POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Enhancing Data Preparation with Adaptive Learning: A Contextual Bandit Approach for Recommender Systems

Laurea Magistrale in Computer Science and Engineering - Ingegneria Informatica

Author: Matteo Pancini

Advisor: Prof. Cinzia Cappiello

Co-advisor: Camilla Sancricca

Academic year: 2023-2024

---

1. Introduction

In the era of data-driven culture, companies increasingly rely on data to support decision-making processes. The Garbage-In-Garbage-Out (GIGO) problem highlights the importance of data cleaning and preparation, which are critical for the success of data-driven projects. This process involves transforming raw data into an analyzable format through standardization, error detection and correction, and duplicate removal. Given the complexity and volume of contemporary data, as well as the variety of data quality issues, significant time and resources are required for data preparation. The adoption of AI, particularly machine learning, underscores the need for accessible data preparation tools that cater to both experts and non-technical users. Various industry solutions, including self-service or black-box environments, aim to support these processes, enabling more efficient data cleaning and preparation. However, these solutions often face challenges related to explainability and the need for human intervention in decision-making. Recent advances in AI and ML have aimed at better understanding and addressing user needs. Data scientists reportedly dedicate a substantial portion of their time, often ranging between 45% and 80%, to data preparation tasks, highlighting the significance of this phase in the ML and AI workflow. Despite progress in automating ML processes, including data preparation, the adoption of automated tools presents notable challenges, such as concerns over user autonomy and potential misalignment with users' decision-making processes, that bring users to lose control (and trust) of the process [2]. To mitigate these issues, an effective strategy involves integrating Human-In-The-Loop (HITL) and Sliding Autonomy principles. HITL leverages human expertise to augment AI systems, ensuring decisions consider relevant contextual factors. Sliding Autonomy allows for a flexible system that adjusts the level of automation based on the user's preferences and expertise in data preparation and analysis [1].

Additionally, the abundance of online information can overwhelm decision-makers. Recommender systems alleviate this burden by predicting preferences and suggesting relevant items. Nevertheless, they encounter challenges such as the cold start problem, limited diversity, scalability issues, and occasionally suboptimal recommendation quality. Reinforcement Learning

(RL), a method of trial-and-error in dynamic environments, emerges as a promising solution, particularly in the context-aware systems of recommender systems. These systems utilize contextual information to enhance recommendation quality within a multi-armed bandit model architecture, offering multicriteria ratings and flexible recommendations effective in uncertain environments and online decision-making. The inherent adaptability of RL algorithms to learn from environmental rewards without predefined training data positions them well for recommendation scenarios characterized by high uncertainty [4]. Leveraging both historical and user session contextual data has demonstrated generalized effectiveness in recommender systems [3]. Contextual-bandit recommendation systems, in particular, present an opportunity to enhance data preparation tasks. These systems offer adaptive learning capabilities, handle a finite number of actions, and manage uncertainty, aligning well with the dynamic nature of data preparation. Furthermore, they enable personalization of recommendations based on user interactions, which is crucial for enhancing user interaction and satisfaction. By ensuring recommendations are relevant and timely, these systems improve the overall user experience in the development of data preparation assistance tools.

This thesis poses the fundamental Research Question: *"Can we recommend an optimal data preparation technique by exploiting user feedback and empirical knowledge?"* The primary goal is to develop a continuous and adaptive learning framework that enhances data preparation tasks through targeted technique recommendations tailored to each user's specific context. This is achieved by leveraging historical interaction data and contextual information derived from textual analysis. Basically, all the variables (such as Session Variables) and UX-related components will be merely described as instrumental for the framework but will be tested in future works. To address the Research Question, three main sub-questions are proposed:

1. *"Can we use context-based policy?"*
2. *"Can a recommender system outperform existing literature?"*
3. *"Does the presence of empirical knowledge influence learning capabilities?"*

2. Method

The architecture's main use case scenario is straightforward: users have a dataset they want to analyze, and the architecture guides them through an optimal data preparation pipeline for that dataset and the type of analysis they intend to perform. A high-level representation of the overall architecture is shown in Figure 1. At the beginning of the process, the user uploads their dataset. Following a description and assessment phase, the system requires two critical pieces of information: a user profile and a data profile. The user profile contains details such as the goal of analysis, the ML algorithm for the analysis, expertise level, and automation preferences. This information is gathered either from textual or bullet point input from the user. The data profile is automatically generated by another tool layer, which gathers information about the dataset's data quality issues (establishing a ranking) and the dataset's characteristics (e.g., min, max, skewness, completeness). At this point, the actual data preparation phase begins. The data preparation process is divided into several steps, each focusing on a different set of actions. For each *Data Quality Dimension* considered, a two-stage contextual-bandit-based recommender system is used. Firstly, it creates a ranking of the techniques for data preparation to address that dimension. Secondly, it selects the number of techniques to show to the user. Based on the presented techniques, users make their informed choices. This architecture employs a Human-In-The-Loop (HITL) approach. Throughout the entire data preparation process, users are continually involved at various levels. Users can accept the suggestions they receive, but they can also reject them. None of the recommended techniques and methods are mandatory; if users are not satisfied with them, they can independently choose the data preparation pipeline to perform. Complete freedom is always guaranteed.

Furthermore, a sliding autonomy (SA) model is incorporated, enabling varying levels of user autonomy. This design caters to expert users desiring greater flexibility in decision-making and non-expert users preferring more structured guidance, allowing the system to adjust its automation level accordingly.

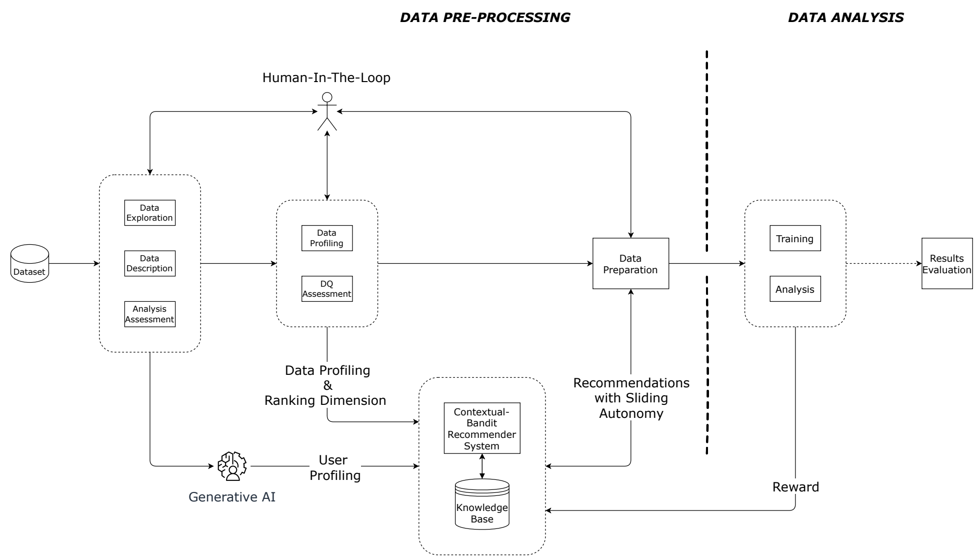

Figure 1: High-Level view of the overall architecture.

All user choices, interactions, characteristics, and recommendations are saved in the knowledge base, which provides the foundation for creating the context to feed the contextual-bandit recommender system. The architecture emulates the structure of a contextual-bandit problem. In this setup, the user environment interacts with the RL-based recommender system, which functions as the learner. This interaction framework is fundamental for tailoring system recommendations to user-specific contexts, thereby enhancing the personalization of the adaptive system.

Personalization is achieved through the integration of session variables, specifically user expertise and automation preference, which tailor the system's recommendations and level of guidance to match individual user profiles throughout the entire session of interaction. These variables are instrumental in personalizing the user experience, aligning with both the user's knowledge and their desired degree of automation. The generation of context vectors is a fundamental component of the methodology. These vectors integrate users' characteristics, needs, and preferences with the data quality dimensions and historical interactions of techniques throughout the decision-making process. The generation of context vectors is critical for integrating users' characteristics, needs, and preferences with the data quality dimensions and historical interactions of techniques throughout the decision-making process.

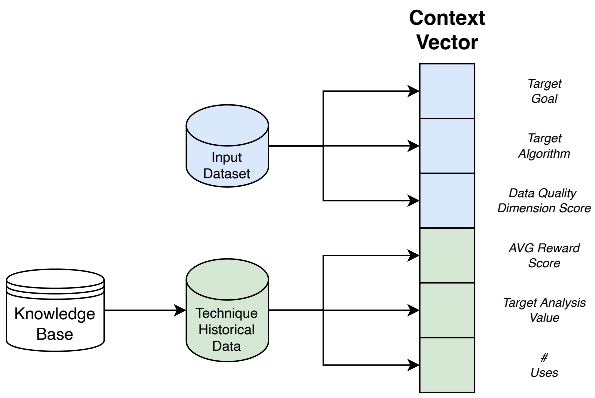

Figure 2: Context vector architectural model.

User-centric features, such as *Target Goal* and *Target Algorithm*, will be explicitly requested along with session variables and evaluated during the Analysis Assessment Phase. In this phase, an internal tool will identify and present potential analysis goals and algorithms suited to the user's dataset. The context vector, unique to each arm (in this case, each imputation technique), involves analyzing both user-specific characteristics and the knowledge base to gather specific historical data and statistics related to each technique, as illustrated in Figure 2. By employing both session-specific characteristics and historical data, recommendations can be tailored to incorporate insights from past interactions and adapt to the specific context of the user.

In the context of the contextual-bandit recommender system, the context vector for each rec-

ommendation step (guided by the *Data Quality Dimension*) comprises two main components. The first component, consistent across each arm, is based on user input, including the dataset and textual context information. The second component, varying by arm, encompasses the historical information and characteristics of each technique. The user-related features include the *Target Goal*, the *Target Algorithm*, and the *Data Quality Dimension Score* (in our case, the completeness score, representing the percentage of missing values in one column of the dataset). The arm-related features consist of the *AVG Reward Score* (mean reward achieved by a technique across all implementations), the *Target Analysis Value* (average final ML value obtained by the *TopK* data objects that most closely match the current object under analysis in terms of profiling, adhering to a predefined $similarity_threshold$), and the *Number of Uses* (frequency with which a specific technique has been used). The context vector will guide the learner in the decision-making process by combining these user-related and arm-related features to tailor recommendations effectively.
The learner is the core of the system's intelligence. It is composed of two sub-modules:

- **Adaptive Slate Number Recommendation Learner**: it is responsible to choose the number of recommendation to show to the user.
- **Contextual Bandit Recommender System**: it is responsible to learn the best possible techniques for the specific user context.

The **Adaptive Slate Number Recommendation Learner** employs a multi-armed bandit-based learner, based on the Thompson-Sampling policy, optimized to adjust the volume of recommendations according to users' desired level of autonomy. The objective is to personalize content delivery by exploiting feedback mechanisms. Figure (3) provides a high-level overview of the Slate Number Learner, illustrating the module's framework and operational dynamics.
The **Contextual Bandit Recommender System** is a contextual bandit framework, based on LinUCB policy, selected for its capability to personalize content delivery despite the complexities of big data. It receives as input the *Context Vector* and its parameters vector and

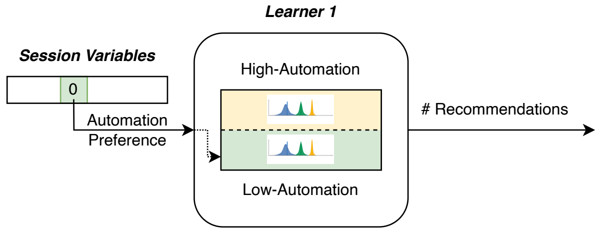

Figure 3: High-Level Overview of Adaptive Slate Number Recommendation Learner.

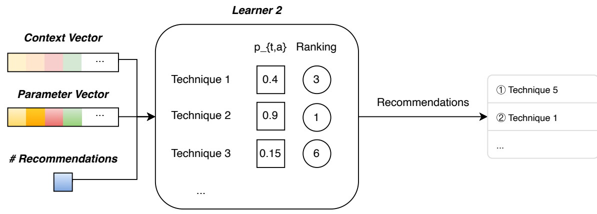

Figure 4: High-Level Overview of Contextual-Bandit Recommendation System.

outputs the recommended techniques based on the optimal number calculated by the *Adaptive Recommendation Manager*. Figure (4) presents a high-level overview of the contextual bandit recommender system. It illustrates the calculated probability $p_{t,a}$ for each technique (arm) a at time t, and the corresponding technique rankings at that timestamp. The last fundamental component is the reward mechanism, meticulously designed to ensure recommendations align with user preferences and closely achieve the analysis's final objectives. The reward calculation for each data preparation technique, corresponding to distinct arms of the system for each data quality dimension, is structured as follows:

$$R = \alpha U + \beta \Delta_{Analysis\ Value}$$

where U represents the user choice (whether or not the user chose our recommendation), and $\Delta_{AnalysisValue}$ is equal to $Final\ Analysis\ Value - Target\ Analysis\ Value$, which is the difference between the actual ML value obtained by the user and our internal KB benchmark. By addressing both personal user choices and tangible technique performances, the reward mechanism promotes continuous improvement in the relevance and efficacy of recommendations.

3. Results

A first sensitivity analysis has been conducted on the four main parameters of the system: Confidence ($\alpha\_confidence$) of the recommendation system, Reward parameters (α_reward and β_reward), Similarity threshold ($sim_threshold$) used in dataset profiling from the knowledge base, and Number of Top K recommendations ($TopK$) to consider from the knowledge base. The Mean Average Precision (MAP) at cut-off 1 and 3 was used as a key evaluation metric for testing. This metric was chosen to simulate not only the best and optimal recommendation but also the average slate. The best results were found for: $\alpha\_confidence = 0.9$, which showed increased exploration as valuable given the very heterogeneous context of data preparation; $\alpha\_reward = 0.9$ and $\beta\_reward = 0.1$, considering the fact that in the original dataset, the differences between minimum and maximum values per each possible data object are in the order of 0.02, making the likability of the recommendation more valuable for learning. The $sim\_threshold = 0.9$ was found to be optimal, considering a more skewed number of past interactions as valuable for creating the benchmark, and $TopK = 5$ was determined as a good balance of past interactions. After obtaining the best and most influential parameters, the next step is to run the experiments to answer the main research question and its subquestions. The first question is: *"Can we use a context-based policy?"* To answer this question, the *Adaptive Contextual-Bandit with KB* policy was compared with other state-of-the-art RL policies (*Thompson Sampling*, *Upper Confidence Bound*, *Epsilon-Greedy*, and *Random*). This comparison contextualizes the effectiveness of the approach within the range of existing strategies and demonstrates its relative merits and potential areas for improvement.

Policy	Cumulative Reward	Cumulative Regret	MAP@3	MAP@1
Adaptive LinUCB with KB	**4406.7122**	**1407.8634**	0.75000	**0.80525**
Thompson Sampling	4404.9058	1409.6696	0.73000	0.74250
Upper Confidence Bound	4397.2204	1417.3546	0.72325	0.72800
Epsilon-Greedy	4402.0698	1412.5052	**0.76650**	0.77750
Random	4397.4036	1417.1712	0.07600	0.11825

Table 1: Results of simulation of different policies. Best results in **bold**, worst results in red.

The results in Table 1 present relatively close performance in cumulative rewards among the strategic policies, suggesting that each can effectively accumulate value over time. However, the distinction in MAP scores, particularly the Top-1 precision, highlights the advantage of integrating contextual understanding and historical data, as demonstrated by our *Adaptive LinUCB with KB* approach. This approach shows a +8% improvement compared to *TS*, +11% compared to *UCB1*, and +4% in identifying the Top-1 best performing technique. Higher results of *Epsilon-Greedy* in MAP@3 were demonstrated through the analysis of arms distribution, in which the *KNN* technique, identified in previous analyses as one of the most effective imputation methods in our dataset, substantiated *Epsilon-Greedy*'s tendency to lean towards this arm as a strategic advantage in maximizing immediate rewards. The propensity of *Epsilon-Greedy* to leverage historically successful techniques underscores the need for a robust initial dataset when training recommendation systems. This strategy also reveals potential risks in scenarios where data characteristics or underlying patterns evolve, potentially diminishing the effectiveness of previously optimal techniques. Policies that integrate both exploration and exploitation, such as our *Adaptive LinUCB with KB*, are likely to offer greater resilience against such changes by continuously adapting to new information.

Dataset	Cumulative Reward	Cumulative Regret	MAP@3	MAP@1	Classifier MAP@1
BachChoralHarmon	4194.590	432.581	0.937	**0.950**	0.924
Run_or_walk_information	4141.683	519.306	0.487	0.500	0.900
abalone	4128.981	528.017	0.795	**0.848**	0.594
bank	4138.961	521.835	0.901	**0.933**	0.667
cancer	4144.106	534.334	0.942	**0.965**	0.511
default of credit card clients	3962.213	846.740	0.825	**0.843**	0.5167
drug	4192.112	435.206	0.157	**0.244**	n.a.
electricity-normalized	4153.338	507.131	0.468	0.483	0.800
fried	4122.137	571.776	0.751	0.820	0.838
frogs	4011.765	783.240	0.784	**0.850**	0.827
german	4182.220	460.974	0.550	**0.583**	0.225
house	3849.020	1061.412	0.971	**0.974**	0.688
iris	4171.997	472.261	0.415	**0.458**	0.325
letter	4027.443	713.646	0.371	0.384	0.687
mv	4145.677	515.693	0.755	**0.805**	0.383
numerai28.6	3978.826	792.950	0.901	**0.917**	0.725
phoneme	4172.887	471.074	0.339	0.316	0.641
ringnorm	4009.025	770.063	0.906	**0.971**	0.970
shuttle	4159.521	518.485	0.805	0.815	0.850
stars	4176.472	467.612	0.126	0.138	0.591
visualizing_soil	4204.416	423.129	0.329	**0.45**	0.366
wall-robot-navigation	4168.779	475.653	0.198	0.397	0.733

Table 2: Results of Leave-One-Dataset-Out Simulation across various Datasets. In **Bold** MAP@1 results in which our recommender systems outperform old classifier.

As regards the second question, *"Can a recommender system outperform existing literature?"*, a Leave-One-Out (LOO) approach was used to simulate a real-world scenario. This approach tests the capabilities of the recommender system with completely new datasets.

The results, summarized in Table 2, show that our recommendation system achieves better results on 65% of the datasets compared to the old classifier (a Random Forest-based model), with an average improvement of 30%. Moreover, these results were achieved using just 10% of the data for training, compared to the data usage of the original classifier, indicating a far more efficient solution. Given the favorable results observed in the policy comparison for *Epsilon-Greedy*, a further analysis was conducted using a Leave-One-Out (LOO) simulation. In this analysis, LinUCB outperformed Epsilon-Greedy by an average of 7% in MAP@1 while achieving identical average results in MAP@3. An analysis of the datasets where performance was worse revealed that the system can suffer from arm bias during training. This bias can cause adaptation to take longer and be very data-intensive.

Finally, to address the question *"Does the presence of empirical knowledge influence learning capabilities?"*, three configurations have been tested: one with KB = training set, one with no use of a knowledge base, and one with our proposed dynamic KB.

Configuration	KB-Train Split	Cumulative Reward	Cumulative Regret	MAP@3	MAP@1
KB = Training Set	10000-5000	1243.907	877.672	0.782	0.803
No KB	0-5000	**1445.181**	**676.398**	0.724	0.784
Dynamic KB	5000-5000	1267.48	854.099	**0.812**	**0.838**

Table 3: Results of Knowledge Base Impact simulations. Best results in **Bold**.

The results, summarized in Table 3, show significant variations in performance across different knowledge base configurations. Notably, the configuration with a *Dynamic KB* achieves the highest MAP@1 and MAP@3 scores, marking increases of 4.47% and 3.84%, respectively, compared to the *KB = Training Set* setup. The *KB = Training Set* configuration seems to restrict the system's learning capabilities and generalization. Having access to the entire training set as a knowledge base might cause the system to be overly tailored to those specific interactions, potentially reducing its exposure to a broader range of scenarios necessary for robust learning. This situation can lead to overfitting, characterized by strong performance on familiar data but poor adaptability to new or unseen data. Although the *No KB* configuration records the highest cumulative reward, it falls short in MAP metrics, showing a decrease of 11% in MAP@3 and 7% in MAP@1 compared to our main *Dynamic KB* approach. This discrepancy likely stems from the reward structure, which prioritizes immediate gains over the optimization of long-term learning and accuracy.

4. Conclusions

This thesis presents a methodology to suggest to users which imputation methods are more suited for their specific context and needs, such as dataset profiles and data analysis goals, to improve the completeness dimension of their data. More precisely, the aim is to provide users with the correct methods to impute the missing values of a specific column of the input dataset. To revisit the initial **Research Question**: *'Can we recommend an optimal data preparation technique exploiting user feedback and empirical knowledge?'* This objective is accomplished by developing an adaptive contextual-bandit recommender system that, leveraging an internal knowledge base, is able to exploit both contextual and historical information to recommend the optimal imputation technique and learn from users' choices and feedback. The recommender system has proven to be reliable and robust in different settings with promising results. The policy, built on top of Linear UCB, has shown good performance compared to well-established policies, like TS (+8% in terms of MAP@1), UCB1 (+11% in MAP@1), and Eps (+4% MAP@1). Under Leave-One-Out conditions, the recommender system has shown a median MAP of 81% (+20% compared to the old classifier), while using just 10% of the total training dataset. Possible improvements have been delineated due to some poor performance in certain datasets: the approach can adapt to new and completely different data, but it takes time to change the knowledge created in the past. The presence of the Knowledge Base has been proven valuable, providing, under the same conditions, an increase of 14% in MAP@1 compared to not using any knowledge. Transitioning from a classifier-based approach to an adaptive contextual-bandit system, specifically the LinUCB model with a knowledge base, introduces several significant advantages, as outlined in Section **??**, including an average increase in Mean Average Precision at rank 1 (MAP@1) of 18%. Firstly, the adaptive LinUCB model greatly enhances realtime learning capabilities, allowing for continuous model updates without the need for retraining. This adaptability ensures that the system reflects current data trends more accurately. Secondly, LinUCB adeptly manages the exploration-exploitation trade-off, meaning it not only leverages well-established techniques but also considers new and potentially more effective imputation methods, thereby optimizing performance over time. Additionally, LinUCB's capability to incorporate a diverse array of contextual information enables it to offer personalized recommendations that align closely with specific user needs and data characteristics, significantly enhancing the relevancy of its recommendations. Lastly, the integration of a knowledge base within the LinUCB framework supports efficient decision-making by leveraging historical interactions, reducing computational overhead and scaling more efficiently compared to traditional static classifiers, by leveraging user's interactions. These improvements underscore the LinUCB system's suitability for dynamic environments such as data preparation, where adaptability and efficiency are crucial, making it an ideal solution for continuously improving data-driven decisions. Future work could include incorporating additional types of analysis, such as Regression and Clustering, beyond the current focus on Classification. Expanding the data preparation processes beyond missing data imputation, such as using reinforcement learning policies like Q-Learning, could optimize recommendations throughout the data preparation phase. Further development could involve creating and deploying a minimum viable product (MVP) of the tool, testing it in real-world environments, and assessing the impact of session variables like *Automation Preference* and *User Expertise*. Additionally, exploring the use of Large Language Models (LLMs) to collect session variables and user-related features for the context vector will enhance personalization and system effectiveness.

References

[1] Cinzia Cappiello, Chiara Francalanci, and Barbara Pernici. Data quality assessment from the user's perspective. In *Proceedings of the 2004 international workshop on Information quality in information systems*, pages 68–73, Paris France, June 2004. ACM.

[2] Arun Kumar. Automation of Data Prep, ML, and Data Science: New Cure or Snake Oil? In *Proceedings of the 2021 International Conference on Management of Data*, pages 2878–2880, Virtual Event China, June 2021. ACM.

[3] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, Raleigh North Carolina USA, April 2010. ACM.

[4] Nicollas de Campos Silva. Active learning in contextual bandits: handling the uncertainty about the user's preferences in interactive recommendation systems. July 2023. Accepted: 2023-12-19T19:51:02Z Publisher: Universidade Federal de Minas Gerais.