini, Zhiyan, and Fallahi 2011; Bilal Zorić 2016; Rahman and Kumar 2020). While these works and our work in this paper address challenges in the financial industry, the specific goals and techniques employed are different, i.e., we employ LSTM models with graph embeddings aimed at understanding customer behavior and preferences.

Methodology

In this section, we describe the dataset, data preparation, and models used for predicting customer goals and actions. In particular, we employ two models – an LSTM model exploiting a bag-of-words representation, and an LSTM model enhanced with a state graph embedding. Our approach is underpinned by a comprehensive semi-synthetic dataset, capturing intricate customer interactions with diverse banking interfaces.

| Date and Time | Event |
|---------------------|---|
| 2022-02-28 18:06:08 | mobile: login |
| 2022-02-28 18:10:24 | mobile: enter menu settings |
| 2022-02-28 18:10:24 | mobile: enter menu profile-maintenance |
| 2022-02-28 18:14:40 | mobile: change information on demographic |
| 2022-02-28 18:18:56 | mobile: log-off |
| 2022-04-30 07:28:32 | web: login |
| 2022-04-30 07:28:32 | web: enter menu credit-card |
| 2022-04-30 07:28:32 | web: get information on credit-card-transaction-history |
| 2022-04-30 07:28:32 | web: exit menu root-section |
| 2022-04-30 07:32:48 | web: enter menu settings |
| 2022-04-30 07:37:04 | web: enter menu alerts-maintenance |
| 2022-04-30 07:41:20 | web: change information on alerts-definition |
| 2022-04-30 07:45:36 | web: log-off |

Table 1: Example trajectory of events for an agent of *medium income, low fail behavior, and digital interface*.

Customer Behavior Data

We utilize a semi-synthetic dataset of customer behavior generated via a domain-independent simulator proposed by Borrajo and Veloso (2020). The dataset consists of interactions between customers and a multitude of bank interfaces: bank-website, mobile app, teller counter, or ATM. Each interface offers customers a (potentially unique) set of actions such as making payments, checking rewards program information, modifying personal information, etc. A set of example actions are provided in Table 1, where we see a customer login to the mobile app and modify their personal information; the same customer then logs into the website two months later and checks their credit-card transaction history as well as modifying their alter settings. Approximately 300 actions are recorded per-customer.

Customer Goals: In addition to customer actions, a diverse set of goals can be induced from the dataset, such as:

- *check information:* The customer wants to get some information about their account (e.g., balance).
- *change information:* The customer wants to change the value of some data point (e.g., address).
- *operational goals:* The customer wants to perform some banking operations, such as deposit-cash, withdrawal, exchange, deposit-check, pay-bill, make-payment, and so on.

These goals correspond to the customers purpose for interacting with any of the bank interfaces. Customers can possess multiple goals; for example, the customer in Table 1 seeks to both check and change information while using the website (second interaction in the table).

Customer Types: The dataset also features labels for the type of customer, characterized by three attributes:

- *Income*: income level of the customer (high, medium, low, standard).
- *Fail behavior*: how frequently the customer’s action correspond to errors (rarely, often, no-failure).
- *Digital behavior*: customers preferred interface type (traditional, digital, mixed).

These attributes represent the probabilities of using different channels (web, mobile app, ATM, banker, and teller), the failure rate of operations, and types of goals. For instance, a student might be categorized as low-rarely-digital, while a medium-class worker could be medium-rarely-mixed.

Data Preparation

Next we discuss our approach to data preprocessing. The predictive features \mathbf{X} are the customer’s historical actions (e.g. Table 1). The target features y are the customer’s future actions, the customer’s goal, or the customer’s type.

Handcrafted Features: To increase the predictive efficacy of the data we first introduce several handcrafted features, and then discuss how both the handcrafted, and original features, are represented. The predictive features \mathbf{X} are provided as Time (an integer) and Action (a string). We break the Event string into more meaningful features which are listed in Table 2. The Event string contains an indicator of which interface the customer is interacting with (web, mobile, teller, ATM), which we refer to as the *primary location* feature.

Moreover, the Event string contains information indicating when the customer navigates through the interface, e.g., in Table 1 the customer has Event string “web: enter menu credit-card”, which indicates that the customer has left the web home-page (which they arrived at when logging in) and navigated to a menu of credit-card options. We refer to these features as *secondary location* features.

Defining secondary locations allows the model to more effectively learn which actions are available to customers at any given time, e.g., customers cannot modify their credit-card information without having first entered the credit-card menu. The combination of the primary and secondary location features can be interpreted as the customer’s current state.

In addition to the state features, we also define corresponding *action* features that fall into three categories: *transitioning-actions* that indicate the customer changing primary (or secondary) locations (e.g., “web: enter menu credit-card”); *information-gaining-actions* that indicate the customer obtaining new information (e.g. “web: get information on credit-card-transaction-history”); and *modification-actions* that indicate the customer modifying their information (e.g. “web: change information on alerts-definition”). Full details on the state-action features are provided in Table 2. The predictive features \mathbf{X} are thus represented in terms of these state-action pairs and a customer trajectory consists are a list of state-action pairs.

| Feature Name | Possible Values |
|---|---|
| Primary location | web, mobile, teller, banker, ATM |
| Secondary location (web and mobile menus) | credit-card, credit-score, offers, rewards, operations, settings, alerts-maintenance, contact-us, account-documents, profile-maintenance |
| Transitioning actions | login, log-off, enter enter-menu, exit, exit-menu |
| Information-gaining actions | alerts-definition, alerts-history, atm-branches, balance, benefits, demographic, documents, faq, help-call, help-email, offers, credit-card-trans-history, credit-card-trans-summary, limit-credit-card, credit-score-history, messages, rewards-activity, rewards-use-points, credit-score-summary, trans-history, trans-summary |
| Modification actions | demographic, password, user-id, limit-credit-card, alerts-definition |

Table 2: List of handcrafted features.

Graph Representation: In addition to a bag-of-word representation (Harris 1954), we also utilize a state graph embedding that will allow for better learning of each of our objectives.

In the graph embedding, nodes correspond to state features (i.e., primary and secondary location features). Nodes are connected by an edge if there exists a transition (i.e., an action that allows a customer to move between the two corresponding states). In addition to neighbor information, the node features of the graph are given as four binary indicators: the customer has visited this node in the past, the customer is currently at this node, the customer performed an information-gaining-action at this node, the customer performed a modification-action at this node. These indicators are referred to respectively as $\langle \text{past-nodes}, \text{ego-node}, \text{info-gain}, \text{modification} \rangle$.

The intuition behind the advantage of this graph embedding is that it allows the model to have a less myopic understanding of the way in which actions affect the customer’s location in each interface, which in turn allows the model to have a less myopic understanding of which actions will be available to the customer at the future time-steps as well as possible paths to nodes which may achieve the customer’s goal.

Predictive Models

To capture the temporal patterns in customer interactions and learn the intricate relationships between actions and goals over time, we utilized the architecture of LSTM networks (Hochreiter and Schmidhuber 1997). Specifically, we used a bag-of-words and one-hot encoding to represent the events as well as constructed a state graph to use as a GNN embedding in the LSTM model. The graph was constructed by examining all customer trajectories in the training dataset and defining nodes and edges as any state or transition action which appeared in more than 10 times. We found 10 to be the best frequency threshold when defining the graph as this value provided an effective balance between including irrelevant states and edges (which the model would then need to learn to ignore) and excluding important states and edges that will increase the myopicness of model predictions. Figure 1 shows an example of bag-of-words representation and state-space graph representation, respectively, generated from the datasets.

| | id | time | is_digital | is_traditional | atm | mobile | web | teller | banker | login | log-off | enter |
|------|----|------------|------------|----------------|-----|--------|-----|--------|--------|-------|---------|-------|
| 0 | 2 | 1677088680 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 1 | 2 | 1677088920 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 3000 | 16 | 1677167640 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 |
| 3001 | 16 | 1677167760 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3002 | 16 | 1677168000 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 |

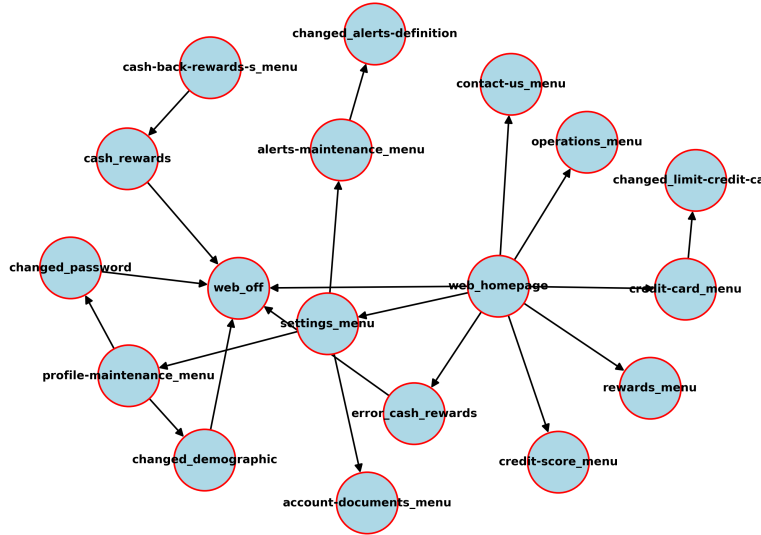


Figure 1: An example of bag-of-words and one-hot encoding representation, and a state-space graph representation.

Predictive Objective: Our primary focus will be on the following predictive tasks:

- *Goal Prediction:* Given n_H historical actions, what goal does the customer have?
- *Type Prediction:* Given n_H historical actions, what is the customer’s type?
- *Trajectory Prediction:* Given n_H historical actions, what are the customer’s next n_F future actions?

Experimental Results

In this section, we present the experimental results of our two proposed models – LSTM and GNN+LSTM. Our experimental setup consisted of the semi-synthetic dataset with 12 thousand customer interactions, split into 70% training, 15% validation, and 15% test sets. Each data point in our dataset represents a sequence of a customer’s previous 20 events. The LSTM and GNN+LSTM models were trained using the Adam optimizer (Kingma and Ba 2014), with a learning rate of 0.01, over 5000 epochs with early stopping based on the validation set performance. The primary metrics used for comparison were prediction accuracy for customer goals, agent types, and future events.

Goal Prediction: Table 3 shows the accuracy for each approach when predicting the goal of the customer. We see that both the LSTM and GNN+LSTM models are capable of accurately predicting customer goals, however neither method

| Model | Accuracy | |
|----------|------------|-------------|
| | Check Info | Change Info |
| LSTM | 71% | 68% |
| GNN+LSTM | 77% | 75% |

Table 3: Accuracy for each approach when predicting agent type. LSTM corresponds to the LSTM approach with a bag-of-words embedding and GNN+LSTM corresponds to the LSTM approach with the graph representation and GNN encoding layer. For each prediction, the model views the customer’s most recent 20 events.

achieves greater than 80% accuracy. With that said, graph embedding does offer a significant improvement to model efficacy when compared to the bag-of-words embedding.

Type Prediction: In Table 4, we see the accuracy when predicting customer type.¹ Both LSTM and GNN+LSTM have high efficacy (roughly 90% or more) when predicting both the agents failure rate type and preferred interface. In the case of preferred interface, customers tend to exclusively use their preferred interface, meaning that this predictive task becomes significantly easier once observing historical events from each customer.

Trajectory Prediction: In Table 5, we see the accuracy

¹The ground truths for each customer type (income, fail behavior, digital behavior) were stated in the dataset.

| Model | Accuracy | | |
|----------|----------|---------------|------------------|
| | Income | Fail Behavior | Digital Behavior |
| LSTM | 63% | 89% | 97% |
| GNN+LSTM | 70% | 90% | 97% |

Table 4: Accuracy for each approach when predicting customer type. For each prediction, the model views the customer’s most recent 20 events.

| Model | Accuracy | | |
|----------|----------|----------|-----------|
| | length 1 | length 5 | length 15 |
| LSTM | 52% | 40% | 33% |
| GNN+LSTM | 67% | 62% | 49% |

Table 5: Accuracy for approach when predicting the next 1, 5, and 15 customer events. For each prediction, the model views the customer’s most recent 20 events.

when predicting the future events of customers. While the accuracy on this task are notably lower than the other tasks, the space of possible predictions (i.e., the space of possible events) is far greater in trajectory prediction than in those other tasks. As is expected we see that as the model is required to forecast customer events farther into the future, its predictive efficacy decreases. However, we again see a large improvement in predictive efficacy when using the graph embedding over the bag-of-word embedding.

When examining the bag-of-words approach with an LSTM model and the graph approach with a combination of a GNN and LSTM, we observe consistent and significant improvement with the graph embedding. This is due in part to the non-myopic nature of the graph embedding in that it explicitly encodes all possible paths customers can take through each of the interfaces; as such the model to reason more efficiently about the ways in which current, or past events, influence future events. This is also visible in Figure 2, where we plot the loss functions of the LSTM and GNN+LSTM models.

Conclusions

We have provided a pipeline for predicting customer behavior, type, and objective, from observations of customers interacting with multiple bank interfaces (web, mobile, teller, ATM). This pipeline is comprised of both a feature extraction procedure which takes reordered customer “events” and builds a state-action graph as well as where states and actions are the result of hand-crafted features (Table 2). Second we use a combined GNN and LSTM architecture to make use of both the temporal and structural nature of customer interactions. We found this approach to be effective at each of the three predictive tasks, and to have consistent improvement over both baselines.

Future Work: Currently we are exploring extensions of our graph based approach to not only predict customer behavior, but also to modify customer behavior. For example, the bank may desire to migrate customers towards digital interfaces, rather than in-person interfaces, i.e., modifying

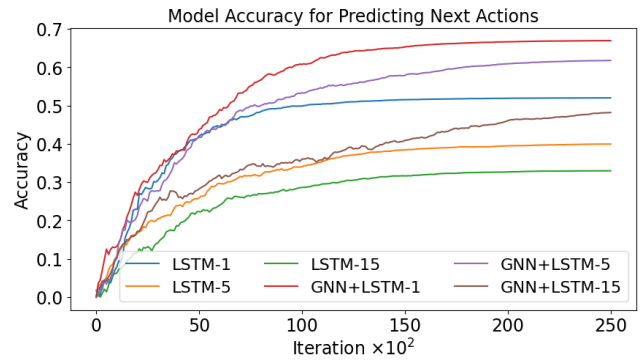


Figure 2: Loss function of LSTM and GNN+LSTM approaches when predicting the next 1, 5, and 15 actions of the customer when viewing the customer’s last 20 actions.

customer type. To modify the behavior of customers which prefer in-person interfaces, we can first learn a state-action graph for those customers, which in turn allows us to surmise the perceived cost that those individuals have for using digital interfaces (unfamiliarity with mobile apps may increase and individual’s perceived cost of taking actions within the bank’s mobile app). With these perceived costs, the bank can then use targeted rewards (e.g., cash rewards for using the mobile app) to incentivize customers to use the mobile app. After using the mobile app several times, individuals would become more familiar with app and would thus have a lower perceived cost of using the mobile app compared to in-person interfaces. Banks could then use this reward shaping technique to help migrate customers to a desired interface with greater precision than simply offering blind rewards to all individuals. Behavior prediction, which is the current focus of this paper, is an essential first-step in modifying behavior as the ability to efficiently shape behavior is directly tied to the ability to predict that behavior.

Acknowledgements

This work was funded in part by J.P. Morgan AI Research. This paper was prepared for informational purposes by the Artificial Intelligence Research group of JPMorgan Chase & Co and its affiliates (“J.P. Morgan”), and is not a product of the Research Department of J.P. Morgan. J.P. Morgan makes no representation and warranty whatsoever and disclaims all liability, for the completeness, accuracy or reliability of the information contained herein. This document is not intended as investment research or investment advice, or a recommendation, offer or solicitation for the purchase or sale of any security, financial instrument, financial product or service, or to be used in any way for evaluating the merits of participating in any transaction, and shall not constitute a solicitation under any jurisdiction or to any person, if such solicitation under such jurisdiction or to such person would be unlawful.

References

- Baeza-Yates, R.; Jiang, D.; Silvestri, F.; and Harrison, B. 2015. Predicting the next app that you are going to use. In *WSDM*, 285–294.
- Baker, C. L.; Saxe, R.; and Tenenbaum, J. B. 2009. Action understanding as inverse planning. *Cognition*, 113(3): 329–349.
- Bilal Zorić, A. 2016. Predicting customer churn in banking industry using neural networks. *Interdisciplinary Description of Complex Systems: INDECS*, 14(2): 116–124.
- Borrajo, D.; Gopalakrishnan, S.; and Potluru, V. K. 2020. Goal recognition via model-based and model-free techniques. In *ICAPS Workshop on Financial Planning (Fin-Plan)*.
- Borrajo, D.; and Veloso, M. 2020. Domain-independent generation and classification of behavior traces. *arXiv preprint arXiv:2011.02918*.
- Harris, Z. S. 1954. Distributional structure. *Word*, 10(2-3): 146–162.
- Hochreiter, S.; and Schmidhuber, J. 1997. Long short-term memory. *Neural Computation*, 9(8): 1735–1780.
- Kaminka, G.; Vered, M.; and Agmon, N. 2018. Plan recognition in continuous domains. In *AAAI*, 6202–6210.
- Keren, S.; Gal, A.; and Karpas, E. 2019. Goal Recognition Design in Deterministic Environments. *JAIR*, 65: 209–269.
- Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Rahman, M.; and Kumar, V. 2020. Machine learning based customer churn prediction in banking. In *ICECA*, 1196–1201.
- Ramírez, M.; and Geffner, H. 2010. Probabilistic plan recognition using off-the-shelf classical planners. In *AAAI*, 1121–1126.
- Shvo, M.; Klassen, T. Q.; Sohrabi, S.; and McIlraith, S. A. 2020. Epistemic plan recognition. In *AAMAS*, 1251–1259.
- Shvo, M.; and McIlraith, S. A. 2020. Active goal recognition. In *AAAI*, 9957–9966.
- Sohrabi, S.; Riabov, A. V.; and Udreă, O. 2016. Plan Recognition as Planning Revisited. In *IJCAI*, 3258–3264.
- Vered, M.; and Kaminka, G. A. 2017. Heuristic Online Goal Recognition in Continuous Domains. In *IJCAI*, 4447–4454.
- Wayllace, C.; Hou, P.; Yeoh, W.; and Son, T. C. 2016. Goal recognition design with stochastic agent action outcomes. In *IJCAI*, 3279–3285.
- Wayllace, C.; Keren, S.; Gal, A.; Karpas, E.; Yeoh, W.; and Zilberstein, S. 2020. Accounting for Partial Observability in Stochastic Goal Recognition Design: Messing with the Marauder’s Map. In *ECAI*.
- Wayllace, C.; and Yeoh, W. 2022. Stochastic Goal Recognition Design Problems with Suboptimal Agents. In *AAAI*, 9953–9961.
- Wilken, N.; Cohausz, L.; Schaum, J.; Lüdtke, S.; and Stuckenschmidt, H. 2023. Investigating the Combination of Planning-Based and Data-Driven Methods for Goal Recognition. *arXiv preprint arXiv:2301.05608*.
- Yaghini, M.; Zhiyan, T.; and Fallahi, M. 2011. A Prediction Model for Recognition of Bad Credit Customers in Saman Bank Using Neural Networks. In *ICDATA*, 1.