Sharing Behaviors in Educational Games:
A Framework for EEDU Elements Mathematics Game

Harri Ketamo

Abstract—User generated media highlights sharing videos, images and texts in social media, as well as sharing character outfits and maps in games. However, behavior is one of the aspects that are not shared. The aim of this study is to show how user generated behaviors can be recorded and shared in educational games. According to examples on this study, game- and media developers can design extensions that enable users to easily construct behaviors. From a game consumer point of view, the most interesting part is in developing behaviors, sharing them and finally playing with them, or against them.

Index Terms—Artificial intelligence, games, learning, mathematics, user generated content.

I. INTRODUCTION

User-generated content, such as the choice of a game character's outfit, its textures and clothing, as well as editing game scenarios, has been an integral part of games for a long time. It seems that users require features and activities that can be personalized and shared. However, there is no such interest in sharing behavior, strategies or game character personalities. All in all, this is not completely because of a lack of technologies for modeling and sharing AI -related contents [1].

Unlike visual objects, sounds or texts, behavior is a complex phenomenon. This complexity has set limits for developing AI's that could enable behavior construction without programming or scripting. In fact, AI programming traditionally requires not only programming skills, but mathematical skills also. Another point of discussion is whether game AI is about intelligence or behavior. Baekkelund [2] argued that game AI is far more difficult to determine than academic AI. Furthermore, while academic AI research focuses on perfect or optimal behavior, game AI should be entertaining: Game AI is allowed to cheat or be 'stupid' in order to achieve the illusion of intelligent and entertaining behavior [3].

Sports behavior modeling is challenging. Several games have received negative feed-back related to unrealistic non-player character behavior. Furthermore, in some multi-player games it is relatively easy to guess when one is playing against AI and when one is playing against another human player. On the other hand, the construction of human-like behavior in, e.g. football and hockey is very challenging. Even steering behaviors are surprisingly complex [4]. One

other interesting approach can be found in Forza Motorsport, a game in which AI can learn to play like the player does in terms of driving patterns.

However, when discussing learning and behavior, we have to make a distinction between behavior as cognitive behavior and behavior as scripted behavior.

Behavior modeling has a long research background: Neural and semantic networks, as well as genetic algorithms, are utilized to model a user's characteristics, profiles and pat-terns of behavior in order to support or challenge the performance of individuals. Behavior recording have been studied and used in the game industry for a good time. In all recent studies the level of behavior is limited, more or less, to observed patterns, [5]-[7]. Furthermore, agent negotiation and it's scripted behavior [8] as well as agent based information retrieval [9] in web-based information systems has been studied for a long time.

In this study, user behavior, competence and learning were seen as Semantic (neural) network that produces self-organizing and adaptive behavior/interaction. The AI technology developed, emulates the human way to learn: According to cognitive psychology of learning, our thinking is based on conceptual representations of our experiences and relations between these concepts. Phenomena when the mental structure change is called learning.

In terms of constructive psychology of learning, people actively construct their own knowledge through interaction with the environment and through reorganization of their mental structures. The key elements in learning are accommodation and assimilation. Accommodation describes an event when a learner figures out something radically new, which leads to a change in his/her mental conceptual structure. Assimilation describes events when a learner strengthens his/her mental conceptual structure by means of new relations [10].

The novelty value of this study is in approach: to build technologies that enable easy construction of intelligent and human like behaviors.

II. RESEARCH OBJECTIVES

The aim of this study is to show how user generated behaviors can be shared in educational games. The challenge/novelty of the study is in the game settings: In order to share user generated behaviors, we should have a computational model that can be 1) easy to construct (user experience point of view), 2) extensible and scalable (useful for game and media developers) and 3) transferrable and reusable

The study can be seen as being a traditional design study

with iterative cycles. The procedure of development was limited to 1) define what kind of activities are observed and taught and 2) construct interface for teaching behavior (according to definitions). The evaluation of the study is not meant to be done in terms of cognitive sciences: it is only done in order to evaluate the usefulness of the method.

III. DESIGNING FRAMEWORK

The framework is based on the author's previous work: research articles have been published from the point of view of cognitive science [11]-[13] and from a technological point of view [14]-[16]. The AI behind framework emulates the human way to learn: According to cognitive psychology of learning, our thinking is based on conceptual representations of our experiences and complex relations between these concepts and experiences. Phenomena when the mental structure change is called learning.
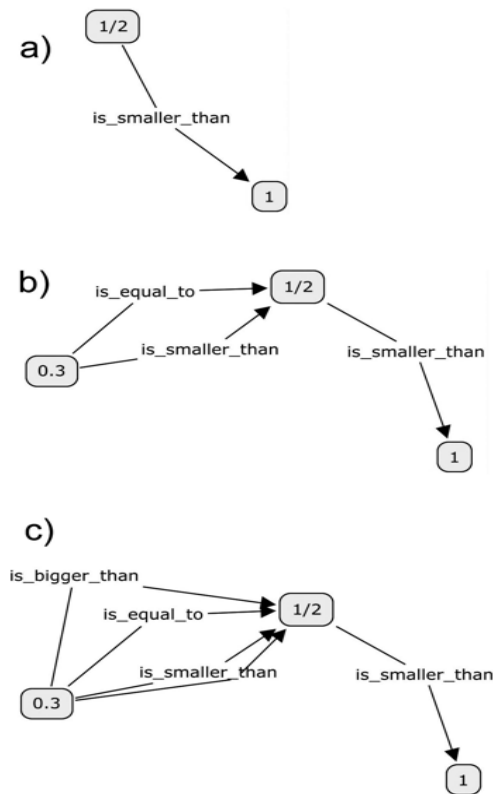

Fig. 1. Semantic network and its development during the teaching phases, part 1.

At first, the player teaches the relation between 1 and 1/2. The question, created by the player is: "Is ½ smaller than 1?" The agent does not have previous knowledge, so it will guess. In case it guesses "true" and the player's evaluation is "Correct." The relation "½ is smaller than 1." is formed in the conceptual structure (Fig. 1 a). The same would occur in a case where the agent guesses "False" and the player evaluates "Wrong".

A teaching phase consists of a question creation and evaluation –pair. Each teaching phase adds new relations into the conceptual structure. Furthermore, if the concept is not taught before, the new concept is also added into the conceptual structure during the teaching phase. The following example briefly describes the development of

conceptual structures in the agent's mind during teaching phases. The understanding of how an agent's conceptual structure develops during playing is important in order to be able to interpret the results of the study. Each teaching phase is recorded in a semantic (conceptual) network within the game AI with one or more 'is (not/option) related to', 'is (not) bigger', 'is (not) equal', etc. relations. The following example is based on is (not) bigger and is (not) equal relations. In the second teaching phase, the player teaches a relation between 0.3 and ½, with the question "Is 0.3 bigger than ½?" The player knows that the question is false, but the agent answers (guesses) "True". So the player evaluates "wrong" and the agent determines that the correct answer is either "0.3 is equal to ½" or "0.3 is smaller than ½". The conceptual network in the agent's mind grows by both of these relations (Fig. 1 b).
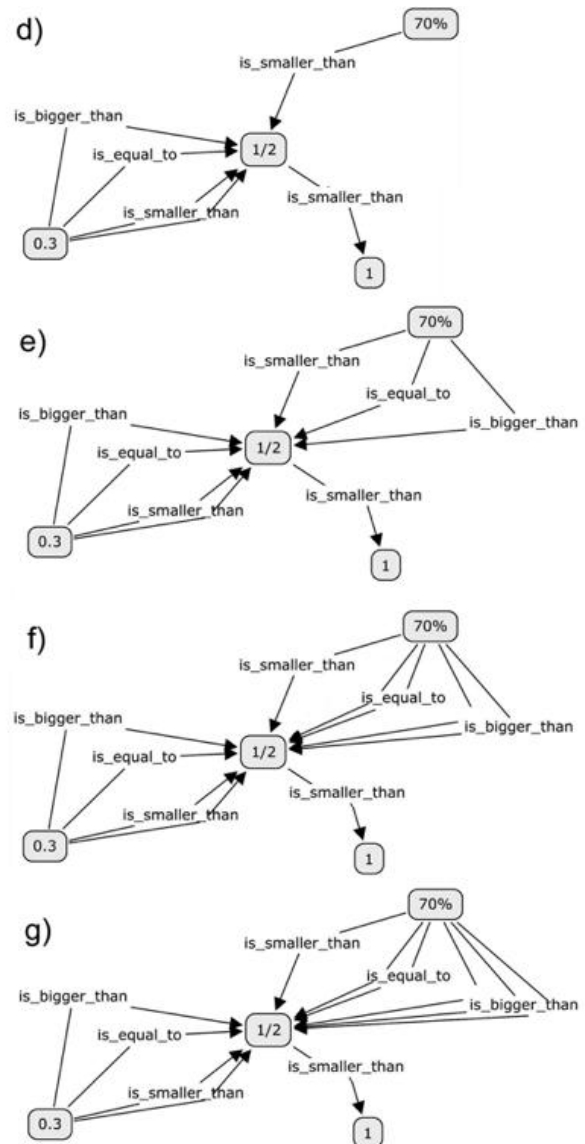

Fig. 2. Semantic network and its development during the teaching phases, part 2.

In the third teaching phase a player forms a question in another way and asks "is 0.3 equal to ½?". Again, we know the statement is false. The agent can guess that statement is either "true" according to an "is_equal_to" relation or "false" according to a "is_smaller_than" relation. The agent guesses "false". When the player evaluates the answer as "correct",

the agent determines that correct answer must be either "0.3 is smaller than ½" or "0.3 is greater than ½". After adding relations into conceptual structure, the agent knows that the correct answer is "0.3 is smaller than ½" because it is the mode (average) relation (Fig. 1 c).

In the fourth teaching phase the player asks, "Is 70% smaller than ½?" and on purpose, s/he teaches it the wrong way. The agent guesses that the statement is "true" and the player evaluates the answer as "Correct", which forms an "is_smaller_than" relation in the conceptual structure (Fig. 2 d).

In the fifth teaching phase the player starts to correct the conceptual structure. S/He asks again," Is 70% smaller than ½?". According to previous teaching, the agent knows that the answer is "true". Because the player now knows that it is incorrect answer, the player evaluates it as "incorrect". In this case the agent determines, that 70% must be equal to ½ or 70% must be greater than ½. After adding relations, the conceptual structure has all the possible comparing statements (Fig. 2 e) and basically behaves like an empty structure.

In the sixth teaching phase, the player asks for the third time, "Is 70% smaller than ½?". Because there is no strongest relation, the agent guesses "true". The player evaluates it again as "incorrect". Again, the agent determines, that 70% must be equal to ½ or 70% must be greater than ½ and adds those relations to the conceptual structure (Fig. 2 f).

In the seventh teaching phase, the player decides to change the question to, "Is 70% more than ½?". The agent guesses "True", because 'is_equal' and 'is_greater_than' do contain the same probability. The player confirms that the answer was correct and one more "is_greater_than" relation was added into the conceptual structure (Fig. 2 g). After that the agent knows that the correct answer is "70% is greater than ½", because such a set of relations are the strongest.

While playing, the conceptual structure will grow to thousands of relations and a single teaching phase only has a limited effect on the areas of the conceptual structure already taught. Understanding this phenomenon is valuable when trying to correct a wrongly taught part of the conceptual structure. Naturally, wrong teaching could be corrected by teaching the correct structure enough times. The game AI uses all the taught information to back its decisions, and therefore it takes time to override the wrong relations in the agent's conceptual structure.

IV. IMPLEMENTATION

The gaming approach is learner centric: mathematics are introduced in a way that learner can build his/her mental conceptual structures by adding new concepts into known ones. The game characters learn like humans do: inductively case-by-case by building relations between new and existing concepts. When the player is responsible for character's mental development, he/she records also his/her mental conceptual structure during the gameplay. Eventually, we can say that while teaching his/her virtual character, learner reproduces a conceptual network about his/her mental conceptual structures.

One important phase in game design is designing the game

play in a way that it is very easy to start to play the game, but the game remains interesting and challenging after hours of play. In our design, we designed an user interface (UI) that is very easy to understand, but in our first user experience (UX) studies we noticed that many user could play the game but did not understand the story behind the game and so the final goal of the game was not clear for them. To make the story more visible, we designed comic strips that introduces the next steps and their relations to the story. Furthermore, the instructions for new interactions was given as comics (example in Fig. 3). The comics was a good solution: pupils seem to understand more on gameplay, but they also like the comics. Some kids draw also their own comic strips, which gives and idea for further research: what is the impact of user generated math comics in terms of learning and motivation.

Fig. 3. Introduction to specific gameplay.

To give pupils a feeling that characters are really their pets, there are several mouses and cats to choose. If player chooses a mouse, it is expected that it is taught correctly. In case player wants to help mice, he/she can start teaching cats wrongly. The idea in teaching wrongly is simple: many pupils are afraid of showing his/her weak skills. When the aim is to fail, it do not matter if you some times teach correctly. Eventually, before you can teach wrongly, you have to know correct answers, and so the process is same as when teaching correctly.

In different classrooms and locations (example in Fig. 4) teacher asks questions from mouses and cats. Player can help her/his pet by pointing the pet's thoughts. The pet learns exactly according the teaching. If player teaches correctly, pet learns correctly and vice versa. After player has taught enough conceptual relations for his/her pet, a challenge icon appears into screen. By clicking the icon, player sends her/his pet into labyrinth to compete against the cat. In labyrinth the pet is on it's own and player's task is to observe how it manages.

In classroom, the skills -tool (Fig. 5) is meant for parents or teachers to quickly observe what learner has taught for his/her pet. The visualization shows correctly taught concepts in the upper part of the skills -area with green bars and wrongly taught concepts in the lower part of the area with red bars. The length of the bar represents the quantity of teaching with the concept. Concepts that has not been taught do not

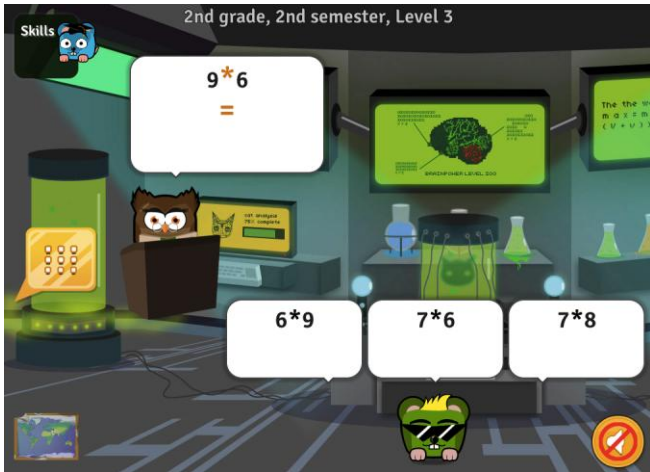appear in the skills -area.
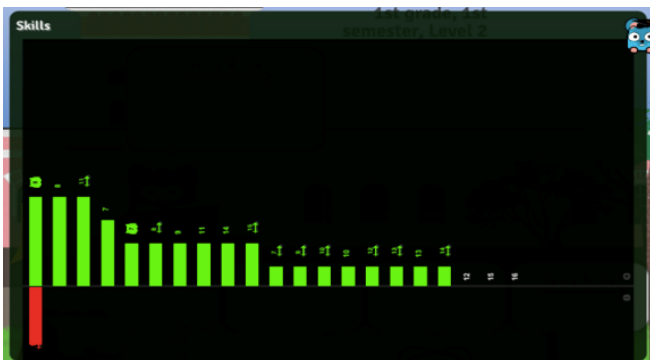

Fig. 4. An example about classrooms.

Fig. 5. Skills -areas. upper green bars represents correct teaching, lower red bars represents with wrong teaching.

Furthermore, when summarizing the game achievements, schools and national level policy makers can receive analysis about competences and skills in general level in order to develop their teaching or formal curriculum. The analytics are that detailed that we can point out general bottlenecks of education: e.g. in Finland there is an interesting bottleneck related to fraction numbers with odd nominator. These numbers mediates or connects nearly all difficulties related to converting numbers between decimal numbers, fraction numbers and percent numbers. In other words, in Finland we should pay attention on how to teach odd nominated numbers.

Fig. 6. Example about labyrinths the virtual characters are conquering.

In the labyrinth (Fig. 6), both characters pic's the doors

according to their taught knowledge. During the labyrinth player can observe what to teach more. If mouse wins, the level is completed and next level becomes playable. The achievements are auto-saved and all completed levels remains playable - of course player wants best possible badge (Fig. 7).

Fig. 7. Levels and earned badges.

Finally, the game characters can compete against any other taught character any time, no matter if the human player is online, because all the taught behavior and skills in characters' AIs' are always available in online. The AI architecture enables access to all game characters even in unfamiliar or untaught context for the character, because of semantic network based approach (Fig. 8). The game as well as content will evolve all the time, so players are not expected to get bored after played game once.
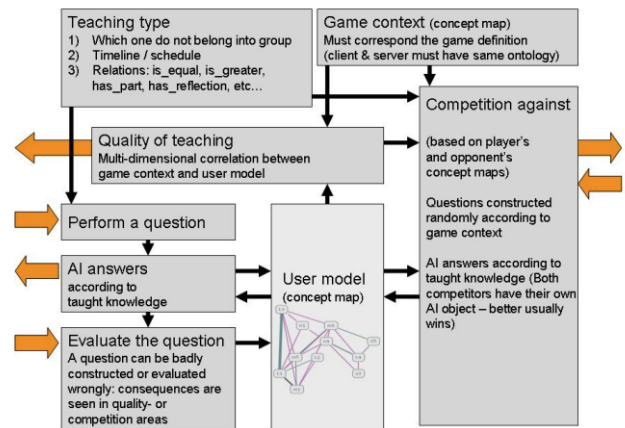

Fig. 8. AI architecture overview.

Furthermore, the production technology has been designed in a way that enables very cost effective production. All assets are reusable and easily updated. Adding new exercises or new levels can be done in hours. Designing a whole new grade takes only few weeks if there is no need for new graphics. Graphics design, as design, we can not speed up.

Technically eedu elements is an online game with client-server architecture. Elements game clients are optimized for tablets and smart phones. Technically they are HTML5 clients and apps for iOS, Android and Windows 8. The challenges related HTML5 development are mostly related on the maturity of HTML5: some graphics functions are not supporting hardware acceleration, all the UI components are not standardized and, in some cases, crossdomain challenges can't be avoided. However, HTML5 is already very powerful and cost-effective way to produce user interfaces. In next few years we'll see it's full potential.

Game mechanics run in server side, built in Google AppEngine. The mechanics enables that game characters can compete against any other character any time, no matter if the opponent is really online, because all behavior is always available in online.

Google AppEngine, as well as other big clouds such as EC2 and Azure, provides global access for millions of users. Some years ago a bottleneck for intelligent server-side game AIs' was computational power. Nowadays the bottleneck, with millions of players, is network bandwidth.

V. CONCLUSIONS

According to our studies, users can relatively quickly and easily teach behavior to a game character. In terms of conceptual learning, the developed AI emulates the way people learn: learning is about concepts and their relations. The behavior modeling makes it possible to model conceptual learning and thus uncover the frequencies, dependencies and patterns behind conceptual change and learning transfer. These results show the strengths of sharing behaviors: without capabilities of sharing the behavior, the kids wouldn't spend that much time on school disciplines. On the other hand, if kids like the idea of sharing behaviors in an educational game, they would definitely love it in an entertaining game.

In the near future user generated behaviors can be developed and shared as all other user-generated content. Furthermore, game developers can design interfaces that enable users to teach versatile behaviors. User-generated behaviors can, e.g. replace AI controlled opponents or extend player's own team. Taught behavior model could be shared on the web. Games and/or developers can upload user-generated behaviors either as AI updates and extensions, or in a development phase. From a game consumer point of view, the most interesting part is in developing behaviors, sharing them and finally playing with them, or against them.

REFERENCES

[1] P. Cowling, "Writing AI as Sport," in S. Rabin, ed., *AI Game Programming Wisdom 3*, Massachusettes: Charles river Media Inc., pp. 89-96, 2006.

[2] C. Baekkelund, "Academic AI Research and Relations with the Game Industry," in S. Rabin, ed., *AI Game Programming Wisdom 3*, Massachusettes: Charles river Media Inc., pp. 77-88, 2006.

[3] L. Liden, "Artificial Stupidity: The Art of Intentional Mistakes," in S. Rabin, ed., *AI Game Programming Wisdom II*, Massachusetts: Charles River Media, Inc., 2003.

[4] H. B. Amor, J. Murray, and O. Obst, "Fast, Neat and Under Control: Arbitraiting Between Steering Behaviors," in S. Rabin, ed., *AI Game Programming Wisdom 3*, Massachusettes: Charles river Media Inc., pp. 221-232, 2006.

[5] M. Bowling, J. Furnkranz, T. Graepel, and R. Musick, "Machine learning and games," *Machine Learning,* vol. 63, pp. 211-215, 2006.

[6] P. Brusilovsky, "Adaptive Hypermedia," *User Modeling and User-Adapted Interaction,* vol. 11, pp. 87-110, 2001.

[7] R. Houlette, "Player Modeling for Adaptive Games," in S. Rabin, ed., *AI Game Programming Wisdom II*, Massachusetts: Charles River Media, Inc., 2003.

[8] S. Kumar and N. E. Mastorakis, "Novel Models for Multi-Agent Negotiation based Semantic Web Service Composition," *WSEAS Transactions on Computers*, vol. 9, no. 4, pp. 339-350, 2010.

[9] C. I. Popirlan, "Knowledge Processing in Contact Centers using a Multi-Agent Architecture," *WSEAS Transactions on Computers*, vol. 9, no. 11, pp. 1318-1327, 2010.

[10] R. Mayer, "Should there be a three-strikes rule against pure discovery learning?" *American Psychologist*, vol. 59, pp. 14-19, 2004.

[11] H. Ketamo, "User Generated AI's for sport games: Framework with semantic networks," in J. Wiemeyer and S. Göbel, eds., in *Proc. GameDays on Serious Games for Sports and Health*, 25-26 March 2010, Darmstadt, Germany. pp. 77-90, 2010.

[12] H. Ketamo and K. Kiili, "Conceptual change takes time: Game based learning cannot be only supplementary amusement," *Journal of Educational Multimedia and Hypermedia,* vol. 19, no. 4, pp. 399-419, 2010.

[13] H. Ketamo and M. Suominen, "Learning-by-Teaching in an Educational Game: The Educational Outcome, User Experience and Social Networks," *Journal of Interactive Learning Research,* vol. 21, no. 1, pp. 75-94, 2010.

[14] H. Ketamo, "Semantic networks -based teachable agents in an educational game," *Transactions on Computers*, vol. 8, no. 4, pp. 641-650, 2009.

[15] H. Ketamo, "Sharing Behaviors in Games," in *Proc. the 5th European Computing Conference*, April 28-30, Paris, France, pp. 120-125, 2011.

[16] H. Ketamo, "Sharing Behaviors in Games and Social Media," *International Journal of Applied Mathematics and Informatics*, vol. 5, no. 1, pp. 224-232, 2011.

Harri Ketamo is a CEO and founder at EEDU ltd. with Ph.D. And he is an adjunct professor at Tampere University of Technology. He have a strong academic background in Educational Technology, Complex Adaptive Systems, Cognitive Psychology of Learning, Neural Computing and Game Design. Before founding eedu, Harri has been e.g. a Director of Education at Satakunta University of Applied Sciences and Academy of Finland granted post-doc researcher.