Case-based reasoning, Multi-Agent Systems and Hawkes Process for ITS

Daniel Soto-Forero, Marie-Laure Betbeder and Julien Henriet

DISC, Université Marie et Louis Pasteur, CNRS, institut FEMTO-ST, 16 Route de Gray, Besançon, F-25000, France.

{daniel.soto_forero, marie-laure.betbeder, julien.henriet}@univ-fcomte.fr

Ensemble Stacking Case-Based Reasoning and a Stochastic Recommender Algorithm with the Hawkes Process Applied to ITS AI-VT

Daniel Soto-Forero10000-0003-0753-4673
Marie-Laure Betbeder10000-0002-8103-4098
Julien Henriet10000-0002-7671-4574

August 18, 2025

Abstract

This paper presents a recommender algorithm integrating a multi-agent ensemble case-based reasoning (ESCBR-SMA), a Thompson sampling-based (TS) recommender system, and a Hawkes process. The final integrated algorithm is applied to improve the real-time adaptation of an Intelligent Tutoring System called AI-VT. We have compared the static recommendation algorithm (ESCBR-SMA with TS) and the dynamic recommendation algorithm (ESCBR-SMA, TS with the Hawkes process) by evaluating the knowledge acquisition evolution of each learner. The metrics used allow us to determine the stability of prediction and change in the probability distributions for each learner and each level of complexity. The results show that the integration between stochastic adaptation, the prediction with the case-based reasoning paradigm, and the Hawkes process allows reinforcement of knowledge as well as a more realistic estimation of the recommendation for each case independently.
Case-Based Reasoning, Stacking, Regression, Ensemble Methods, Stochastic Recommender, Intelligent Tutoring System, Machine Learning

1 Introduction

The AI-VT (Artificial Intelligence - Virtual Trainer) system is an Intelligent Training System (ITS) created to assist learners in understanding and acquiring knowledge in various domains. The system is generic and aims to identify learner weaknesses and adapt the exercises accordingly to improve their learning indicators in a personalized way regardless of the course content or domain of study. The system uses a database of questions associated with multiple skills depending on the domain. These questions are organized according to the level of complexity estimated for the learner.

There are different classifications of an ITS, one of which is that an ITS is composed of four elements: an expert model, a student model, pedagogic knowledge, and an interface. These components interact with each other to make the system dynamic and capable of modeling the learner in various scenarios to build a personalized curriculum [3]. Another possible classification is one that divides an ITS into three logical layers: a presentation layer (user interface), an e-learning system layer (course enrollment and management, user profile and activities, teaching or learning assessment and feedback, user communication, and collaboration), and a data layer (collected, stored, and used education data) [17]. In any case, this kind of system allows the development of individual learning education, which is much more effective than classroom learning [9].

One of the main modules of an ITS is the recommender system, which aims to find weaknesses and adapt the platform locally or globally to facilitate the learning process and knowledge acquisition. This module is very important because it allows adaptation of the system and customization of the contents and exercises according to the needs and results of each learner. The effectiveness of the system in the acquisition of knowledge and adaptation to different types of learning depends on this module [17]. It is therefore necessary to find techniques and algorithms that can exploit the available data and explore the learning options dynamically, thereby improving the overall performance of the ITS.

The contributions of this paper are:

- Forgetting curve simulation in the learning process using the stochastic Hawkes process.
- Integration of case-based reasoning, multi-agent systems, and the Hawkes process in a recommender algorithm.
- Verification of the progression, stability, and precision of the proposed stochastic recommendation algorithm using simulated-student database and heterogeneous real student database.

This paper is organized as follows: Section II presents a background of definitions and concepts. Section III contains the related works about case-based reasoning, ensemble techniques, Thompson sampling, and regression. The proposed algorithm is explained in Section IV. Section V shows the experimental description, the results, and the discussion. Lastly, the conclusions and future work are discussed in Section VI.

2 Background

This section introduces the concepts and definitions necessary to understand the proposed algorithm as well as fundamental models and metrics. The first fundamental paradigm used in this work is case-based reasoning (CBR), which is used to exploit historically acquired knowledge and accumulated experience with respect to a specific problem. This paradigm is used to generate emergent solutions for a new problem using a knowledge database. The main idea is to search for similar past situations and use the experience to solve new problems. CBR is especially useful when the underlying causes of a problem are not well understood. CBR defines a cycle of four steps to propose a solution [18].

The proposed recommendation algorithm associated with AI-VT is based on the reinforcement learning paradigm. Reinforcement learning is a machine learning technique that allows, through actions and rewards, improvement of the system's knowledge about a specific task [1]. The algorithm used for adaptation is a reinforcement learning algorithm called Thompson Sampling that, through an initial probability distribution (an a priori distribution) and a set of predefined update rules, can adapt and improve the initial estimates of a specific analyzed process [22]. The initial probability distribution is generally set up as a specific distribution of the Beta family of distributions (Equation 1) with initial predetermined values for α and β [30, 21].

$$Beta(\theta|\alpha,\beta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1} \tag{1}$$

where Gamma function Γ is formally defined as equation 2.

$$\Gamma(x) = \int_0^\infty t^{x-1}e^{-t}dt \tag{2}$$

Using the explicit Γ definition and a double variable replacement, the family of beta distributions can be written as in Equation 3. The metrics used in this paper are written based on this equation.

$$Beta(\theta|\alpha,\beta) = \frac{\theta^{\alpha-1}(1-\theta)^{\beta-1}}{\int_0^1 t^{\alpha-1}(1-t)^{\beta-1}dt} \tag{3}$$

The forgetting curve is an important component of human learning, and for Intelligent Tutoring Systems (ITS), it is a good indicator to evaluate the long-term retention of knowledge and to adapt the system content [32]. Generally, the forgetting curve is modeled using a decreasing exponential function associated with time [25].

In this paper, we use the Hawkes process to simulate the forgetting curve. The Hawkes process is a class of self-exciting point processes whose jump rate is determined by their history. They are usually considered continuous-time processes but can also be used with discrete-time processes. Formally, the Hawkes process can be described as shown in Equation 4 by an intensity function μ and an excitation function ϕ, depending on time t and history events t_i [24].

$$\lambda(t) = \mu(t) + \sum_{t_i<t}\phi(t-t_i) \tag{4}$$

The prediction used in the proposed algorithm is based on the work of Soto *et al.* [28]. It is a case-based reasoning stacking algorithm that implements two levels of integration. Globally, it uses the stacking strategy to run multiple algorithms to search for information in a dataset and generate solutions to different generic problems. In addition, there is an evaluation stage that allows the selection of the most optimal solution for a given problem according to an adaptive metric defined for regression problems. We decided to implement the stacking-based algorithm because it is an ensemble method based on Stein's paradox since it combines the points of view of different estimators to the case-based reasoning retrieve and reuse stages [19].

3 Related Works

To improve the learner experience, an ITS adapts the contents to the needs and knowledge of each learner to enable each one of them to advance in their knowledge acquisition and achieve their learning objectives. This identification and adaptation are generally made with new artificial intelligence techniques such as neural networks, Support Vector Machines (SVM), decision trees, Naive Bayes[2], or long short-term memory networks (LSTM) [29]. Other techniques have been employed to enhance the ITS functionalities, such as the work of Arnau-Gonzalez *et al.* [4] that allows natural language interaction with the system, implementing a conversational agent through the artificial intelligence-driven natural language understanding (NLU) using the Rasa framework. The results show good cooperation between the NLU and the ITS in producing consistent dialogue and identifying user intents with high precision.

An application of reinforcement learning can be found in the work of Mao *et al.* [20], where a reinforcement learning-based two-sided recommender system (RTR) is proposed to personalize a quiz, selecting the relevant questions for each learner by considering different parameters such as knowledge level, question type, and question difficulty. The reward for each question is calculated with a learner preferences evaluation. To test and compare the model, a simulation was performed with random selection and greedy selection. The results using the cumulative reward are better with the RTR after 10 and 30 steps.

The integration of historical data and insights gained through an ITS application enhances the effectiveness of artificial intelligence algorithms, facilitating better system adaptation as demonstrated by the method proposed by Li *et al.* [16], which leverages knowledge graphs (KG) to incorporate the structural information of knowledge concepts. By doing so, it enables an Intelligent Tutoring System (ITS) to select questions that are more informative and representative. The method is composed of two main elements. The first one obtains previous information from the learner and proposes a list of possible questions to be followed in the learning sequence. The second one is in charge of evaluating each one of the proposals predicting of the learner's performance so that it can proceed to select the question that potentially gives a higher performance. The model was tested with two public datasets: ASSISTments 2009-201 and Eedi 2020. The results demonstrate that the system can recommend correct exercises to the learners, improving their performance.

The use of information to predict learner performance can contribute to system adaptation by anticipating possible changes and evaluating possible adaptations before proposing them to learners. The work of Clemente *et al.* [8] infers information about the progress of the learners and a flexible and adaptable ontology based on competencies to detect and correct weaknesses, and to adapt the recommender system. To create the ontology, some criteria and key points have been defined. The global architecture has been divided into two models. The first one contains the information about the learner, and the second one has the system recommender rules. In general, the effects of adaptation and personalization of an ITS on learning and knowledge acquisition are positive. As an example, we can look at the work of Badier *et al.*[5], where a mobile recommendation application has been implemented to adapt the navigation and pedagogical resources according to the results and interests of each learner using a three-module architecture. In this case, the metrics used measure the use of the application rather than the performance of the learners to evaluate whether the recommendations provided by the proposed model allow greater interaction of the learner with the application. If the interactions with the application increase, it means that the learners work longer as demonstrated by the results obtained. The authors can then conclude a positive effect on learners. Prediction with case-based reasoning is possible given the implicit analogical reasoning process. The analogical process is able to work with a small number of instances to handle context or to allow creativity. The principle of the analogical process is that similar situations have similar outcomes [6]. In the work of Louvros *et al.* [18], case-based reasoning is used to predict the real-time survivability of ships. The proposed model combines machine learning predictions and case-based reasoning in similar cases, where the machine learning gets a prediction based on the case-based reasoning results. The goal is to predict the evolution of ship damage scenarios in real time. Chun *et al.* [7] also predicts stock prices with an adaptation of case-based reasoning to retrieve neighboring cases using graphical pattern identification. In this case, data are represented as a time series, thus demonstrating that the reasoning from cases is adapted to this type of representation for the retrieve phase where the model gets acceptable results.

Another application appears in Pei *et al.* [23] leveraging the good performance of case-based reasoning for predicting the hazard grade of coal spontaneous combustion. The complete model integrates the reasoning from cases with principal components analysis (PCA) and fuzzy clustering (FM), obtaining good prediction results and improving the computational efficiency of the calculations.

In general, recommender systems are used in various fields, ranging from the sciences to online product stores. This type of system facilitates decision-making and allows in some way to customize content and/or products, as can be seen in the works mentioned below. Iftikhar *et al.* [13] modeled a recommender system with the Markov Decision Process (MDP). The complete model is composed of multiple stages that seek to reorganize the information of the user evaluations for different products according to a bi-cluster representation and thus identify user preferences and decide on personalized recommendations. The model obtains information from multiple users on all products and can use this information to tailor recommendations to specific preferences. The results show that the proposed algorithm achieved a better start state that yields an optimal policy to achieve the goal. However, a common obstacle with recommender systems is the cold-start problem, which consists of generating recommendations without historical data or with little initial data. To try to solve that problem, reinforcement learning is generally used, as can be seen in the work of Giannikis *et al.* [12], where reinforcement learning has been used successfully and has obtained better results than some of the more popular AI paradigms. A reinforcement learning algorithm widely used in recommender systems for its ability to work with data whose level of uncertainty is high and that also provides acceptable solutions in cold-start cases is Thompson Sampling. Zhu and Van Roy [34], propose a recommender system based on an epistemic neural coupling with TS to solve a recommender problem defined as a contextual bandit problem. TS is also used because it is a good algorithm for exploring the research space and keeping the computational cost at a minimum. The experiments with two databases in comparison with other algorithms present better scalability. Also, Ghoorchian *et al.* [11] integrate the Thompson Sampling strategy after a random projection to reduce the dimensionality, because high dimension can reduce the TS accuracy. By posing the recommendation problem also as a multi-armed bandit problem, the model was compared with other recommendation algorithms on three different databases and showed an average gain in computation time and cumulative reward. Another work with TS applied to a recommender system is Eide *et al.* [10], which proposes a dynamic sequential recommender system based on Thompson Sampling. The model changes the recommendations over time according to the evolution of user data preferences, demonstrating that TS applied sequentially allows for increasing the diversity of the search space exploration and improvement of the specific learning algorithm. Since the recommendation in ITS is also highly variable per learner and dynamic over time, using a TS-based algorithm is a good strategy. This has been seen in previous works, including the work of Soto *et al.* [26], which serve as the model presented in this paper. That model uses the Beta probability distributions family to estimate learner knowledge and adapt an ITS to each learner. The basic Thompson sampling model has been mixed with the stratification sampling, and the information is updated in a correlated manner to get better estimations of a learner level in each complexity level and to avoid the Simpson's paradox.

The dynamics of the Hawkes process are useful to some tasks because they improve the results and allow simulation of real-life situations. The work of Zhang *et al.* [33] uses the Hawkes process to predict user preferences in a spacio-temporal context based on historical sequential data to improve the recommendations. The proposed approach outperforms the baseline. In the case of simulation, the work of Lamprinakou and Gandy [15] uses the Hawkes process with stratification to make an epidemic model more realistic. The model produces dynamics very similar to the spreading of a real epidemic, so it is possible to characterize behavior and make predictions to improve prevention measures.

4 Proposed Algorithm

The proposed algorithm is an integration of stochastic adaptation (Thompson Sampling based), ensemble case-based reasoning (ESCBR-SMA), and the Hawkes process. In this case, the recommender algorithm produces an adaptation according to learner grades, the ESCBR-SMA performs a prediction to validate the generated adaptation, and the Hawkes process simulates the forgetting curve in the learning process.

The idea of unification is to obtain information from the local point of view where a recommendation is obtained using only the information of individual learners (Thompson Sampling-based model), the global prediction (where the information is obtained from all learners who have similar results through a collaborative filter with CBR), and the dynamic learning process with the Hawkes process. The algorithm architecture is shown in Figure 1, where it can be seen that TS and CBR are executed in parallel and independently with the information extracted from the same database. Once the results of each algorithm are obtained, the results are unified through a weighting function, and the distribution of probabilities are updated dynamically according to past events and the selected complexity level. The final recommendation is the one that maximizes Expression 7. Consolidating the two global results allows mitigation of the effect of Simpson's paradox [31].

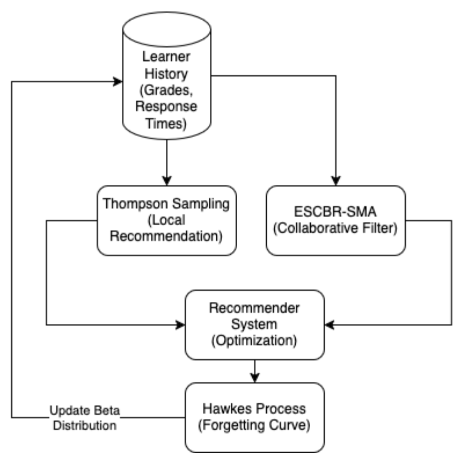

Figure 1: Proposed Algorithm Architecture

The first step is the adaptation with Thompson Sampling and the ECBR-SMA prediction. Then the decision is made and sent to the learner. The recommender system obtains a probability value for all the complexity levels for the learner, and the ECBR-SMA evaluates the proposition with a prediction for each complexity level. Table 1 shows the variables and parameters for the proposed algorithm and the employed metrics.

The integration is made in three steps. The first step is to get random values

Table 1: Parameters (p), variables (v), and functions (f) of the proposed algorithm and metrics

ID	Type	Description	Domain
α	p	Beta distribution parameter	$[1, \infty] \in \mathbb{R}$
β	p	Beta distribution parameter	$[1, \infty] \in \mathbb{R}$
t	p	Time defined as iterations	$\mathbb{N}$
c	p	Complexity level	$\mathbb{N}$
x_c	p	Mean grades for complexity level c	$\mathbb{R}$
y_c	p	Number of questions for complexity level c	$\mathbb{N}$
n_c	v	Normalized probability value for complexity level c	$[0, 1] \in \mathbb{R}$
TS_c	v	Thompson sampling reward for a complexity level c	$[0, 1] \in \mathbb{R}$
TSN_c	v	Normalization of TS_c with others complexity levels	$[0, 1] \in \mathbb{R}$
$ESCBR_c$	v	Grade prediction for a complexity level c	$\mathbb{R}_+$
p_c	f	Probability density function for complexity level c	$\mathbb{R}_+$
r	f	Recommender metric function	$[0, 1] \in \mathbb{R}$

for each c complexity level using the probability distributions generated with the TS (Equation 5). Once all the probability values corresponding to all the levels of complexity have been obtained, the normalization of all of them is calculated as shown in Equation 6. The normalization values serve as priority parameters for the predictions made by the ESCBR-SMA as calculated in Equation 7.

$$TS_c = rand(Beta(\alpha_c, \beta_c)) \tag{5}$$

$$TSN_c = \frac{TS_c}{\sum_{i=0}^{4} TS_i} \tag{6}$$

$$n_c = argmax_c(TSN_c * ESCBR_c) \tag{7}$$

With the final values calculated for each level of complexity, the level of complexity that has the highest value is proposed as the final recommendation (Equation 7).

After the complexity-level selection, all the distributions of probability are updated according to the Hawkes process (Equation 4) for each α and β parameter using the constant defined intensity function (Equations 8 and 9) and the excitation function (Equation 10), which generates the dynamic evolution of the Beta probability distributions, thus simulating the forgetting curve.

$$\mu_{\alpha,c}(t) = \begin{cases} 2, & c = 0 \\ 1, & 1 \leq c \leq 4 \end{cases} \tag{8}$$

$$
\mu_{\beta,c}(t) = \begin{cases} 1, & c = 0 \\ 3, & c = 1 \\ 5, & c = 2 \\ 7, & c = 3 \\ 9, & c = 4 \end{cases} \tag{9}
$$

$$
\phi_h(t) = (10)(0.02)e^{-0.02t} \tag{10}
$$

Then, Equation 11 shows the complete definition for all α, and Equation 12 shows the definition for β parameters.

$$
\lambda_\alpha(t) = \mu_{\alpha,c}(t) + \sum_{t_i<t} \phi_h(t - t_i) \tag{11}
$$

$$
\lambda_\beta(t) = \mu_{\beta,c}(t) + \sum_{t_i<t} \phi_h(t - t_i) \tag{12}
$$

Finally, Equation 13 describes the distribution of probability for each complexity level c.

$$
P_c(x, \lambda_\alpha(t), \lambda_\beta(t)) = \frac{x^{\lambda_\alpha(t)}(1-x)^{\lambda_\beta(t)}}{\int_0^1 u^{\lambda_\alpha(t)}(1-u)^{\lambda_\beta(t)}du} \tag{13}
$$

5 Results and Discussion

5.1 Recommender System with a Real-Student Database (TS with Hawkes)

The TS recommender system has been tested with an adapted dataset extracted from real data of student interactions with a virtual learning environment for different courses [14]. The total of learners is 23,366. In this database, there are the learner grades in different courses and multiple evaluation types. For this test, the database format is adapted to the AI-VT structure (grades, response times and complexity levels). The complexity levels are divided into five stages and calculated with the weight percentage defined in the dataset. Figure 2 was generated after 100 executions of the algorithm and shows that despite the stochasticity, the algorithm is stable because the global variance in all the complexity levels is low according to the total number of learners and the total number of recommendations.

The algorithm recommends more low-complexity levels with Hawkes because the knowledge tends to decrease with time. The algorithm force to reinforce the learner knowledge in all complexity levels and since the initial configuration gives a higher probability to the lower levels, the algorithm tends to repeat the more accessible levels needed to reach the higher levels.

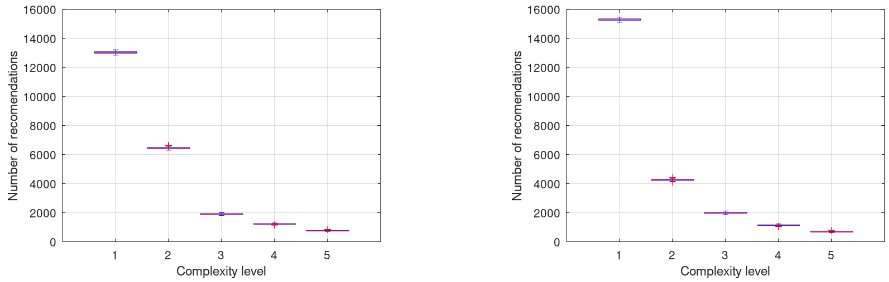

Figure 2: Number of recommendations per complexity level (left: static learning process, right: dynamic learning process with the Hawkes process)

5.2 Simulated Database (ESCBR, TS with Hawkes)

The simulated database is generated using a log-normal distribution of probability to simulate grades of 1,000 learners in five complexity levels, each one with fifteen questions. The generator simulates more complexity by reducing the mean distribution and increasing the variability.

The comparison is according to Equation 14, which calculates the relation between the grades mean x_c and the number of questions y_c for each complexity level c.

$$r(x_c, y_c) = e^{-2(x_c+y_c-1)^2} \tag{14}$$

A specific scenario was defined without initial data (grades and answer times), i.e., a cold start. Table 2 shows the numerical results after 10,000 executions (1,000 learners) for TS and TS-Hawkes in the evaluated scenario. Even with the changes in each complexity level, the total change is only 3.7% in eight questions. Comparative results with others scenarios, a deterministic model and BKT (Bayesian Tracing Model) model was executed and can be consulted in our previous work Soto *et al.* [27].

Table 2: ESCBR-TS and ESCBR-TS-Hawkes Metric comparison

	r_{C0}	r_{C1}	r_{C2}	r_{C3}	r_{C4}	Total	Percent
TS	0.951	0.812	0.675	0.605	0.563	3.606	72.12
TS-Hawkes	0.941	0.718	0.643	0.576	0.545	3.423	68.46

The variance evolution (Figure 3) shows that with the Hawkes process, the values are maintained around the initial configuration, which allows greater adaptability to the dynamic changes in knowledge that occur in the learning process. Since the Beta probability distribution converges rapidly to a single value, as more values are obtained, the variance is smaller. If there is a change in the convergence value, the distribution requires more data to converge to the new value, since the changes in the mean are proportional to the value of

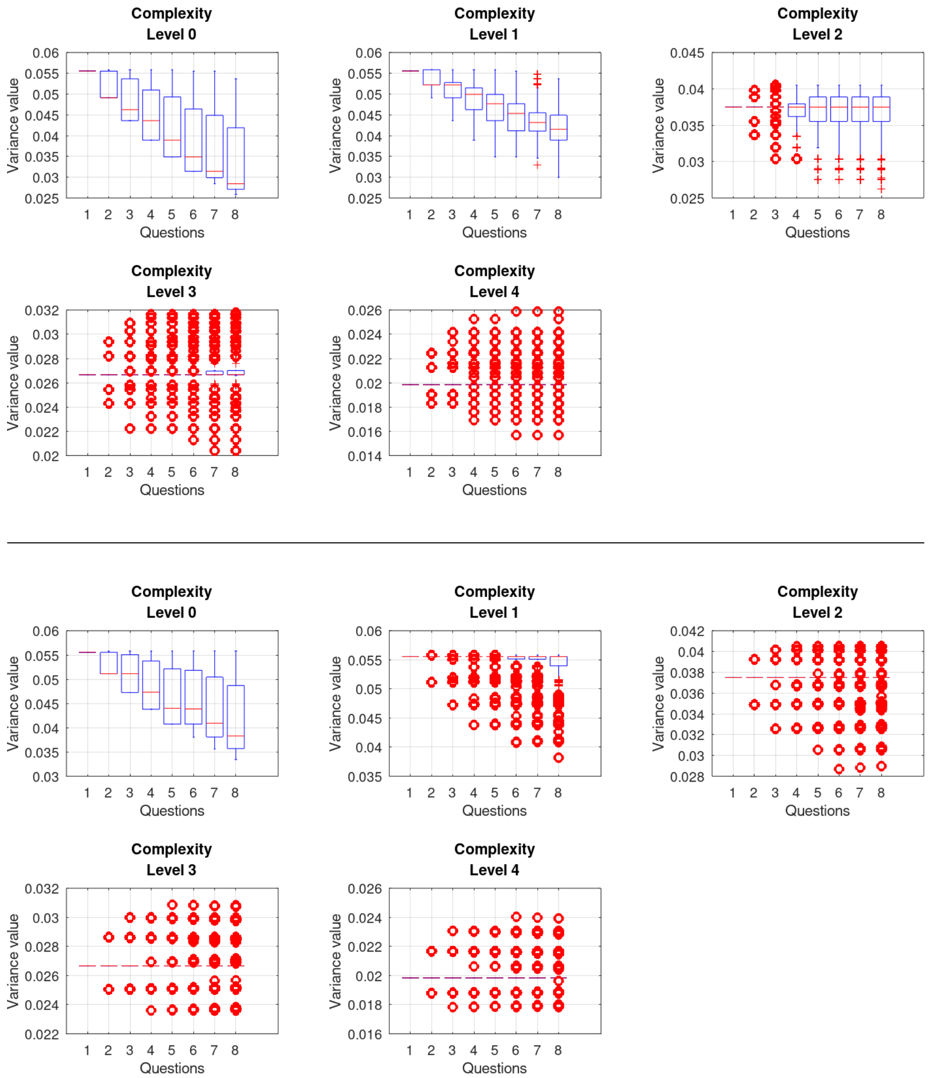

Figure 3: Variance evolution for Beta distribution of probability and all complexity levels (Top: static learning process. Bottom: dynamic learning process with Hawkes process)

the variance with a constant step of change for the parameters. In the case of modeling the learning process, it is preferable to maintain a relatively high variance value to facilitate adaptation to unforeseen changes, that is the main contribution of the Hawkes process to Thompson Sampling for modeling the knowledge evolution.

6 Conclusion

This paper presents an integrated algorithm based on previously developed models: a recommender system based on a Thompson sampling algorithm, an ensemble regression model based on case-based reasoning, and a forgetting curve simulation using the Hawkes process. The integrated algorithm is applied to an ITS called AI-VT. The results show that the integration allows obtaining similar results but with a more realistic process giving the possibility of better personalization of the system and facilitating knowledge acquisition.

The advantages of the proposed model are: i) It allows the generation of personalized recommendations for each learner with relatively little historical data; ii) Since multiple points of view (different algorithms) on the same problem and with the same database are integrated based on Stein's paradox, the risk of falling into Simpson paradoxes is reduced; iii) The two models with the Hawkes process is more realistic and dynamic in the global learning process.

As future work, it is proposed to integrate into the model other variables obtained with complementary artificial intelligence algorithms such as video analysis, audio analysis, and even the analysis of data obtained from learners throughout the learning process. It would also be beneficial to evaluate the learners performance and progression according to proposed recommendations as well as analyze the model with different parametric configurations in order to determine which are the most appropriate configurations and how each variable influences the global behavior of the executed algorithms in the final result.

References

[1] Abel, D., Barreto, A., Van Roy, B., Precup, D., van Hasselt, H.P., Singh, S.: A definition of continual reinforcement learning. In: Oh, A., Naumann, T., Globerson, A., Saenko, K., Hardt, M., Levine, S. (eds.) Advances in Neural Information Processing Systems. vol. 36, pp. 50377–50407. Curran Associates, Inc. (2023)

[2] Ahmed, E.: Student performance prediction using machine learning algorithms. Applied Computational Intelligence and Soft Computing **2024**(1), 4067721 (2024). https://doi.org/https://doi.org/10.1155/2024/4067721, `https://onlinelibrary.wiley.com/doi/abs/10.1155/2024/4067721`

[3] Alrakhawi, H.A., Jamiat, N., Abu-Naser, S.S.: Intelligent tutoring systems in education: A systematic review of usage, tools, effects and evalua-

tion. Journal of Theoretical and Applied Information Technology **2023**(4), 4067721 (2023)

[4] Arnau-González, P., Arevalillo-Herráez, M., Luise, R.A.D., Arnau, D.: A methodological approach to enable natural language interaction in an intelligent tutoring system. Computer Speech and Language **81**, 101516 (2023). https://doi.org/https://doi.org/10.1016/j.csl.2023.101516, https://www.sciencedirect.com/science/article/pii/S0885230823000359

[5] Badier, A., Lefort, M., Lefevre, M.: Comprendre les usages et effets d'un système de recommandations pédagogiques en contexte d'apprentissage non-formel. In: EIAH'23. Brest, France (Jun 2023), https://hal.science/hal-04092828

[6] Badra, F., Lesot, M.J.: Case-based prediction – a survey. International Journal of Approximate Reasoning **158**, 108920 (2023). https://doi.org/https://doi.org/10.1016/j.ijar.2023.108920, https://www.sciencedirect.com/science/article/pii/S0888613X23000440

[7] Chun, S.H., Jang, J.W.: A new trend pattern-matching method of interactive case-based reasoning for stock price predictions. Sustainability **14**(3) (2022). https://doi.org/10.3390/su14031366, https://www.mdpi.com/2071-1050/14/3/1366

[8] Clemente, J., Yago, H., de Pedro-Carracedo, J., Bueno, J.: A proposal for an adaptive recommender system based on competences and ontologies. Expert Systems with Applications **208**, 118171 (2022). https://doi.org/https://doi.org/10.1016/j.eswa.2022.118171, https://www.sciencedirect.com/science/article/pii/S0957417422013392

[9] Desmarais, M.C., Baker, R.S.J.d.: A review of recent advances in learner and skill modeling in intelligent learning environments. User Modeling and User-Adapted Interaction **22**(1), 9–38 (Apr 2012). https://doi.org/10.1007/s11257-011-9106-8, https://doi.org/10.1007/s11257-011-9106-8

[10] Eide, S., Leslie, D.S., Frigessi, A.: Dynamic slate recommendation with gated recurrent units and thompson sampling **36** (2022). https://doi.org/https://doi.org/10.1007/s10618-022-00849-w, https://doi.org/10.1007/s10618-022-00849-w

[11] Ghoorchian, S., Kortukov, E., Maghsudi, S.: Non-stationary linear bandits with dimensionality reduction for large-scale recommender systems. IEEE Open Journal of Signal Processing **5**, 548–558 (2024). https://doi.org/10.1109/OJSP.2024.3386490

[12] Giannikis, S., Frasincar, F., Boekestijn, D.: Reinforcement learning for addressing the cold-user problem in recommender systems. Knowledge-Based Systems **294**, 111752

(2024). https://doi.org/https://doi.org/10.1016/j.knosys.2024.111752, https://www.sciencedirect.com/science/article/pii/S0950705124003873

[13] Iftikhar, A., Ghazanfar, M.A., Ayub, M., Ali Alahmari, S., Qazi, N., Wall, J.: A reinforcement learning recommender system using bi-clustering and markov decision process. Expert Systems with Applications **237**, 121541 (2024). https://doi.org/https://doi.org/10.1016/j.eswa.2023.121541, https://www.sciencedirect.com/science/article/pii/S0957417423020432

[14] Kuzilek, J., Hlosta, M., Zdrahal, Z.: Open university learning analytics dataset. Scientific Data **4**(1), 170171 (Nov 2017). https://doi.org/10.1038/sdata.2017.171, https://doi.org/10.1038/sdata.2017.171

[15] Lamprinakou, S., Gandy, A.: Stratified epidemic model using a latent marked hawkes process. Mathematical Biosciences **375**, 109260 (2024). https://doi.org/https://doi.org/10.1016/j.mbs.2024.109260, https://www.sciencedirect.com/science/article/pii/S0025556424001202

[16] Li, L., Wang, Z.: Knowledge graph-enhanced intelligent tutoring system based on exercise representativeness and informativeness. International Journal of Intelligent Systems **2023**(1), 2578286 (2023). https://doi.org/https://doi.org/10.1155/2023/2578286, https://onlinelibrary.wiley.com/doi/abs/10.1155/2023/2578286

[17] Liu, M., Yu, D.: Towards intelligent e-learning systems. Education and Information Technologies **28**(7), 7845–7876 (Jul 2023). https://doi.org/10.1007/s10639-022-11479-6, https://doi.org/10.1007/s10639-022-11479-6

[18] Louvros, P., Stefanidis, F., Boulougouris, E., Komianos, A., Vassalos, D.: Machine learning and case-based reasoning for real-time onboard prediction of the survivability of ships. Journal of Marine Science and Engineering **11**(5) (2023). https://doi.org/10.3390/jmse11050890, https://www.mdpi.com/2077-1312/11/5/890

[19] Malladi, R.K.: Application of supervised machine learning techniques to forecast the covid-19 u.s. recession and stock market crash. Computational Economics **63**(3), 1021–1045 (Mar 2024). https://doi.org/10.1007/s10614-022-10333-8, https://doi.org/10.1007/s10614-022-10333-8

[20] Mao, K., Dong, Q., Wang, Y., Honga, D.: An exploratory approach to intelligent quiz question recommendation. Procedia Computer Science **207**, 4065–4074 (2022). https://doi.org/https://doi.org/10.1016/j.procs.2022.09.469, https://www.sciencedirect.com/science/article/pii/S1877050922013631, knowledge-Based and Intelligent Information and Engineering Systems: Proceedings of the 26th International Conference KES2022

[21] Nguyen, A.: Dynamic metaheuristic selection via thompson sampling for online optimization. Applied Soft Computing **158**, 111566 (2024). https://doi.org/https://doi.org/10.1016/j.asoc.2024.111566, https://www.sciencedirect.com/science/article/pii/S1568494624003405

[22] Ou, T., Cummings, R., Avella Medina, M.: Thompson sampling itself is differentially private. In: Dasgupta, S., Mandt, S., Li, Y. (eds.) Proceedings of The 27th International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 238, pp. 1576–1584. PMLR (02–04 May 2024), https://proceedings.mlr.press/v238/ou24a.html

[23] Pei, Q., Jia, Z., Liu, J., Wang, Y., Wang, J., Zhang, Y.: Prediction of coal spontaneous combustion hazard grades based on fuzzy clustered case-based reasoning. Fire **7**(4) (2024). https://doi.org/10.3390/fire7040107, https://www.mdpi.com/2571-6255/7/4/107

[24] Seol, Y.: Non-markovian inverse hawkes processes. Mathematics **10**(9) (2022). https://doi.org/10.3390/math10091413, https://www.mdpi.com/2227-7390/10/9/1413

[25] Shah, D.P., Jagtap, N.M., Shah, S.S., Nimkar, A.V.: Spaced repetition for slow learners. In: 2020 IEEE Bombay Section Signature Conference (IBSSC). pp. 146–151 (2020). https://doi.org/10.1109/IBSSC51096.2020.9332189

[26] Soto-Forero, D., Ackermann, S., Betbeder, M.L., Henriet, J.: Automatic real-time adaptation of training session difficulty using rules and reinforcement learning in the ai-vt its. International Journal of Modern Education and Computer Science(IJMECS) **16**, 56–71 (2024). https://doi.org/https://doi.org/10.5815/ijmecs.2024.03.05, https://www.mecs-press.org/ijmecs/ijmecs-v16-n3/v16n3-5.html

[27] Soto-Forero, D., Ackermann, S., Betbeder, M.L., Henriet, J.: The intelligent tutoring system ai-vt with case-based reasoning and real time recommender models. In: Recio-Garcia, J.A., Orozco-del Castillo, M.G., Bridge, D. (eds.) Case-Based Reasoning Research and Development. pp. 191–205. Springer Nature Switzerland, Cham (2024)

[28] Soto-Forero, D., Betbeder, M.L., Henriet, J.: Ensemble stacking case-based reasoning for regression. In: Recio-Garcia, J.A., Orozco-del Castillo, M.G., Bridge, D. (eds.) Case-Based Reasoning Research and Development. pp. 159–174. Springer Nature Switzerland, Cham (2024)

[29] Subha, R., Gayathri, N., Sasireka, S., Sathiyabanu, R., Santhiyaa, B., Varshini, B.: Intelligent tutoring systems using long short-term memory networks and bayesian knowledge tracing. In: 2024 5th International Conference on Mobile Computing and Sustainable Informatics (ICMCSI). vol. 0, pp. 24–29 (2024). https://doi.org/10.1109/ICMCSI61536.2024.00010

[30] Uguina, A.R., Gomez, J.F., Panadero, J., Martínez-Gavara, A., Juan, A.A.: A learnheuristic algorithm based on thompson sampling for the heterogeneous and dynamic team orienteering problem. Mathematics **12**(11) (2024). https://doi.org/10.3390/math12111758, `https://www.mdpi.com/2227-7390/12/11/1758`

[31] Xu, S., Ge, Y., Li, Y., Fu, Z., Chen, X., Zhang, Y.: Causal collaborative filtering. In: Proceedings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval. p. 235–245. ICTIR '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3578337.3605122, `https://doi.org/10.1145/3578337.3605122`

[32] Zaidi, A., Caines, A., Moore, R., Buttery, P., Rice, A.: Adaptive forgetting curves for spaced repetition language learning. In: Bittencourt, I.I., Cukurova, M., Muldner, K., Luckin, R., Millán, E. (eds.) Artificial Intelligence in Education. pp. 358–363. Springer International Publishing, Cham (2020)

[33] Zhang, X., Weng, H., Wei, Y., Wang, D., Chen, J., Liang, T., Yin, Y.: Multivariate hawkes spatio-temporal point process with attention for point of interest recommendation. Neurocomputing **619**, 129161 (2025). https://doi.org/https://doi.org/10.1016/j.neucom.2024.129161, `https://www.sciencedirect.com/science/article/pii/S0925231224019325`

[34] Zhu, Z., Van Roy, B.: Scalable neural contextual bandit for recommender systems. In: Proceedings of the 32nd ACM International Conference on Information and Knowledge Management. p. 3636–3646. CIKM '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3583780.3615048, `https://doi.org/10.1145/3583780.3615048`