Indonesian Journal of Electrical Engineering and Computer Science
Vol. 40, No. 2, November 2025, pp. 860~870
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v40.i2.pp860-870

Generalized domain tutoring framework for AI agents with integrated explainable AI techniques

László Csépányi-Fürjes, László Kovács
Institute of Information Science, Faculty of Mechanical Engineering and Informatics, University of Miskolc, Miskolc, Hungary

Article Info

Article history:

Received Sep 25, 2024
Revised Jul 17, 2025
Accepted Oct 14, 2025

Keywords:

Architectures for educational technology system
Artificial intelligence
Explainable AI
Intelligent tutoring system

ABSTRACT

This paper proposes a novel approach to integrate tutoring functionality into AI systems to counteract the potential decline of human intelligence caused by AI-driven over-automation. Existing explainable AI methods primarily emphasize transparency while lacking inherent educational functionality. Consequently, users are essentially left as passive recipients of AI-driven decisions without any structured learning mechanism in place. To address this, this paper introduces the knowledge-sharing-bridge (KSB), a component designed to transform AI into an active tutor. Unlike traditional intelligent tutoring systems (ITS), which operate separately from AI decision-making processes, the KSB is embedded within AI frameworks, ensuring continuous and context-aware learning opportunities. The proposed framework uses structured knowledge representation tools, such as category maps and word-clouds, to improve the user's understanding of the decisions made by the AI systems. Prototype implementation demonstrates how these elements work together to provide real-time, interactive learning experiences. The results indicate that integrating KSB into AI enhances both explainability and user learning. This approach promotes a more in-depth interaction with AI insights and enables AI systems to become lifelong learning companions, closing the gap between automation and education.

Corresponding Author:

László Csépányi-Fürjes
Institute of Information Science, Faculty of Mechanical Engineering and Informatics
University of Miskolc
3515 Miskolc, Egyetem str. 1, Hungary
Email: laszlo.csepanyi-furjes@uni-miskolc.hu

1. INTRODUCTION

Artificial intelligence (AI) and machine learning (ML) are rapidly transforming modern society, offering advanced automation capabilities, decision support, and creative content generation. While these innovations bring significant benefits, they also pose critical challenges to human development [1], [2]. One pressing concern is the potential decline of human cognitive engagement due to AI systems increasingly taking over complex tasks [3]. As AI becomes more prevalent in decision-making, humans may become overly reliant on these systems, risking a loss of expertise and mental autonomy. Research emphasizes the importance of meaningful work for human well-being [4], highlighting the satisfaction derived from skill utilization and refinement [5]. At the same time, there is an urgent societal demand for lifelong, universally accessible continuous learning opportunities to adapt to this technological shift. The emergence of AI large language models (LLMs) like ChatGPT [6], capable of complex task execution and natural language interaction [7], underscores this need.

Although explainable AI (XAI) has emerged to address AI's transparency issues, providing explanations alone does not ensure user learning [8]. XAI methods clarify how decisions are made [9] but often fall short of guiding users through the conceptual understanding required to internalize AI-generated insights [10]–[12]. In contrast, intelligent tutoring systems (ITS) are designed to support active learning through personalization and feedback [13]. However, ITS are typically domain-specific, standalone systems and are not integrated into everyday AI applications. This disconnection limits their accessibility and usefulness during real-world AI interactions.

There is a large volume of published studies attempting to address this issue. XAI techniques - such as model-specific visualizations [14], [15], local surrogate models like local interpretable model-agnostic explanations (LIME) [16], and counterfactual explanations - have improved the interpretability of complex models, particularly in regulated or safety-critical fields [17]. Recent work has explored the synergy between XAI and ITS, emphasizing the need for explanations that support instruction, not just interpretation [18]. interactive machine learning (IML) [19]–[22] and knowledge-sharing techniques in multi-agent systems (MAS) [23] have also contributed frameworks for feedback and collaboration, yet their focus remains on improving machine performance rather than enhancing human learning. Educational AI and ontology-driven ITS systems provide adaptive instruction [24]–[27], but they are often decoupled from the tools and platforms where users encounter AI-generated decisions in practice.

Despite advancements in ITS, access to high-quality educational support remains uneven, especially for learners with limited resources. Recent work has explored the use of decentralized technologies, such as the Ethereum blockchain, to democratize tutoring services and reduce educational inequality by offering scalable, low-cost solutions [28]. While promising, such approaches focus primarily on logistical and economic accessibility rather than on integrating tutoring capabilities directly into the AI agents themselves.

This reveals a significant gap: current AI systems lack integrated mechanisms to promote user learning during real-time interaction. While explanations help build trust, they do not teach. Similarly, ITS solutions offer effective pedagogy but are not embedded within everyday AI tools, leaving a void where learning could - and should - occur.

To address this, we propose a novel framework that embeds an implicit tutoring mechanism directly into AI systems through the introduction of the knowledge-sharing-bridge (KSB). The KSB converts conventional AI agents into hybrid entities that can serve two purposes: executing tasks and instructing users. By combining core XAI functions with interactive, personalized teaching elements, the KSB enables continuous, contextual learning without requiring users to leave the environment where the AI operates. Prior research emphasizes that both intrinsic and extrinsic motivations significantly influence individuals' intention to share knowledge, particularly within formal virtual communities. The proposed KSB component seeks to leverage these motivational insights by designing AI systems that not only explain but also encourage and facilitate user learning, acting as a motivational partner within interaction.

The KSB comprises four interlinked components: explain (XAI engine), report (operational analytics), control (user configurability), and teach (instructional guidance). This integration allows AI systems to not only justify their actions but also to act as informal tutors, gradually enhancing user competence. By utilizing structured knowledge representations- such as category maps and word-clouds - our framework makes complex decision logic intuitively accessible and pedagogically valuable.

This paper presents the high-level design and a prototype implementation of the KSB-enabled Teaching AI framework. The proposed solution fills a critical void in current AI applications, offering a novel pathway to blend automation with embedded learning - ensuring that AI systems not only inform but educate their users in real time.

The article is organized as follows: section 2 details the theoretical modeling (TM) and high-level-design (HLD) process used to develop the KSB framework, outlining the design principles and sub-component functionalities. Section 3 presents the theoretical validation of the framework as well as the proposed algorithms for prototype implementation, discussing its advantages and challenges. Finally, Section 4 concludes the paper, summarizing the contributions and outlining potential future research directions.

2. METHOD

This section explores the framework development process by listing the guiding principles that determined the architecture. It defines and examines the components and their interactions of the KSB framework. Additionally, it outlines the validation and analysis steps of the theoretical framework model.

2.1. Framework development through theoretical modeling

The KSB framework was developed using a TM approach grounded in principles from knowledge space theory (KST) [29], ontology-based educational modeling [30], and the evolving knowledge space graph (EKSG) [31]. KST provided the foundation for representing knowledge as a structured set of prerequisite-dependent units, while the ontology-based model enabled the semantic categorization of learning content, learner profiles, and strategies. By introducing the concept of abstract-time into this merged model, the EKSG was developed to account for the dynamics of fast-changing knowledge in today's technological landscape.

This theoretical integration formed the basis for identifying a key gap in current AI applications: the lack of universal, implicit, and continuous learning opportunities embedded within the systems themselves. Thus, the KSB framework was designed not as an external educational tool, but as an internal component of any AI system interacting directly with users, making learning a result of usage rather than a separate, explicit process.

2.2. High-level-design principles guiding the KSB framework

XAI, while improving transparency, focuses on explanation rather than active teaching. This gap necessitated a framework that seamlessly embeds teaching functionality into AI systems. The development of the KSB framework was guided by the following core design principles:

- Intelligence Augmentation (IA) over AI: The framework prioritizes enhancing human intelligence and autonomy, rather than replacing it.
- Implicit learning over explicit training: Learning should occur in the flow of using technology, reducing barriers like cost, time, and motivation.
- Universality and accessibility: The KSB is designed to be integrated into any AI system regardless of domain, thus enabling lifelong learning for all users.
- Transparency and Trust: By explaining and reporting decisions, the AI can foster a more trusted relationship with users.

These principles seek to transform AI from a marginalizing force to an empowering tool. The necessity for KSB arose from multiple theoretical and practical observations. Despite rapid advancements in AI, most systems lack the ability to teach users how they function, leading to dependence rather than empowerment. Existing ITS focus narrowly on academic domains and are not embedded within general-purpose AI systems. Furthermore, emerging challenges related to AI over-automation and loss of meaningful human work highlighted the urgent need for AI systems to play a more supportive and educational role in human society. Consequently, the KSB was conceptualized as an internal AI module designed to transfer knowledge from the AI to the user through intuitive and context-sensitive interactions.

2.3. Rationale behind KSB subcomponents

The KSB framework, depicted in Figure 1, was designed using a HLD process, defining the overall architecture and subcomponent interactions. The inclusion of the four sub-components - explain, report, control, and teach – as well as the remaining sub-components was guided by the need of facilitating user learning:

- Explain: Integrates XAI techniques to make the AI's decisions interpretable. This supports cognitive understanding and enhances user trust. It is not enough to provide low level explanation; it is advisable to translate the explaining result into a human understandable format. For instance, in a legal environment explainability means legal explanation.
- Report: Offers statistical and performance feedback to users, helping them track system behavior and identify improvement areas in their own interaction or decision-making. To prepare thoughtful decisions in terms of AI control it is crucial to monitor the working of the system as well as to follow the communication between the user and the AI agent. Every user must be able to analyze her own interaction with the system, this is why the Report subsystem must be part of the KSB. There are already known metrics to evaluate AI agents (e.g., success rate, accuracy, etc.) and according to the increasing demand of control it is sure that more are to come.
- Control: Empowers users with configurable options that promote autonomy and self-regulation, aligning with principles from self-directed learning. To implement controllability, AI developers must put a set of rules into force in the Control subsystem so that the external users can intervene in the working of the system in a predetermined way. To avoid demonization of AI technology there must be much larger control possibility provided to the users than today, however it means a great challenge to the system's security. Therefore, to avoid malicious interactions, careful implementation of the Control subsystem regarding security issues is crucial.
- Teach: Provides personalized, context-sensitive instructional content, enabling users to develop procedural knowledge on how to replicate or modify the AI-driven task. In a healthy synergy, AI learns from humans and humans learn from AI. Teach is the subsystem that facilitates human learning by providing premeditated feedback in a teaching manner. As opposed to the explain subsystem, where the aim is to understand the AI's response, the Teach subsystem provides information how to learn the skills

of the AI agent. For example, if a legal document is being rejected by the AI-classifier agent the explain subsystem can point on to the key factors why the document was rejected, while the Teach subsystem gives information how the document needs to be constructed to get it accepted.

– Internal gateway: Having an internal gateway makes it possible to scale the internal components of the KSB so that can be extended and customized. Either by adding more components or more instances from existing components the internal gateway can encapsulate the communication and can realize internal security features that protect the subcomponents from malicious impacts. By providing private API the system can be integrated seamlessly into various AI systems.
– User interface and the integration layer: Users are communicating the AI agent using the user interface (UI). In the proposed framework the UI must be extended for the user to be able to interact with the KSB and to access learning materials, get explanation, realize control or to query statistical information. These functionalities can be implemented separately from the core AI functionality making it possible to apply advanced learning capabilities. The integration layer provides public API to implement the core functionality as well as to access the KSB functionalities behind the gateway. It ensures seamless integration with various AI systems and platforms, adhering to industry standards.
– Core AI functionality: The proposed model is describing a simplified AI agent that consists of the trained AI model as well as a processor layer that implements the business logic of the system. Usually the input/output is realized by the sensor and effector subcomponents. Furthermore, it is important to mention that the system always need a database where the core functionality related data, user related information or system settings are stored. In the proposed model the KSB related data is also located in this database.

Together, these modules transform the AI from a static decision-maker into a dynamic, educative agent that can provide support to users in real-time, enabling skill development and knowledge enhancement. The inclusion of the KSB into general AI systems introduces a new model of implicit, continuous learning. Unlike formal educational systems or traditional e-learning platforms, the KSB supports just-in-time knowledge delivery, addressing the knowledge needs of users as they arise during real-world tasks. This aligns closely with the goals of lifelong learning and adaptive learning environments but broadens their reach to include all AI-driven interactions, not just educational applications.

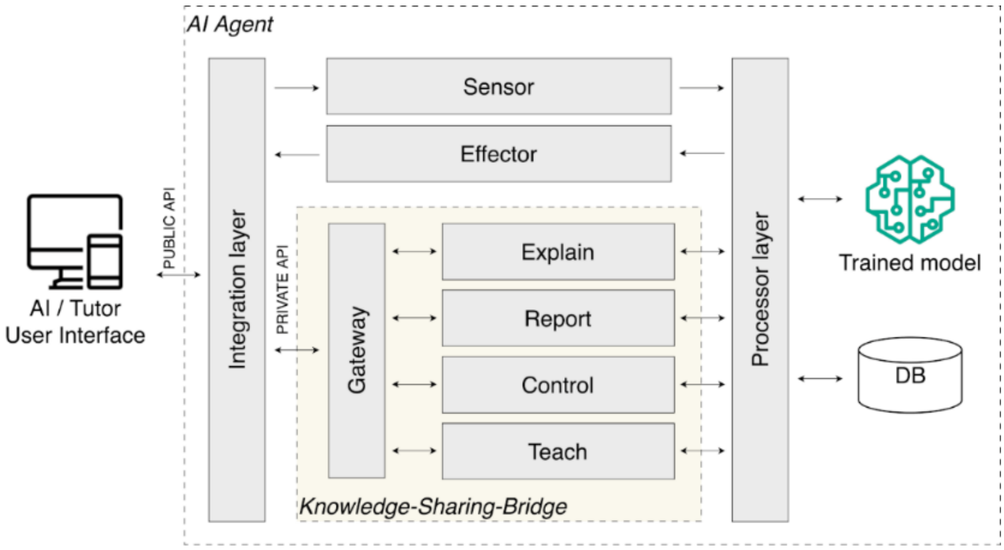


Figure 1. KSB framework

2.4. Theoretical validation

The framework was subjected to Theoretical Validation (TV). The TV of the KSB framework involved a comprehensive analysis of its advantages and potential challenges. Key advantages include universal application, user empowerment, and enhanced trust and transparency. However, challenges such as the complexity of XAI integration and the need for high-quality automated tutoring solutions were also identified. The research acknowledges limitations, including the high-level description of the framework and the use of small university student group for preliminary evaluation.

3. RESULTS AND DISCUSSION

In this section, we explore the practical realization of the KSB framework through a prototype implementation, showcasing how the explain, report, control, and teach components operate in an integrated AI environment. Also, this section highlights how the KSB components contribute to the overarching goals of user empowerment and intelligence augmentation.

3.1. Workflow

The aim of the answer validator (AV) is to automatically evaluate a textual answer coming from a customer service employee. AV is a simple, language-model-based AI system that acts as a virtual customer service trainer. The prototype KSB was implemented as part of the AV module.

The workflow of the system is as follows: the prototype UI shows a question to the employee, then the employee submits a textual answer using the UI to the AV module through the public API that is implemented as a REST service. The REST response received contains the evaluation result as well as a universal unique identifier (UUID) to identify the communication flow. The UI then extracts the UUID and requests an explanation as well as teaching information from the KSB through the public API. The integration layer forwards the request using the private API to the KSB gateway. The gateway routes the request to the explain and Teach subcomponents. Both components are fetching the necessary information from the DB by the UUID and using the capability of the AI language model to generate response. The UI displays both information to the user and expects a corrected answer. The process goes until the answer reaches the acceptance criteria level of the AV module.

3.2. XAI engine of the explain and teach modules

The proposed domain tutoring system is different from the general tutoring systems in many aspects. The main differences include the following elements:

- local scope problem domain
- small knowledge topic focusing on a specific problem
- flexible content
- open interface

The fundamental elements of the framework comprise the explain and teach modules, which generate a clear explanation of the prediction process carried out by the neural network and supply guidance to the user on how to enter an input that the neural network (NN) recognizes as a correct response. The XAI engine of the explain module will analyze the NN architecture and generate an interpretable representation of the NN's knowledge model. Considering the usual knowledge representation formats used in expert systems, we can highlight the following two tools:

- Category map: it shows a visual representation of the relationship between the feature sets and categories.
- Word-cloud: it shows the key concepts related to decision process.

3.3. Proposed algorithm of category Map generation for functional approximation

The category map is easy to understand for humans, this kind of representation format is used in self organization map [32] or in cross reference tables [33]. In our investigation, we focus on the generation of the category map. The domain of the map is a subset of the feature space, usually it is a sub-cube. Each point in the cube represents a feature vector which corresponds to an object in the problem domain. The map shows the corresponding category values or regression value related to the given position. The resulting map is very useful information for the users to learn which parts of the objects space belong to which categories. Regarding the generation of the category map, we proposed and compared two approaches:

- Feed forward generation (model agnostic approach)
- Backwards propagation (model specific)

We assume that the object space (feature space) X is D dimensional vector space: $X \subset R^D$ and Y is the set of categories or regression values. The investigated neural network model is denoted by Λ. In the feed forward method, we generate random points in X and calculate $\Lambda(x), x \in X$ values. The resulting map shows the distribution of the different categories or regression values. As in the homogeneous areas we need lower granularity than in the border regions, it seems useful to have a dense sampling in border regions and a rare sampling in the homogeneous zone.

To manage the inhomogeneity, the object domain cube is partitioned into a grid. For each cell, we introduce a homogeneity factor, as the entropy value based on the category distribution of the current cell:

$$h = ent(c) = -\sum_{i \in c} p_i \log p_i \qquad (1)$$

thus, in the generation of the sampling process, the probability to select a cell is proportional to the corresponding h value.

In the case of backwards propagation, we approximate the inverse function of the network. Usually, the network represents a not invertible function, thus we propose a probabilistic approximation method. For a given output $y \in Y$, the proposed method selects one input point x in the feature space randomly from the set of points related to y, i.e. $x \in \{v \mid \Lambda(v) = y\}$. We assume now a MLP NN architecture.

The calculation steps are based on the following considerations. Having a $y \in Y$. selected randomly, let us take the output layer with neurons $\{n_i\}$. The output of n_i is y_i. First, we determine s_i, where $y_i = f(s_i)$, where $f()$ is the activation function of the neuron. Usually, the activation function is invertable. If not, then we select one value randomly from the domain yielding y_i. Thus, we have $\{s_i\}$ for all nodes of the layer. Next, we calculate $\{z_i\}$ values, where $s_i = \sum_j z_j w_{ij} + b_i$ and w_{ij}, b_i are the weight and bias values. As it is a linear mapping, the solution set $\{z\}$ is a hyperplane. The z_i values denote the output of the previous layer. As the solution must meet all equations belonging to the nodes of the current layer, we get a system of equations to be solved.

$$s_1 = \sum_j^m z_j w_{1j} + b_1$$

$$s_2 = \sum_j^m z_j w_{2j} + b_2$$

$$s_n = \sum_j^m z_j w_{nj} + b_n \tag{2}$$

In (2), n denotes the neurons in the current layer, while symbol m is the size of the preceding layer in the NN architecture. In the general case, the solution is a single unique point or a linear subspace or it can be empty. If no solution exists, we will terminate this process. Otherwise, we take only one point from this plane as a solution. The calculated $\{z_i\}$ vector can be considered as the output of the previous layer; thus, we can repeat the same algorithm as presented before to get the input of the previous layer. On this way, we get an input vector x , where $y = \Lambda(x)$. In this way, for any selected category we get an approximation of the category distribution. The implementation of the backwards propagation method for state space discovery is based on the Python method in Figure 2.

```
def inv_layer(model, L, Y0):
    weights, biases = model.layers[L].get_weights()
    Y = np.copy(Y0)
    if L < 2:
        Y = iactivf(Y)
    B = Y - biases
    W = weights.transpose()
    B = B.transpose()
    C = np.zeros(W.shape[1])

    x_p = np.linalg.lstsq(W, B, rcond=None)[0]

    N = null_space(W)
    z = np.random.randn(N.shape[1])
    X = x_p + N @ z
    X = np.array(x_p)
    X = X.transpose()

    return X
```

Figure 2. State space discovery

According to the test experiments with the backward approach, this method is suitable only for simple network structures as the method suffers from many issues. The main issues are the following:

- Complexity of solving the summation inversion. Here, there are two key difficulties. First, the equation system in general, can be solved only with a conditional optimization method, as the activation function of the preceding layer generates points in a specific subfield of the numerical space. For example, in the cases of RELU function, only non-negative values are generated. The second problem is that the method usually yields only a weak approximation, thus each layer increases the prediction error.
- The high computational costs. The applied methods are usually based on some iterations or evolutionary approaches; thus, the cost of backward iteration is much higher than the cost of forward prediction.

In the case of forward propagation space discovery, we use a regression problem to demonstrate the benefits of the proposed position weighting approach. The real function to be approximated is shown in Figure 3. The generated approximation points are presented in Figure 4, where Figure 4(a) presents random position selection and Figure 4(b) shows position selection using the proposed metrics. As the results show, the controlled generation highlights the key sections in the figure in a significantly better way than the random approach.

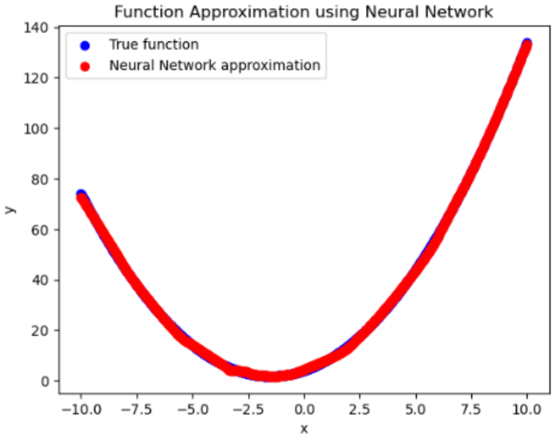
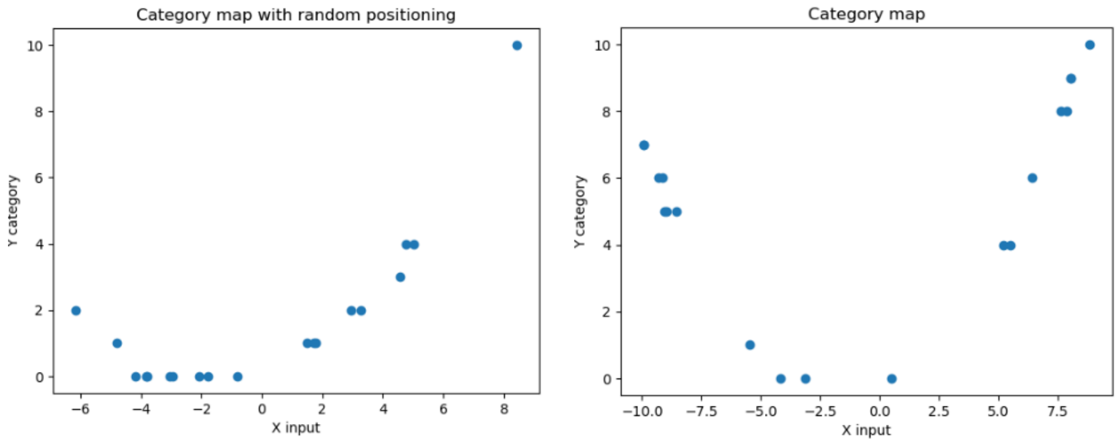


Figure 3. The real function y = f(x) to be approximated (x = input variable; y = target variable)



(a) (b)

Figure 4. Generated approximation points (a) random position selection (b) proposed position selection

3.4. Proposed algorithm of word-cloud generation of the prototype AV module

To understand the algorithm of the prototype AV module's evaluation, let us observe the AV process itself first. In the open text AV domain, there are two main use cases. In case A, exact words need to be used to answer, while in case B the meaning of the answer is important, but the words themselves can differ. In other words, in case A the answer can be verified by comparing the words one by one while in case B the semantics of the answer needs to be matched. Use case B is more common in real life. For example, a case A question may look like this: "What do the initials HAL for the HAL 9000 computer mean in the film 2001: A Space Odyssey?" The right answer looks like this: "Heuristically programmed algorithmic computer". A case B question may look like this: "How do you greet a customer?". It can be a question in a customer service environment. The right answer could be: "Hello, how may I help you?". But semantically it is also acceptable to answer like this: "Hi, can I help you?".

The implemented AV module's algorithm evaluates textual answers by using cosine similarity (X, A) between the expected answer $X = \{x_1, x_2, \dots, x_m\}$, and the actual answer $A = \{a_1, a_2, \dots, a_n\}$, where m is the number of expected answer tokens and n is the number of actual answer tokens. The implemented logic can be used in both case A and case B. For text representation the system uses the STSB-ROBERTA-LARGE language model [34].

The implemented prototype explain subcomponent evaluates the given answer words, one by one and displays them in a word-cloud like this: the bigger the word in the cloud the further it takes the answer away from the expected answer. To calculate the word relevance the algorithm skips tokens a_i of the answer one by one and generates new answers $A_i = \{a_1, \dots, a_{i-1}, a_{i+1} \dots, a_n\}$ and calculates similarity between the resulting new answer and the expected answer (X, A_i). The similarity value in this case is in correlation with the relevance of the skipped token. In other words, the bigger the word in the cloud the more you need to change that word to get to the correct answer.

The prototype teach subcomponent works similarly to the explain subcomponent. The only difference is that it calculates the relevance of the expected answer's words and provides a cloud with the few most relevant expected words only. This gives a hint to the user about what terms should be included in the answer to get acceptance from the AV module.

The report subcomponent provides information about the number of questions, evaluated questions and statistics about the result scores in JSON format. Using the control subcomponent the learner can change the evaluation scheme to another one that is better fit to the learner's needs. For example, there is a predefined evaluations scheme that gives a binary answer, like: GOOD/WRONG, or another one that can give a grade based on the similarity score.

3.5. Experiment with the prototype AV module

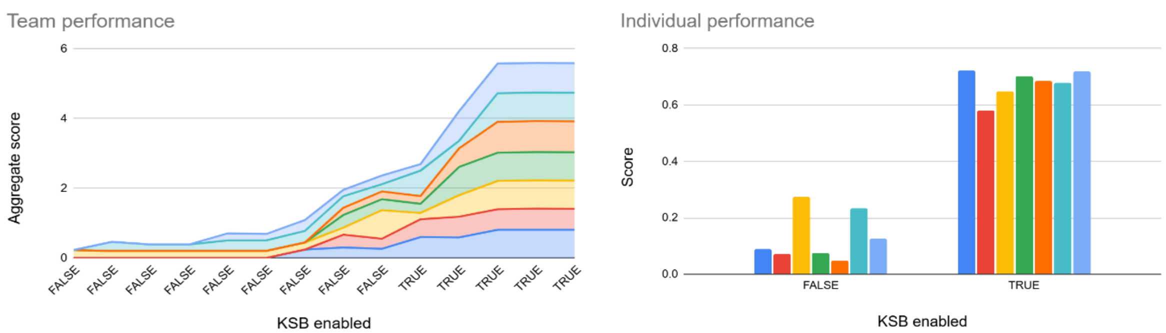
The implemented prototype AV system with built in KSB was evaluated with a group of 7 university students. The following evaluation objectives (EO) were defined:

- EO1: To evaluate how the overall team performance is affected by the KSB component.
- EO2: To evaluate how the individual user's performance is affected by the KSB component.
- EO1 and EO2 are motivated to empirically demonstrate the usefulness of the proposed KSB component by assessing the results of students when they can access the KSB component and when they have access only to the AV module itself.

During the experiment a set of open-ended questions were presented to the students. The students started giving answers and receiving evaluation responses from the AV module. At a certain point the system setting was altered by the administrator, so the students started getting not only evaluation responses, but also explanation and teaching word-clouds from the KSB component. The aim was to observe how the students are performing when the KSB component is available and when it is not available, as depicted in Figure 5.

EO1: Within the team there are better performing and less performing students, just like in any team. When the students were able to use only the AV module (KSB is enabled = FALSE) the better performing students got higher scores and started achieving better results quicker than the others. When KSB was activated (KSB is enabled = TRUE) the gap between the students becomes lesser and the overall team performance became even as Figure 5(a) is showing.

EO2: Since the presented question was new to the students even the better performing students were struggling with them, but when the KSB module became available all students started performing on a much higher level as shown in Figure 5(b).



(a) (b)

Figure 5. Evolving performance over time (Colors representing individual learners) (a) team performance (b) individual performance

3.6. Discussion

The experiment with the university student group revealed several important insights. Figure 5(a) demonstrates that enabling the KSB component significantly reduced the performance gap between high and low-performing students. This suggests that the KSB subcomponents support personalized learning by adapting to the user's needs and helping weaker users catch up, thus promoting more equitable outcomes across a group. Figure 5(b) further indicates that all users, regardless of initial skill level, benefited from the KSB's interactive feedback, ultimately improving their answer quality.

These findings imply that the KSB framework has strong potential to be integrated into AI systems where decision explanation and user guidance are required. In domains such as customer service training, onboarding, or general workplace learning, the ability to teach users on the fly in a context-sensitive and non-intrusive way could be transformative. Future applications may extend to healthcare, finance, and any domain where trust, understanding, and human-AI collaboration are vital.

Furthermore, the explain and teach modules powered by XAI principles introduce transparency into a traditionally opaque process. The use of category maps and word clouds provides interpretable visual cues, making it easier for users to understand why their responses are incorrect and how to improve. The report and control modules, while more administrative in nature, contribute to a customizable and trackable learning experience, offering flexibility that aligns with continuous, lifelong learning goals.

From a theoretical perspective, the prototype implementation validates the hypothesis that embedding a knowledge-sharing component within AI systems can bridge the gap between performance evaluation and human learning. Compared to the original AV-only scenario, the KSB-enhanced system demonstrates that explainability and guidance not only improves outcomes but also empowers users by making AI decisions comprehensible and actionable.

While the system currently targets a relatively narrow domain with a small test group, the implications are far-reaching. These early results, though limited in scope, suggest that knowledge-sharing AI frameworks could be universally beneficial in enhancing human learning across various contexts. However, scalability, domain adaptation, and further refinement of the XAI components will be critical in future developments.

4. CONCLUSION

This paper introduces a framework that embeds an implicit tutoring mechanism directly into AI systems. The proposed framework consists of four major subcomponents, namely explain, report, control and teach. The explain and teach subcomponents powered by a dedicated XAI engine prove their feasibility through successful implementation. The results suggest promising applications across a range of domains where human-AI collaboration is essential. Beyond customer service training, the proposed framework could be extended to education, corporate learning environments, technical support, and any scenario in which users benefit from immediate, transparent feedback and contextual guidance. For the broader research field, this work contributes to the growing movement toward human-centered AI by demonstrating that explainability and interactivity are not just desirable features, but key drivers of usability, learning, and trust. For the community of practice - including educators, developers, and researchers - this prototype provides a blueprint for designing systems that don't just automate decision making but actively foster understanding and growth. Outstanding questions remain regarding scalability, domain generalization, and the performance of the system with larger, more diverse user groups.

FUNDING INFORMATION

Authors state no funding involved.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
László Csépányi-Fürjes	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	
László Kovács	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	

C : Conceptualization	I : Investigation	Vi : Visualization
M : Methodology	R : Resources	Su : Supervision
So : Software	D : Data Curation	P : Project administration
Va : Validation	O : Writing - Original Draft	Fu : Funding acquisition
Fo : Formal analysis	E : Writing - Review & Editing	

CONFLICT OF INTEREST STATEMENT

Authors state no conflict of interest.

DATA AVAILABILITY

The prototype implementation is openly available at https://github.com/csepanyifurjes/uom-ksb/tree/main/results.

REFERENCES

[1] R. Fu, Y. Huang, and P. V. Singh, "Artificial intelligence and algorithmic bias: source, detection, mitigation, and implications," in *Pushing the Boundaries: Frontiers in Impactful OR/OM Research*, INFORMS, 2020, pp. 39–63. doi: 10.1287/educ.2020.0215.

[2] T. Grønsund and M. Aanestad, "Augmenting the algorithm: Emerging human-in-the-loop work configurations," *Journal of Strategic Information Systems*, vol. 29, no. 2, p. 101614, Jun. 2020, doi: 10.1016/j.jsis.2020.101614.

[3] U. De, P. O. Nanterre, L. Défense, and B. David, "Computer technology and probable job destructions in Japan: an evaluation Computer technology and probable job destructions in Japan: an evaluation *," 2015.

[4] S. Voswinkel, "Meaningful work," *Comparative Sociology*, vol. 19, no. 6, pp. 741–755, Dec. 2020, doi: 10.1163/15691330-12341529.

[5] T. Turja, J. Minkkinen, and S. Mauno, "Robotizing meaningful work," *Journal of Information, Communication and Ethics in Society*, vol. 20, no. 2, pp. 177–192, Apr. 2022, doi: 10.1108/JICES-06-2021-0063.

[6] OpenAI, "ChatGPT," ChatGPT model (GPT-4). Accessed: Sep. 17, 2024. [Online]. Available: https://chat.openai.com/

[7] A. L. Overono and A. S. Ditta, "The rise of artificial intelligence: a clarion call for higher education to redefine learning and reimagine assessment," *College Teaching*, vol. 73, no. 2, pp. 123–126, Apr. 2025, doi: 10.1080/87567555.2023.2233653.

[8] D. Castelvecchi, "Can we open the black box of AI?," *Nature*, vol. 538, no. 7623, pp. 20–23, Oct. 2016, doi: 10.1038/538020a.

[9] P. P. Angelov, E. A. Soares, R. Jiang, N. I. Arnold, and P. M. Atkinson, "Explainable artificial intelligence: an analytical review," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 11, no. 5, Sep. 2021, doi: 10.1002/widm.1424.

[10] A. Ignatiev, "Towards trustable explainable AI," in *IJCAI International Joint Conference on Artificial Intelligence*, California: International Joint Conferences on Artificial Intelligence Organization, Jul. 2020, pp. 5154–5158. doi: 10.24963/ijcai.2020/726.

[11] R. Confalonieri, L. Coba, B. Wagner, and T. R. Besold, "A historical perspective of explainable Artificial Intelligence," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 11, no. 1, Jan. 2021, doi: 10.1002/widm.1391.

[12] F. K. Dosilovic, M. Brcic, and N. Hlupic, "Explainable artificial intelligence: A survey," in *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2018 - Proceedings*, IEEE, May 2018, pp. 210–215. doi: 10.23919/MIPRO.2018.8400040.

[13] B. I. Martin, "Intelligent tutoring systems: the practical implementation of constraint-based modelling," *Computer Science*, 2001, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.150.7481&rep=rep1&type=pdf

[14] R. Dwivedi *et al.*, "Explainable AI (XAI): core ideas, techniques, and solutions," *ACM Computing Surveys*, vol. 55, no. 9, pp. 1–33, Sep. 2023, doi: 10.1145/3561048.

[15] P. Linardatos, V. Papastefanopoulos, and S. Kotsiantis, "Explainable ai: a review of machine learning interpretability methods," *Entropy*, vol. 23, no. 1, pp. 1–45, Dec. 2021, doi: 10.3390/e23010018.

[16] A. Holzinger, A. Saranti, C. Molnar, P. Biecek, and W. Samek, "Explainable AI methods - a brief overview," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 13200 LNAI, 2022, pp. 13–38. doi: 10.1007/978-3-031-04083-2_2.

[17] V. Bidve *et al.*, "Use of explainable AI to interpret the results of NLP models for sentimental analysis," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 35, no. 1, pp. 511–519, Jul. 2024, doi: 10.11591/ijeecs.v35.i1.pp511-519.

[18] C. Conati, O. Barral, V. Putnam, and L. Rieger, "Toward personalized XAI: A case study in intelligent tutoring systems," *Artificial Intelligence*, vol. 298, p. 103503, Sep. 2021, doi: 10.1016/j.artint.2021.103503.

[19] J. J. Dudley and P. O. Kristensson, "A review of user interface design for interactive machine learning," *ACM Transactions on Interactive Intelligent Systems*, vol. 8, no. 2, pp. 1–37, Jun. 2018, doi: 10.1145/3185517.

[20] N. Pfeuffer *et al.*, "Explanatory interactive machine learning: establishing an action design research process for machine learning projects," *Business and Information Systems Engineering*, vol. 65, no. 6, pp. 677–701, Dec. 2023, doi: 10.1007/s12599-023-00806-x.

[21] S. Teso, Ö. Alkan, W. Stammer, and E. Daly, "Leveraging explanations in interactive machine learning: An overview," *Frontiers in Artificial Intelligence*, vol. 6, Feb. 2023, doi: 10.3389/frai.2023.1066049.

[22] T. Spinner, U. Schlegel, H. Schafer, and M. El-Assady, "explAIner: a visual analytics framework for Interactive and Explainable Machine Learning," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 1064–1074, 2019, doi: 10.1109/TVCG.2019.2934629.

[23] A. Dorri, S. S. Kanhere, and R. Jurdak, "Multi-agent systems: a survey," *IEEE Access*, vol. 6, pp. 28573–28593, 2018, doi: 10.1109/ACCESS.2018.2831228.

[24] J. Traxler, "Distance learning—predictions and possibilities," *Education Sciences*, vol. 8, no. 1, p. 35, Mar. 2018, doi: 10.3390/educsci8010035.

[25] C. Lwande, L. Muchemi, and R. Oboko, "Identifying learning styles and cognitive traits in a learning management system," *Heliyon*, vol. 7, no. 8, p. e07701, Aug. 2021, doi: 10.1016/j.heliyon.2021.e07701.

[26] A. Almasri *et al.*, “Intelligent tutoring systems survey for the period 2000-2018,” *International Journal of Academic Engineering Research*, vol. 3, pp. 21–37, 2019, [Online]. Available: www.ijeais.org/ijaer
[27] K. Mageira, D. Pittou, A. Papasalouros, K. Kotis, P. Zangogianni, and A. Daradoumis, “Educational AI chatbots for content and language integrated learning,” *Applied Sciences (Switzerland)*, vol. 12, no. 7, p. 3239, Mar. 2022, doi: 10.3390/app12073239.
[28] S. Lgarch, M. Hnida, and A. Retbi, “Empowering e-learning through blockchain: an inclusive and affordable tutoring solution,” *International Journal of Electrical and Computer Engineering*, vol. 14, no. 5, pp. 5554–5565, Oct. 2024, doi: 10.11591/ijece.v14i5.pp5554-5565.
[29] J. P. Doignon and J. C. Falmagne, “Spaces for the assessment of knowledge,” *International Journal of Man-Machine Studies*, vol. 23, no. 2, pp. 175–196, Aug. 1985, doi: 10.1016/S0020-7373(85)80031-6.
[30] W. T. Sewunetie, G. Hussein, A. Ahmed, and L. Kovács, “the development and analysis of extended architecture model for intelligent tutoring systems,” *Gradus*, vol. 6, no. 4, pp. 128–138, 2019.
[31] L. Csépányi-Fürjes and L. Kovács, “Evolving graph based knowledge space model for tutoring systems,” *Pollack Periodica*, vol. 19, no. 3, pp. 40–45, Oct. 2024, doi: 10.1556/606.2024.01058.
[32] M. Nikoo, P. Zarfam, and H. Sayahpour, “Determination of compressive strength of concrete using Self Organization Feature Map (SOFM),” *Engineering with Computers*, vol. 31, no. 1, pp. 113–121, Jan. 2015, doi: 10.1007/s00366-013-0334-x.
[33] E. Parthé, L. Gelato, B. Chabot, M. Penzo, K. Cenzual, and R. Gladyshevskii, “Cross — reference tables,” in *TYPIX Standardized Data and Crystal Chemical Characterization of Inorganic Structure Types*, Berlin, Heidelberg: Springer Berlin Heidelberg, 1993, pp. 261–668. doi: 10.1007/978-3-662-10641-9_1.
[34] N. Reimers and I. Gurevych, “Sentence-BERT: sentence embeddings using siamese BERT-networks,” in *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, Stroudsburg, PA, USA: Association for Computational Linguistics, 2019, pp. 3982–3992. doi: 10.18653/v1/D19-1410.

BIOGRAPHIES OF AUTHORS

László Csépányi-Fürjes is a Teaching Assistant and Ph.D. candidate at University of Miskolc (Hungary). He received the M.Sc. degree in Computer Science from the University of Miskolc (ME) in 1998. Since 2023 he has been a Ph.D. student. His research topic is Intelligent Tutoring Systems as well as Artificial Intelligence in Education. He participated in project ALICE at CERN, Geneva, Switzerland in 2000 as a Java Developer. His research interests include Human-Centered AI, AI in Education and Knowledge Representation of Software Systems. He can be contacted at email: laszlo.csepanyi-furjes@uni-miskolc.hu.

Prof. Dr. László Kovács is a full professor at University of Miskolc (Hungary), Institute of Information Science. He is the head of the Department of Software Engineering. His main research area includes database and knowledge base modeling, concept lattice structures, discrete optimizations, and machine learning algorithms in natural language processing. He is the author of 85 journal publications and 109 conference publications. He is currently the supervisor of 6 Ph.D. candidates. He can be contacted at email: laszlo.kovacs@uni-miskolc.hu.