International Journal of Web-Based Learning and Teaching Technologies
Volume 17 • Issue 7

Multiagent Intelligent Tutoring System for Financial Literacy

Rafael Marin, Mackenzie Presbyterian University, Brazil

Pollyana Notargiacomo, Mackenzie Presbyterian University, Brazil

ABSTRACT

Financial literacy is a theme that integrates public policies for the social development of a country and an element to be worked on from different aspects to improve people's living standards, providing well-being. In this context, Stima is proposed, a system capable of acquiring knowledge from experts and tutors to help students to follow the bases of financial planning and decision-making. It aims to relate different approaches to artificial intelligence and institute a language that, through syntactic, lexical, and semantic analyses, executed by different agents within the model, makes it possible to define financial profiles and recommend financial planning. The knowledge stored is used to propose financial monitoring standards and provides tools to assist financial decision making. A set of eight profiles, with three indicators each, was configured by experts inside a prototype, and a volunteer student was accompanied for four months giving validations to the system.

KEYWORDS

Association Rule Mining, Bayesian Networks, Decision Support System, Knowledge Representation Language, Knowledge-Based Systems, Text Mining

INTRODUCTION

According to the Organisation for Economic Co-operation and Development, financial literacy is the process by which individuals seeks their well-being by acquiring knowledge in personal finance and through the use of assimilated learning on ways to deal with money and financial products, improve their standard of living and the society to which they belong (OECD, 2005). This study is in line with a worldwide effort to apply financial literacy (Mitchell and Lusardi, 2011). The main research problem explored is how to systematize the financial planning professional's knowledge to consider the specific profiles and respect the needs of each individual. These professionals are expensive and could not be accessible by all peoples in a country.

In summary, financial literacy is composed of eight areas of knowledge, which can be divided into basic and advanced, as shown in figure 1. The first step should be to search for motivation and basic numeracy skills, and then, it seeks to structure a budget plan to guide the proposed goals in the project that motivated the beginning of learning. Also, new consumption habits are needed when the budget planning is started and constantly monitored, mainly for debt management and the construction of an emergency reserve. With these steps properly executed, the advanced steps are knowledge related to risk management, investments, and retirement (Lusardi & Mitchel 2010; Seeber & Retzmann, 2016; Marin & Notargiacomo, 2019).

DOI: 10.4018/IJWLTT.288035

Figure 1. Financial literacy training structure (Marin & Notargiacomo, 2019)

Lusardi and Mitchel (2010) determine that, for the first area of knowledge (self-motivation), there are two basics motivations for an individual to seek financial literacy: the accumulation of wealth to satisfy a desire or need, such as the realization of a project, and planning for retirement. Both are salutary given that the first reason reaches the achievement of self-esteem, the top of Maslow's motivational pyramid (Maslow, 1943) when all individual needs are properly satisfied, and the fullness of personal well-being is reached. The second reason is comparable in importance, given that the lack of funds in elderly life makes the individual more vulnerable to the adversities caused by old age, such as health problems and reduced mobility, as well as more dependent on public policies to supply such personal deficit (Rooij et al., 2011).

A study developed by Drexler, Fischer, and Schoar (2014) supports these areas in a summary of financial literacy training programs. It starts with the Savings area and understands the motivations to start it (Self-Motivation), setting saving goals, and continuing with Consumption, Debt Management, Account Separation (budget Planning), and Estimation Methods. These studies reinforce that the effectiveness of this training process depends on the constant efforts of students in monitoring their accounts budget and expenses.

This article contribution is to present a system architecture and prototype capable of integrating artificial intelligence tools to consolidate this training pattern through the acquisition of expert's knowledge, also to reduce the amount of effort that finance students have to expend to maintain financial control in their lives and, be an independent solution capable of unifying the users' financial data sources. The next section explores the related concepts and works relevant to this proposal. After that, it has presented the architecture of the STIMA system (an Italian word for esteem and estimation), a multiagent intelligent tutoring system for basic financial literacy training and monitoring, its structures of agents, and then presenting the system used for the inference engine that uses an interpretation of its self-language to list the nodes registered by experts. A technical test was performed with a database previously labeled to validate the system's accuracy and performance, and then a test user was followed by a tutor within 4 months to infield validations, and finally, the test results and possible improvements of the system are discussed in the conclusions.

Literature Review And Related Works

As seen, all areas depend on budget planning to be executed correctly and with constant monitoring(Seeber & Retzmann,2016), and knowing that there is software for this purpose, it was sought to know in literature, which software is directly linked to financial literacy using artificial intelligence techniques, and a mapping study with this end is commented as follows (Marin & Notargiacomo,2019). In the study, was found 81 articles and after a selection filter, 18 were related to the subject. These solutions were analyzed to fit into five classifications that defined their software approach, platform, artificial intelligence techniques, gamification techniques, and target audience. In the software approach, the content presentation format was evaluated, whether in application format (App), requesting information and using the data to present results, or game format (Game), presenting content for users involving a plot and capturing user data through gamified structures. In terms of platform, the software was evaluated for deployment technology, being divided into Web, Mobile, and desktop. In Artificial Intelligence Techniques, solutions using prediction or classification algorithms were verified, as well as other solutions related to the area. The item Gamification Techniques sought to classify solutions that used game elements, such as leaderboards, levels, or scores in the context of applications to apply concepts and content related to financial education. As conclusions in this study, it was found that the software developed for financial literacy are, in the majority (58%) applications with some specific functionality, such as making available reading material on the topic or even to accompany the daily spending routine, to be used in a mobile platform (53%) or web (29%) with a target audience focused on teenagers and adults (84%).

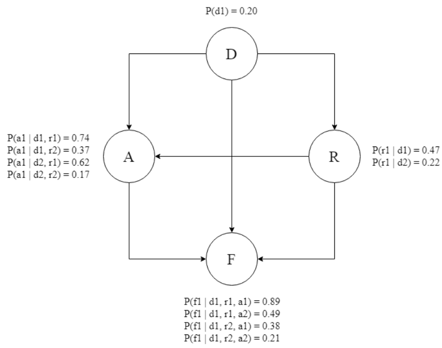
A study developed by Bahrammirzaee (2010) evaluated 278 articles on artificial intelligence uses in finance. From these, fourteen solutions are personal finance planning systems and combine technologies on knowledge-based systems and multicriteria decision-making support and are used by banks or financial institutions to propose products to their clients, such as savings plans or stock market investments according to their profiles (Bahrammirzaee, 2010, p. 1175-1176; Murugesan & Manohar, 2019). Another study conducted by Waliszewski and Warchlewska (2020) evaluates the peoples' behavior on artificial intelligence applications for personal financial planning in Europe, not evaluating a specific software, but the general behavior, and concluded that the percentage of respondents indicating their satisfaction with using a software to analyze consumption habits was higher than the use of software to make investment decisions.

Knowledge-based systems are those capable of assisting decision-making within a specific domain area using the knowledge of experts saved in the database (Zagoruiko, 1988; Do et al., 2018; Caroleo et al., 2018). This is an area of artificial intelligence that contains Expert Systems and Intelligent Tutoring Systems topics, among others. There are four main structures inside in common, the Inference Engine, knowledge database, working memory, and the profile database (Liao, 2005).

The inference engine translates the information contained inside the knowledge base to assist in decision making; the working memory is the area of expert systems that combines the rules evaluated by the inference engine with user data added to the profile database (Clancey, 1984; Blum, 1988; Hadi, 2011; Arevalillo-Herráez, 2013). Intelligent tutoring systems (ITS) follow the same structure, adding tools for supporting the Tutor, a system actor responsible for adding the pedagogical model, to bring the content closer to the individual needs of each student. Tutorial strategies should bridge the gap between expert and student knowledge (Alkhatlan and Kalita,2019; Jonassen and Wang, 1993; Nkambou et al., 2010). This relationship is observed in MYCIN (Shortliffe, 1976; Buchanan and Shortliffe, 1984), an expert system for supporting medical diagnostics for NEOMYCIN (Clancey and Letsinger, 1981), a system for teaching diagnostic strategies.

For the first architecture goal, the inference engine presented in the next section makes use of the architecture of Bayesian networks presented by Neapolitan (Neapolitan, 2003, p. 4). These are models that allow relating variables in a probabilistic and statistical way, synthesizing human knowledge from several areas in expert systems (Russell & Norvig, 2010, p. 518). In the form of graphs, topology places each variable in nodes, and the edges give the relationship between these nodes in a directed

Figure 2. Bayesian network based on Neapolitan (2004, p.4)



way. An example is used for medical diagnostics, as shown in figure 2 and table 1, but the concept is adapted and used in financial literacy.

In the example, a network is constructed with some symptoms to define the relationship between diabetes, renal damage, anemia, and fatigue. Nodes represent the diseases, and variables configure their representative factor in the model (Cobb and Shenoy, 2005). To exemplify this, the node *F*(Fatigue) have three variables anemia (*a*) and diabetes (*d*) and renal damage (*r*), with two possible states each, present (*a1, d1, r1*) or absent (*a2, d2, r2*), these variables are related by probability notation P(*f1*| *a1,d1,r1*). It also could be converted from the mathematical model to the form of a truth table (Russell and Norvig, 2010, p. 519).

In Stima System, a multicriteria decision-making method relates these nodes and variables, helping the tutor actor select students' profiles according to the answers given to indicators. According to Triantaphyllou (2000), there are three steps in common to all these methods, which are:

1. Determine relevant criteria and alternatives.
2. Attach numerical measures of importance for each criterion
3. Process the criterion rank according to alternatives.

This process integrates the structure of deterministic Bayesian networks, commented on previously, using the weighted Sum Model (WSM) according to equation 1 as follow (Fishburn,1967; Triantaphyllou, 2000):

Table 1. Variables of the Bayesian network as treated by Neapolitan (2004)

Feature	Value	Description
D	d1	Diabetes Present
	d2	Diabetes Absent
R	r1	Renal Damage Present
	r2	Renal Damage Absent
A	a1	Anemia Present
	a2	Anemia Absent
F	f1	Fatigue Present
	f2	Fatigue Absent

$$A^{*}_{WSM-score} = max_i \sum_{j=1}^{n} a_{ij} w_j, \; for \;\; i = 1,2,3,\ldots,m \tag{1}$$

Given a set that contains the decision criteria (*n*), $A^*_{WSM-score}$ is the sum score of each criterion (*j*), that contains a subset of alternatives (*m*), where each alternative (a_{ij}) weighted according to criterion weight (w_j) (Boza et al., 2017). Affinity is the name given to this score in Stima System. Profiles are the criteria, and alternatives are the indicators, and the profile weights change according to exclusive indicators. These structures and their uses are detailed in Stima's Architecture.

The architecture used connects these functionalities and actors through software agents. Russell and Norvig (2010, p. 42) defined softbots (software robots) as software with the ability to interact in the environment using sensors to receive information data and, after processing, return the result via actuators. They also highlight a way of describing the agents separately, called the PEAS description. PEAS is an acronym for Performance, Environment, Actors, and Sensors and describes the agent functions. For example, in the agent responsible for interpreting the language developed, the Performance Index is accuracy in searching for profiles according to students' responses, the Environment is Inference engine, the Actuators are Affinity, a measure based on weighted Sum Model (WSM) and Sensors are the Indicators responses. Therefore, the concept of multiagent systems is in the interaction of several agents within a common infrastructure that allows communication through previously defined protocols, where each agent has a limited part of the information (Wooldridge, 2009).

For the second and third architecture goals (to reduce the amount of effort to maintain financial control in their lives and be an independent solution capable of unifying the users' financial data sources), the two more difficult tasks related to financial literacy students are to annotate expenses and incomes and to find the correct classification to this accounts (Drexler, Fischer & Schoar, 2014). Two minor agents work together in a symbiotic way and use text mining techniques and confidence-based classification (aPriori method).

Text mining is the analysis of a corpus for pattern discovery (Chen & Kan, 2013). The process is performed with initial filtering of the data, removing the articles and prepositions, then transforming the data is necessary to remove the gender and the gerund (Alzahrani and Ghorbani, 2016). After this initial treatment, several tokens are tested within the corpus to find relevant information such as dates and times of movement, value, and origin of operation (Duan et al., 2016). After this process, a final treatment is done to infer missing data necessary for the model, such as year of occurrence on dates that have only day and month and other information. An association rule mining agent is used to encounter relationships between the account found in SMS mining with a profile account registered in the system. The mining of association rules is used to propose rules and quantify their relevance concerning the student, with an aPriori algorithm (Agrawal et al., 1993; Han et al., 2012; Shatnawi et al., 2020), using its two units of measure, support and confidence. The support of an association rule based on antecedents and consequences $\left(\mathrm{A} \rightarrow \mathrm{C}\right)$ indicates the probability of this relationship found in the data set evaluated using equation 2 as follow:

$$Sup\left(A \rightarrow C\right) = P\left(A \cup C\right) = \frac{\sigma\left(A \cup C\right)}{n} \tag{2}$$

In the equation, the count of occurrences of the relation $\left(\sigma\left(A \cup C\right)\right)$ divided by the total transactions of the evaluated data set (*n*), this unit of measurement serves to perform the first pruning of the model, and all rules that are below a previously used indicator, this method performs the association with performance in large databases.

The confidence counts the occurrence of the consequent part of the rule about the antecedent, determining the degree of relationship between the items based on equation 3 as follows:

$$Conf\left(A \rightarrow C\right) = P(C \mid A) = \frac{\sigma\left(A \bigcup C\right)}{\sigma\left(A\right)} \tag{3}$$

The confidence is the count of the antecedent about the consequent $\sigma\left(A \cup C\right)$ divided by the total of antecedents $\sigma\left(A\right)$. This unit performs the second pruning of the model, where all records found below a previously determined factor are cut. With this, the system selects the most confident account for the profile.

These functionalities are integrated into the system to make it possible to monitor the finances of students in a multiagent architecture integrating experts, tutors, and students. These structures are used in the construction of a prototype demonstrated in the next, and the results are discussed in section Conclusion.

The Architecture of the Stima System

The system architecture demonstrates the unification of the discussed concepts and the functions performed in the interaction between agents to enable the system's communication on multiple platforms. For this, the mediating agents are represented in figure 3. These agents coordinate the actions of minor agents responsible for performing the processing of specific system actions, as follows:

As could be seen in figure 3, three actors are expected to use the system, with different roles and responsibilities. The expert determines the Bayesian network nodes, and these nodes are treated here as Profiles, which controls the specific standards for each financial problem. If compared to the example in the medical field, each profile represents a disease to be diagnosed. Each of these profiles is related to variables, which are called in the system as indicators. In medical example, indicators are the symptoms demonstrated by the diseases. These indicators are added in a question format, and it is possible to diagnose the student according to their responses. The relationships are stored within the knowledge base as a third structure called Profile Indicators, which contain the notations in a system language to relate the profiles to the indicators and the weight of this relationship, as shown in figure 4.

Another role of the expert actor is to determine a possible treatment for the diagnosed financial disease. It is a chart of accounts with delimited percentages that teaches the student how to use money

Figure 3. Stima's multiagent architecture

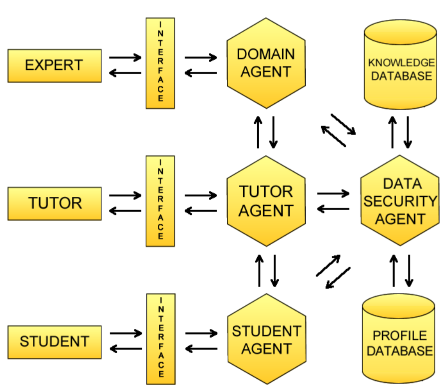
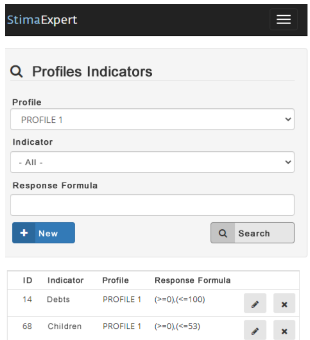


Figure 4. Profile indicators configuration in Expert agent



according to the profile. This knowledge is stored by the domain agent within the knowledge base to be used later by the tutor agent, making the responsibility of the inference engine to be distributed among the agents asynchronously.

The STIMA language, used for knowledge representation and storage, follows three basic steps common to programming languages: syntax analysis, lexical analysis, and semantic analysis (Sciore, 2020, Andujar et al., 2020). In the first step, the parser present in the domain agent validates whether the command is written in the correct sequence, evaluating if the language tokens are properly positioned so that the lexical analyzer could decompose it, as could be seen in Table 3. Because it is a language delimited by parenthesis, the parser validates if the command is properly started. If so, it evaluates the depth of the conditional and logical operators. It also assesses whether all open structures are properly closed and whether the variables used exist and have a value. An indicator can be compared to numeric values, text values (between quotes), Boolean values, or other indicators (in brackets). The equation represents a declaration made in the correct syntax, which includes comparison operators and logical operators AND (,) and OR (;), in the example equation 4 as follows.

$$(>0),\left(\left(!=1\right);(<10)\right) \tag{4}$$

For the statement exemplified, the lexical analyzer will decompose the terms for a memory area, as can be seen in Table 2.

In the tutor agent, the semantic analyzer uses the structure presented in table 3 to evaluate each Level of the declaration according to the value defined for the indicator by the student, starting by solving the highest level comparisons operations (*compLevel*) and executing them with logic operators (*logicOp*)of the same Level (*logicOp Level*) until all are leveled at level 1, and finally, only one value (TRUE or FALSE) is defined for the operation. A minor tutor agent tests students' indicators responses for all registered profiles, using semantic analyzer results, always starting with exclusive indicators, that are rules capable of zero the weights of all profile indicators and the Affinity of the

Table 2. Representation of memory area used for the lexical analyzer

Comp	value	compLevel	logicOp	logicOpLevel
>	0	1	,	1
!=	1	2	;	2
<	10	2		

tested profile if a profile indicator condition is not attended, in other words, if an exclusive indicator is marked as false, all the rules of a profile stop being tested and the profile is excluded from that student's possibilities, at the end of the tests an affinity value is defined to the profiles.

The Affinity score in Stima System follows the same equation as the WSM score (Triantaphyllou, 2000; Fishburn, 1967), and the weight of the indicator could be customized according to profile or changed to zero if some exclusive indicator is not attended. According to the responses given, the profile with the greatest Affinity is defined to the student. If more than one profile is evaluated with the same score for the student, the tutor actor must choose from those selected by the system. According to the profile defined, the tutor agent selected a chart of accounts with specific limits for the student's budget planning. If it is used the medical diagnosis example mentioned earlier, the chart of accounts with specific limits would be the recommended treatment. With the diagnosis done, the Tutor actor is also responsible for the functions that allow account customization for each student separately.

The student actor is responsible for answering the indicators proposed by experts, which are used to determine the profile with the tools described so far. With this personal data, the Student Agent has tools for monitoring the use of the chart of accounts developed for the proposed financial profiles. It is composed of two minor agents and a report. The first minor agent is a mobile miner capable of extracting information from SMS messages from banks and uploading it to the student profile, using text mining techniques, as well as registering income and expenses manually. The second agent is a classifier that organizes this information inside an account determined to the profile according to the Confidence index. Moreover, the Advisor, a report that shows the usage percentage status of each account, determined previously by experts and tutors, and helps the user to make the financial decision upon these indicators.

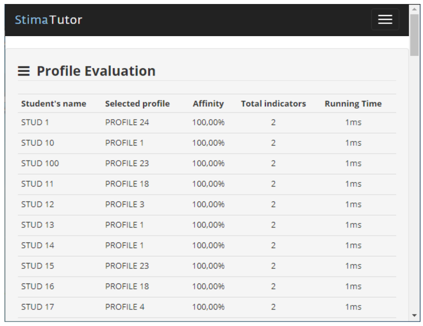
Due to personal and financial data storage, the Data Security Agent adds a security level for accessing knowledge base and profile data, negotiating tokens for each access, and encrypting sensitive information. A prototype was developed to test this system's model. It uses the CSharp programming language with the MVC design pattern to develop student question and answers agents and the expert and tutor agents. This application was hosted on a dedicated cloud server, with an Intel Xeon processor CPU D-1541 with 2.10GHz, 16 GB DDR4 2667 MHz, 2 TB HDD (RAID level 1), Microsoft Windows Server 2016 Standard R2, and Microsoft SQL Server 2019 Database. The student account monitoring tools were developed in react-native to be used in mobile platforms. Each agent was developed as an independent application, and the conversation between them is realized by exchanging JSON packages in a previously defined pattern. The actors (experts, tutors, and students) perform the login through specifics interfaces. However, the same user can have more than one role in the system.

The technical tests consisted of assessing the accuracy of the inference engine. The network was started with 50 nodes (profiles), two variables per node (indicators per profile), totalizing 100 variables, 100 students, and two responses per student, totalizing 200 tested responses. For the test, the indicators number of children and debt value were controlled so that each student had only one selected profile (accuracy of 100%), as can be seen in figure 5. In performance tests, 20 random markings were made among the various tests, and there were no fluctuations greater than 16 ms for the total processing time, the minimum time is 96 ms, and the maximum is 112 ms for this mass of data.

After these technical tests, standard profiles were developed to be inserted as an initial system configuration by professional CFPs (certified financial planners). And from his recommendations was possible to create basics profiles, indicators, and chart of accounts as follows in table 3.

The profile indicators were marked as exclusive, and their weights equal to 1 (neutral), and each of the eight profiles is unique, and they are mutually exclusive. Thus, regardless of the responses given by students to the indicators, only one profile is selected. Therefore, the indicators were registered in a format of 3 questions to be answered by the students, marital status, number of dependents on the household, and the current value of debts.

Figure 5. Profile selection tests


StimaTutor

Profile Evaluation

Student's name	Selected profile	Affinity	Total indicators	Running Time
STUD 1	PROFILE 24	100,00%	2	1ms
STUD 10	PROFILE 1	100,00%	2	1ms
STUD 100	PROFILE 23	100,00%	2	1ms
STUD 11	PROFILE 18	100,00%	2	1ms
STUD 12	PROFILE 3	100,00%	2	1ms
STUD 13	PROFILE 1	100,00%	2	1ms
STUD 14	PROFILE 1	100,00%	2	1ms
STUD 15	PROFILE 23	100,00%	2	1ms
STUD 16	PROFILE 18	100,00%	2	1ms
STUD 17	PROFILE 4	100,00%	2	1ms

Table 3. Profiles and indicators configuration

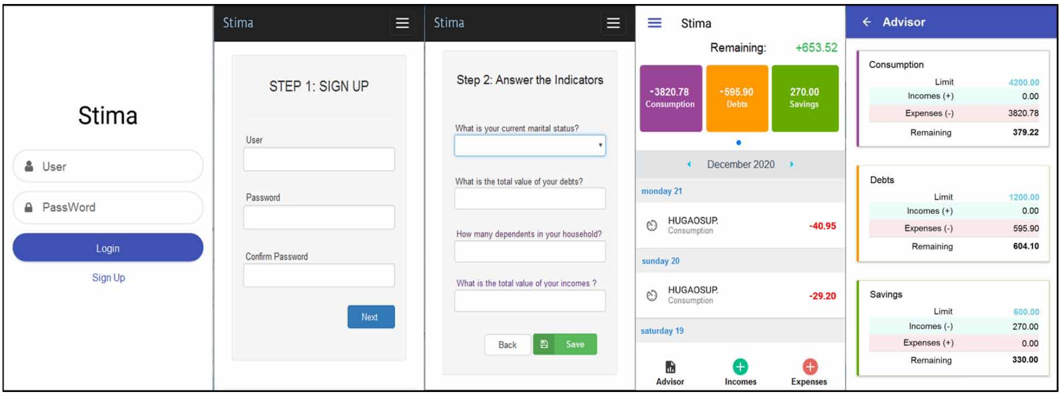
	Indicators			Chart of Accounts (% of incomes)		
Profiles	**Marital Status**	**Dependents**	**Debts**	**Consumption**	**Debts**	**Savings**
P1	Married	>0	>0	70	20	10
P2	Married	>0	==0	80	0	20
P3	Married	==0	>0	60	20	20
P4	Married	==0	==0	60	0	40
P5	Single	>0	>0	70	20	10
P6	Single	>0	==0	80	0	20
P7	Single	==0	>0	60	20	20
P8	Single	==0	==0	50	0	50

A volunteer user was accompanied by a tutor for a period of four months to test the configurations. The user answered that he is married, having two dependents and debts above US$ 10,000.00 (ten thousand US dollars), and because he is self-employed, the incomes are variable. The Affinity selected by the system, according to the answers, is to the profile P1.

The accounts and percentages proposed by the experts were kept for the student. The total forecast for the consumption account (seventy percent of the incomes) was exceeded at the end of the first month. However, with the use of the student agent's tools, as could be seen in figure 6, it was possible to identify several expenses that could be cut so that the percentage would not exceed again in the next periods. Regarding the value of debts related to real estate financing, these did not exceed the expected total (twenty percent of the incomes). Due to excess spending on the consumption account, savings were not made in this first testing period.

In the Student agent, the user initiates to signing up to the system choosing a user name and password and answering the indicators inserted by the experts in a web interface. According to the answers, the system automatically defines a profile and a chart of accounts with a limited income percentage (table 3). The text mining agent runs in smartphones to capture banking communications and add the expenses automatically in control, the association rule agent uses this information to classify the expense with the most confident account, according to the selected previously by users to the same expense, and users could change the automatic selection. The advisor report shows how the accounts progress is going in the month selected, showing how much is remaining to complete its goals.

Figure 6. Student agent (login, sign up, answers to indicators, account management and advisor report)



In the first month of student monitoring, it was possible to identify market expenses on practically every day of the month, which demonstrated a lack of control in relation to his pantry, which was easily resolved on his own, since the student himself institutes a monthly market list, which reduced his monthly expenses in the consumption account by 19%, another expense also controlled on his own was the expenses with restaurants, where it was instituted that he should go to restaurants only once a month as a commemoration of the realization of the emergency reserve. The student started an emergency reserve in the second month as recommended by the Advisor, and it continues in the other months as a habit.

Savings was a point that divided the opinions of experts and tutors during the development of the chart of accounts, as some commented that the amount related to rates paid on debts was greater than the income offered by savings. The argument that maintained the current scenario has a psychological nature and says that the growth of a personal reserve motivates the continuity of the plan due to the sense of security provided by an emergency reserve (Jappelli & Padula, 2013; Babiarz & Robb, 2014). From the second month of follow-up, the tested user was able to reduce the use of his consumption account so as not to exceed the proposed percentage, and with that, he started the savings deposit. And this saving motivation effect began to be reported by the user after the fourth month of using the system, where reserves reached thirty percent of normal revenues.

Conclusions and Future Works

From the search for the foundations of financial education, a form to systematize experts' knowledge and assist students in changing consumption habits using artificial intelligence techniques.

For this, the concept of deterministic Bayesian networks was converted from the mathematical model (relational graphs) to the computational model using a specific language developed to relate the nodes (profiles) to the variables (indicators), thus enabling the acquisition and storage of knowledge.

The strength of these relationships was calculated using a measurement unit called Affinity, which uses the Weighted Sum Method, a multicriteria decision-making model, to calculate the weights by alternatives, thus allowing each indicator to be tested with different weights according to the treated profile. The concept of exclusive indicators was also used, which makes it possible to reset the weights of all indicators to zero if an exclusive profile indicator condition is not attended.

The developed prototype, and its respective test, demonstrated that the language created for knowledge acquisition of the system (storage, retrieval, and execution) allows it to be performed by different agents in a decentralized way, also allowing for asynchronous execution.

The system's architecture is structured to be used by several areas of knowledge, such as finance or medicine for example, as long as it allows the expert to have control of all levels and relationships, being able to add profiles, indicators, and relationships in it self-language, storing knowledge in the database for later use as demonstrated. As it is a multiagent system, it is possible to communicate any agent with the results of an inference engine, increasing the range of possible uses, and once the profile is correctly found, the output can be used to propose a chart of accounts for a specific financial situation, as well as to propose a treatment for the symptoms of diseases, which demonstrates the versatility of the model.

For testing, an initial configuration was added by experts in the system, and a tutor accompanied a volunteer student for four months. With this, it was possible to validate the functioning of the student agent that automatically captured all the student's financial transactions through bank SMS mining and proposed classifications for each expenditure incurred, thus making it possible to make changes in consumption habits throughout the test and it was possible to initiate an emergency reserve.

As future works, it is intended to add training to the text mining algorithm to cover more banks and use other mining sources, such as APIs and push notifications from banking applications, and also publish the entire solution for testing with a larger number of students.

REFERENCES

Agrawal, R., Imielinski, T., & Swami, A. (1993). Mining association rules between sets of items in large databases. *Sigmod conference. SIGMOD Record*, *22*(2), 207–216. doi:10.1145/170036.170072

Alkhatlan, A., & Kalita, J. (2019). Intelligent tutoring systems: A comprehensive historical survey with recent developments. *International Journal of Computers and Applications*, *181*(43), 1–20. doi:10.5120/ijca2019918451

Alzahrani, A. J., & Ghorbani, A. A. (2016). Sms-based mobile botnet detection module. *Sixth International Conference on IT Convergence and Security (ICITCS)*, 1–7.

Andujar, C., Vijulie, C. R., Vinacua, A., Santos, B. S., & Alford, G. (2020). Syntactic and semantic analysis for extended feedback on computer graphics assignments. *IEEE Computer Graphics and Applications*, *40*(3), 105–111. doi:10.1109/MCG.2020.2981786 PMID:32356732

Arevalillo-Herráez, M., Arnau, D., & Marco-Giménez, L. (2013). Domain-specific knowledge representation and inference engine for an intelligent tutoring system. *Knowledge-Based Systems*, *49*, 97–105. doi:10.1016/j.knosys.2013.04.017

Babiarz, P., & Robb, C. (2014). Financial literacy and emergency saving. *Journal of Family and Economic Issues*, *35*(1), 40–50. doi:10.1007/s10834-013-9369-9

Bahrammirzaee, A. (2010). A Comparative Survey of Artificial Intelligence Applications in Finance: Artificial Neural Networks, Expert System and Hybrid Intelligent Systems. *International Journal of Neural Computing and Application*.

Blum, B. I. (1988). A simple expert system. *SIGBIO Newsl.*, *10*(1), 22–28. doi:10.1145/43965.43967

Boza, M., Zizovic, M., Petojevic, A., & Damljanovic, N. (2017). New weighted sum model. *Filomat*, *31*(10), 2991–2998. doi:10.2298/FIL1710991M

Buchanan, B., & Shortliffe, E. (1984). *Rule-based Expert System – The MYCIN Experiments of the Stanford Heuristic Programming Project*. Academic Press.

Caroleo, B., Palumbo, E., Osella, M., Lotito, A., Rizzo, G., Ferro, E., Attanasio, A., Chiusano, S., Zuccaro, G., Leone, M., & De Gregorio, D. (2018). A knowledge-based multicriteria decision support system encompassing cascading effects for disaster management. *International Journal of Information Technology & Decision Making*, *17*(05), 1469–1498. doi:10.1142/S021962201850030X

Chen, T., & Kan, M.-Y. (2013). Creating a live, public short message service corpus: The nus SMS corpus. *Language Resources and Evaluation*, *47*, 299–335.

Clancey, W. (1984). Knowledge acquisition for classification expert systems. *Proceedings of the 1984 annual conference of the ACM on the fifth-generation challenge*, 11–14. doi:10.1145/800171.809597

Clancey, W., & Letsinger, R. (1981). Neomycin: Reconfiguring a rule-based expert system for application to teaching. *Proceedings of the Seventh International Conference on Artificial Intelligence*, 2, 829–836.

Cobb, B., & Shenoy, P. (2005). Hybrid Bayesian networks with linear deterministic variables. *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence*.

Do, N. V., Nguyen, H. D., & Selamat, A. (2018). Knowledge-based model of expert systems using rela-model. *International Journal of Software Engineering and Knowledge Engineering*, *28*(08), 1047–1090. doi:10.1142/S0218194018500304

Drexler, A., Fischer, G., & Schoar, A. (n.d.). Keeping it simple: Financial Literacy and Rules of Thumb. *American Economic Journal. Applied Economics*, *2*, 1–31.

Duan, L., Li, N., & Huang, L. (2016). *A new spam short message classification* (vol. 2). Academic Press.

Fishburn, P. C. (1967). Additive utilities with incomplete product sets: Application to priorities and assignments. *Operations Research*, *15*(3), 537–542. doi:10.1287/opre.15.3.537

Hadi, A. S. (2011). Article. *Expert Systems: International Journal of Knowledge Engineering and Neural Networks*, 480–482.

Han, J., Kamber, M., & Pei, J. (2012). *Data mining: Concepts and techniques* (3rd ed.). Morgan Kaufmann.

Jappelli, T., & Padula, M. (2013). Investment in financial literacy and saving decisions. *Journal of Banking & Finance*, *37*(8), 2779–2792. doi:10.1016/j.jbankfin.2013.03.019

Jonassen, D. H., & Wang, S. (1993). The physics tutor: Integrating hypertext and expert systems. *Journal of Educational Technology Systems*, *22*(1), 19–28. doi:10.2190/7L7F-57H5-APYU-AAPA

Liao, S. (2005). Expert system methodologies and applications - a decade review from 1995 to 2004. *Expert Systems with Applications*, *28*(1), 93–103. doi:10.1016/j.eswa.2004.08.003

Lusardi, A. & Mitchell, O. (2010). *The outlook for financial literacy. Financial Literacy: Implications for Retirement Security and the Financial Marketplace*. Academic Press.

Marin, R., & Notargiacomo, P. (2019). Software solutions for financial literacy: A mapping study. *Computers & Education*, *1*, 16–27.

Maslow, A. (1943). A theory of human motivation. *Psychological Review*, 50.

Mitchell, O., & Lusardi, A. (2011). Financial literacy around the world: An overview. *Journal of Pension Economics and Finance*, *10*(4), 497–508. doi:10.1017/S1474747211000448 PMID:28553190

Murugesan, R., & Manohar, V. (2019). AI in Financial Sector – A Driver to Financial Literacy. *Shanlax International Journal of Commerce*, *7*(3), 66–70. doi:10.34293/commerce.v7i3.477

Neapolitan, R. (2003). *Learning Bayesian networks*. Pearson.

Nkambou, R., Bourdeau, J., & Mizoguchi, R. (2010). Introduction: What are intelligent tutoring systems, and why this book? *Studies in Computational Intelligence*, *308*.

OECD. (2005). *Recommendation on principles and good practices for financial education and awareness*. OECD.

Rooij, M., Lusardi, A., & Alessie, R. (2011). Financial literacy, retirement planning, and household wealth. *Economic Journal (London)*, *122*(560), 449–478. doi:10.1111/j.1468-0297.2012.02501.x

Russell, S., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (3rd ed.). Pearson.

Sciore, E. (2020). Parsing. In Database design and implementation. Data-Centric Systems and Applications (vol. 1). Academic Press.

Seeber, G., & Retzmann, T. (2016). *Financial Education in general education schools: A competence model*. International Handbook of Financial Literacy.

Shatnawi, R., Althebyan, Q., Ghaleb, B., & Al-Maolegi, M. (2020). A Student Advising System using Association Rule Mining. *International Journal of Web-Based Learning and Teaching Technologies*, *16*.

Shortliffe, E. (1976). Computer-based medical consultations: Mycin. *Artificial Intelligence*, *388*.

Triantaphyllou, E. (2000). Multicriteria decision-making methods. Multi-criteria Decision-Making Methods: A Comparative Study, 5–21.

Waliszewski, K., & Warchlewska, A. (2020). Attitudes towards artificial intelligence in the area of personal financial planning: A case study of selected countries. *Entrepreneurship and Sustainability Issues*, *8*(2), 399–420. doi:10.9770/jesi.2020.8.2(24)

Wooldridge, M. J. (2009). *An introduction to multiagent systems*. John Wiley & Sons.

Zagoruiko, N. (1988). Expert systems and pattern recognition. *International Journal of Pattern Recognition and Artificial Intelligence*, *3*.