Semantic representation of action games

Costas Boletsis, Dimitra Chasanidou, Panagiotis Pandis, and Katia Lida Kermanidis

Dept. of Informatics, Ionian University, Kerkyra 49100, Greece
{c10bole,c10chas,c10pand,kerman}@ionio.gr

Abstract. Player modeling is a crucial procedure for the user-oriented videogame design. For player modeling to be achieved, modeling domain knowledge is essential. Mastering the semantics of a domain is to learn the "language" of the domain. In the present study, two approaches to representing "words" are considered and analyzed comparatively. The first approach ("grid" representation) is based on the division of the game terrain as a grid, capturing in each cell the content-based information of the action game. The second approach ("holistic" representation) captures the contextual information in which the action takes place (life, score, shield, number of hit asteroids etc.). For the present comparative analysis, we use the action videogame SpaceDebris. We analyze the data using the classification algorithms J48, Naive Bayes, and SMO, as well as the K-means clustering and we compare the results in an attempt to identify the approach that represents the semantic space more reliably. The data acquired from the"grid" representation perform better, however the low value of the performance's difference does not allow us to come to rock solid conclusions.

Keywords: action games; gaming styles; player modeling; semantic representation;

1 Introduction

Player modeling is a crucial procedure for the user-oriented videogame design. For player modeling to be achieved, modeling domain knowledge is essential [11]. Mastering the semantics of a domain is to learn the "language" of the domain, i.e. to become exposed to various sequences of concepts that carry units of meaning related to the domain (domain lexemes or "words"), in numerous contexts [13, 11]. There are two possible ways for supplying domain knowledge [23]: by hand, making use of domain experts know-how, and automatically, by deriving the semantics from large corpora of "word" sequences [11]. The first approach is more accurate, but domain-dependent, while the second is useful when no hand-crafted knowledge is available.

Action games have properties that resemble those of complex dynamic environments: causality relations (actions or decisions often affect subsequent actions or decisions), time dependence (the environmental circumstances that affect actions and decisions vary over time), and latent, implicit relations between domain properties that are not straightforward [11]. Identifying the domain vocabulary, as well as well-formed sequences of "words" that constitute complete descriptions of actions or context conditions is of significant research interest [11].

In the present study, two approaches to representing "words" are considered and analyzed comparatively. The first approach (henceforth called "grid" approach or representation) is based on the division of the game terrain as a grid, capturing in each cell the content-based information of the action game [11]. The second approach (henceforth called "holistic" approach or representation) captures the contextual information in which the action takes place (life, score, shield, number of hit asteroids etc.) [11].

For the present comparative analysis, we use the action videogame SpaceDebris [1] and we implement the two aforementioned approaches. Consecutively, we analyze the data using various classifiers, as well as, clustering and we compare the results in an attempt to identify the approach that represents the semantic space more reliably.

1.1 Contribution & Paper Organisation

The contribution of this work is summarized as follows:

- identify and implement the vocabulary of the game domain,
- identify a semantic representation approach that optimizes performance for player modeling purposes, making use of the theoretical background provided in [11].

The rest of the paper is organised as follows. Section 2 describes background and related work, Section 3 provides a thorough presentation of the videogame that is used for experimental purposes. Section 4 provides a complete account of the two approaches of representing semantic space. Next, Section 5 describes the algorithms used herein. Subsequently, Section 6 presents and discusses the experimentation and results obtained, while the paper is concluded in Section 7.

2 Related Work

The present paper is based on the work of Kermanidis and Anagnostou [11], where an experimental implementation of LSA in data derived from the "grid" and "holistic" approaches, is theoretically described. In our work, we implemented and tested experimentally the theoretical work of [11] focusing on the authors' work on semantic representation.

Semantics is generally defined as the study of meaning and relations among signs [6]. Driel and Bidarra, in [6], specified game semantics that are concerned with the structure and meaning of game elements, such as level geometry and player characters. They, also, described how the semantics of game elements deals with the multiple domains of game development and how different building blocks of the game are understood in each domain [6]. Tutenel et al [21] distinguished levels of semantics, such as object level, relationships between different objects and global semantics, as for example time.

Defining the semantics of a video/computer game domain is often applied for the purposes of player modeling. Lately, this domain has attracted the interest of the research community, although it is a relatively new area. Existing work [2, 20, 22] demonstrates that the power of player's models can be utilized for improving entire game or in-game situations, high interest gaming levels and players' satisfaction. In a publication on adaptive game design, Charles et al. present a method for player modeling based on profiling [2], a technique whereby a set of characteristics of a particular class of person is inferred from past experience [3]. A study from Drachen et al [5] focuses on constructing models of players for the commercial game Tomb Raider. The unsupervised learning approach utilized reveals four types of players which are analyzed within the context of the game.

Various algorithms are used for expliciting the semantic relations of a video game. Clustering gaming styles of users is mentioned by research in bayesian networks [19] and self-organizing maps [4], which have been used for clustering player's waypoints laying a simple level exploration game. Finally, Yannakakis and Maragoudakis [22] in their work used naive bayesian models for prediction of player actions in previously unseen world states.

3 SpaceDebris

Fig. 1. The action videogame SpaceDebris [1].

The videogame that is used for the collection of data that represent semantic space is a modified version of the action game SpaceDebris. SpaceDebris is a Space Invaders-like action game by Anagnostou and Maragoudakis [1], which takes place within the borders of the screen (Fig. 1). The game concerns space battles, with the player trying to destroy as many enemy spaceships as possible with his laser gun, and survive. There are two types of enemies: the carrier spaceship, which is slow and is more resistant to the laser blasts of the player's spaceship (2-3 laser shots), and the fighter spaceship, which is smaller, faster and much easier to be destroyed (1 laser shot). In the game environment, there are floating asteroids which work in favour of the player, since the player has the opportunity to alter the asteroid's direction, leading it to an enemy spaceship and, finally, destroying it. The alteration of an asteroid's direction is suceeded by firing at him or by using extra weapons: the blast, which pushes further away all the asteroids, the freezer, which freezes the asteroids that are in the proximity of the player and the grabber, which attracts the nearest asteroids and it is used in combination with the blast. Shield and life power-ups are, also, floating in the game environment and the player must fire at them in order to acquire them.

4 Semantic Representation Approaches

The vocabulary identification of this study is based on the work of Kermanidis and Anagnostou, as described in [11]. Each "word" or lexeme of the domain vocabulary can be represented in two ways.

4.1 Grid representation

Firstly, in the grid approach, the "word" consists of two parts. The first one is derived from the consideration of the game terrain as a grid of 11x8 (Fig. 2). Every half a second, an instance is printed on a log file. The instance consists of 88 strings, each one describing the state of the cell at that particular moment by using binary values. The distinct cell states are portrayed in Table 1. Therefore, every 0.5 seconds we get 88 cell states, each one appearing as the example in Table 2.

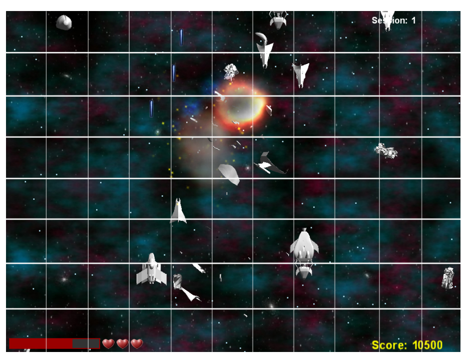

Fig. 2. A graphical representation of the game terrain as a grid of 11x8.

Of course, there can be more than one states in the same cell, meaning that there can be more than one true binary attribute (as in Table 2). Consecutively, for a 10-minute session, we get approximately 1200 instances of 88 cells.

The second part of the "word" models further out-of-the-grid (non-spatial) information, like result, score, number of available life upgrades, number of available shield upgrades.

The "grid" approach takes into account long-distance semantic dependencies, i.e. the semantics of each cell (no matter how distant) participates in the domain knowledge and it "mines" the causality relations between the environment and the player's reaction to it implicitly [11].

Table 1. The total number of distinct cell states.

Distinct cell states
1) The cell contains an asteroid
2) The cell contains an "energized" asteroid
3) The cell contains the player's ship
4) The cell contains the player's ship being hit by enemy 1
5) The cell contains the player's ship being hit by enemy 2
6) The cell contains the player's ship being destroyed
7) The cell contains the player's ship firing a laser
8) The cell contains enemy 1
9) The cell contains enemy 1 being hit by a laser
10) The cell contains enemy 1 being hit by an asteroid
11) The cell contains enemy 1 firing a laser
12) The cell contains enemy 1 being destroyed
13) The cell contains enemy 2
14) The cell contains enemy 2 being hit by a laser
15) The cell contains enemy 2 being hit by an asteroid
16) The cell contains enemy 2 firing a laser
17) The cell contains enemy 2 being destroyed
18) The cell contains a player laser
19) The cell contains an enemy 1 laser
20) The cell contains an enemy 2 laser
21) The cell contains a life upgrade
22) The cell contains a life upgrade hit by laser
23) The cell contains a shield upgrade
24) The cell contains a shield upgrade hit by laser
25) Empty cell

Table 2. A distinct cell state stating that, in this particular cell, there is a player laser and a life upgrade.

Grid approach string example
0_0_0_0_0_0_0_0_0_0_0_0_0_0_0_0_0_1_0_0_1_0_0_0_0

4.2 Holistic representation

In the second approach, the “word” consists completely of non-spatial information, like score, number of available life upgrades, number of available shield upgrades etc. (Table 3). These features are also printed every 0.5 seconds on a log file. The values of these features form a string, constituting a complete “word” (Table 4).

Table 3. The total number of context features.

Context features
1) The number of enemies close to the player (denoting danger)
2) The total number of enemies on screen
3) The number of player lasers fired
4) The number of enemy 1 lasers fired
5) The number of enemy 2 lasers fired
6) The position of the player
7) The number of life upgrades performed
8) The number of shield upgrades performed
9) The number of hit asteroids
10) The number of visible asteroids
11) The number of hit enemy 1 ships
12) The number of hit enemy 2 ships
13) The score value
14) The number of the player’s available life upgrades
15) The number of shields available to the player

Table 4. Contextual information about lives (3), shield (80%), position X (-100), position Y (-300) and other features.

Holistic approach string example
1_0_3_80_-100_-300_2_0_0_0_0_0_0_0_2_2_8_4_0_6

The second approach to defining the vocabulary using context information is more “holistic” [11]. Each “word” represents a player action, like *move to a location* or *fire*, and it is accompanied by a concatenation of features that represent the state of the context in which the action took place. Therefore the reasoning behind the player’s actions (causality relations) are clearly identifiable [11].

5 Algorithms

For our experiment's needs, we implemented the following algorithms:

- J48 [16, 14, 17]
- Naive Bayes [18, 8, 7, 12]
- Sequential Minimal Optimization (SMO) [15, 10]
- K-means clustering [9]

The choice of these specific algorithms was based on the algorithms used in past related works (Section 2).

6 Performance Evaluation

In this section, we experimentally compare the grid and the holistic approach on representing the semanting space. We initially describe the experimental setup, then present the results and finally provide a short discussion.

6.1 Experimental Setup

For the purposes of performance evaluation of the alternative methods to represent semantic space, we accumulated two datasets from applying the two representation approaches on SpaceDebris. The first dataset, henceforth titled *dataset A*, is comprised of data acquired by implementing the grid approach, as described in Sec. 4.1. Thus, the second dataset, henceforth titled *dataset B*, is comprised of data acquired by implementing the holistic approach (Sec. 4.2). The aim of the experimental setup is to evaluate two approaches on semantic representation using widely-used algorithms for classification and clustering. A top-down approach in the experimental setup is adopted. Firstly, we categorize the players by their gaming styles. We defined four gaming styles categories [11]: *novice* - a player with little gaming experience and playing SpaceDebris without any particular style, mostly losing, *tactical* - a player keen on playing strategy or adventure games and when playing SpaceDebris makes wise use of the laser and power-ups, *aggressive* - a player keen on action games and when playing SpaceDebris fires constantly without frequent use of the power-ups, and *defensive* - a player keen on puzzle and internet games and when playing SpaceDebris does not fire or tries to avoid the enemies in order not to be killed. Following the player's categorization, several game sessions are conducted in order to collect data and feedback. Consecutively, we applied the aforementioned algorithms to the extracted data, aiming to classify and cluster the players' gaming styles and, finally, evaluate the two competitive approaches. Namely, the learning examples are single state-action pairs with a player classification as target.

Participants Ten users between 20 and 30 years of age participated in our experiment. The participants were selected randomly, having various and different gaming backgrounds.

Apparatus The "tool" used for our experiment was a modified version of the action game SpaceDebris [1]. The game was developed for PC by Anagnostou and Maragoudakis using the C# programming language and its source code was modified in order to gather all the necessary data in log files.

Procedure Each participant had a 5-minute trial playing SpaceDebris. Then, he/she was given a short questionnaire consisting of general gaming questions (about gaming experience, preferences etc.) and specific questions about SpaceDebris, trying to categorize the gaming style of the player, according to four categories: novice, tactical, aggressive, and defensive [11]. Afterwards, the participant played a 10-minute gaming session, with the presence of a domain expert in order to witness the player's gaming style and accept or dispute the questionnaire's categorization. When the 10-minute session was over the data was collected and stored.

Data Analysis For the evaluation of the approaches concerning the semantic representation of the action game SpaceDebris, we implemented the following classification algorithms: J48, Naive Bayes, and SMO. Furthermore, we implemented the K-means clustering algorithm. The software used was the Waikato Environment for Knowledge Analysis (WEKA), developed at the University of Waikato, New Zealand.

6.2 Experimental Results

The domain expert did not strongly disagree with any of the questionnaire's categorization but she was responsible for recognizing the thin line that separates each gaming style from the other and provide a fair categorization according to the set criteria. From the evaluation of each player's gaming style, we extracted the following results: 2 out of 10 participants were categorized as novice, 4 as tactical, 2 as defensive and 2 as aggressive. After the formation of our datasets the applications of the aforementioned algorithms took place as 10-fold cross-validations, leading to the results of Table 5.

Taking into consideration the F-measure, we examined the results of Table 5. As can be seen, data acquired from the "holistic" representation perform better when analyzed with J48, whereas Naive Bayes and SMO favor dataset A (data derived from the "grid" representation). Even though, there are no significant differences in the total F-measure, dataset A presents better performance. K-means clustering (Table 6) presents a difference of approximately 5% in favour of dataset A.

Analyzing the performance of the two datasets per gaming style, we extracted the results of Fig. 3, according to the algorithm used. From the graphical representation of the results, we witness a uniform distribution of the performance, as far as the defensive and tactical gaming style are concerned (approximately 0.7 F-measure). An interesting observation is the poor performance of the novice gaming style class, for all datasets and algorithms, despite the fact that it shares

Table 5. Application of the J48, Naive Bayes, and SMO classifiers on datasets A & B.

Accuracy (Weighted Avg.)

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Classifier
Dataset A (Grid)	0.691	0.134	0.686	0.691	0.68	0.848	J48
Dataset B (Holistic)	0.724	0.109	0.724	0.724	0.723	0.863	J48

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Classifier
Dataset A (Grid)	0.741	0.104	0.741	0.741	0.737	0.918	NaiveBayes
Dataset B (Holistic)	0.669	0.114	0.686	0.669	0.671	0.869	NaiveBayes

	TP Rate	FP Rate	Precision	Recall	F-Measure	ROC Area	Classifier
Dataset A (Grid)	0.76	0.1	0.758	0.76	0.758	0.893	SMO
Dataset B (Holistic)	0.71	0.113	0.712	0.71	0.707	0.864	SMO

Table 6. Application of the K-means clustering algorithm on datasets A & B.

Incorrectly clustered instances

Dataset A (Grid)	4386.0 (41.6287%)
Dataset B (Holistic)	4937.0 (46.8584%)

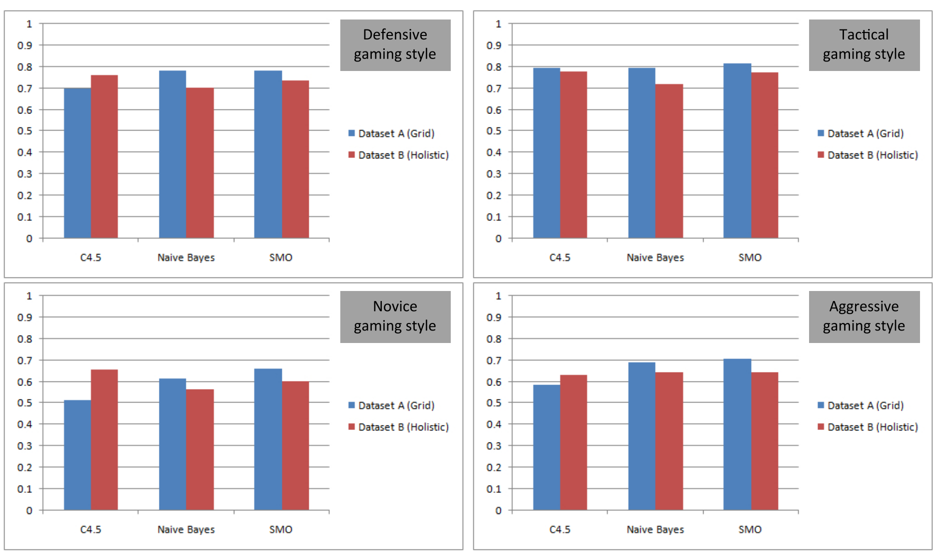

Fig. 3. The F-measure per gaming style.

the same number of training datasets as the defensive and the aggressive gaming style. The characteristics of the novice gaming style was the unpatterned, even unstable, gameplay and the constant losing. We, then, conclude that the data, coming from both the "grid" and the "holistic" representation, of a novice player might lack in quality, because of the repeatitive and unpatterned nature of the gaming style. Overall, we witness the better performance of dataset A (the "grid" representation) in every gaming style, using all the algorithms, apart from J48. J48 may not be the proper algorithm for classifying "grid" data, since the decision tree characteristics combined with the long-distance semantic dependencies of the "grid" approach data may cause conflicts. J48 is not sophisticated enough to detect the underlying semantic relationships that govern the "lower-level", "more basic" words of the grid approach, compared to the higher-level semantics of the vocabulary of the holistic approach. Although the decision tree follows a natural course of events by tracing relationships between events, it may not be possible to plan for all contingencies that arise from a decision, and such oversights can lead to bad decisions.

6.3 Discussion

The presented performance evaluation results can be summarised as follows:

- Using specific algorithms, our datasets, obtained from two different approaches on representing the semantic space, fail to present statistically significant differences. However, there were differences on each algorithm's performance on our datasets and the overall "picture" leads us to the conclusion that data acquired from the "grid" representation might perform better for the purpose of player modeling, without this conclusion to be rock solid.
- According to the used algorithms both datasets present an F-measure close to 70% and, specifically, SMO presents a high performance on data acquired from the "grid" representation. The same dataset is preferable when clustering using K-means. SMO, being capable of coping with high-dimensional data more efficiently than decision trees or Naive Bayes, can capture the underlying semantics of the vocabulary of the grid approach without being affected by the large number of attributes.
- An unpatterned, uncertain way of playing, such as the novice one, may affect the semantic representation of the game.

7 Conclusion

Semantic representation is a research area that is of great importance for the purposes of player modeling. As far as, action games are concerned, there are expressed two competitive approaches [11]: content vs. context. The content-based information derives from a "grid" representation of the game terrain. The contextual information is acquired by logging non-spatial data of the game.

In this work, we examined the accuracy of the most widely used for classification and clustering in two datasets, acquired using the "grid" and the "holistic" approach. Experimental results indicate the superiority of the "grid" representation, revealing the importance of the long-distance semantic dependencies when representing the semantic space. Nevertheless, the value of the significant difference between the performance of the two datasets was not high, in order to come to rock solid conclusions.

Future research directions include the examination of more classification and clustering algorithms, in search of statistically significant differences, as well as, implementing both approaches for experimental player modeling purposes.

References

1. Anagnostou, K., Maragoudakis, M.: Data mining for player modeling in videogames. Proceedings of the 2009 13th Panhellenic Conference on Informatics pp. 30–34 (2009)
2. Charles, D., Black, M.: Dynamic Player Modelling: A Framework for Player-Centered Digital Games, pp. 29–35. The University of Wolverhampton (2004)
3. Clarke, R.: Profiling: A hidden challenge to the regulation of data surveillance. Journal of Law and Information Science 4, 403 (1993)
4. Commercial, B.B.F.: Combining self organizing maps and multilayer perceptrons to learn (2003)
5. Drachen, A., Canossa, A., Yannakakis, G.N.: Player modeling using self-organization in Tomb Raider: Underworld, pp. 1–8 (2009)
6. Driel, L., Bidarra, R.: A semantic navigation model for video games. In: Proceedings of the 2nd International Workshop on Motion in Games. pp. 146–157 (2009)
7. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern classification and scene analysis. Wiley New York (1973)
8. Hsu, C.C., Huang, Y.P., Chang, K.W.: Extended naive bayes classifier for mixed data. Expert Syst. Appl. 35, 1080–1083 (2008)
9. Kalyani, S., Swarup, K.S.: Particle Swarm Optimization based K-means Clustering Approach for Security Assessment in Power Systems. Expert Systems with Applications In Press, Accepted Manuscript (2011)
10. Keerthi, S.S., Shevade, S.K., Bhattacharyya, C., Murthy, K.R.K.: Improvements to platt's smo algorithm for svm classifier design. Neural Comput. 13, 637–649 (2001)
11. Kermanidis, K.L., Anagnostou, K.: LSA-based semantic representation of action games. In: International Conference on Knowledge Engineering and Ontology Development (2010), http://www.keod.ic3k.org/Abstracts/2010/KEOD_2010_Abstracts.htm
12. Langley, P., Iba, W., Thompson, K.: An analysis of bayesian classifiers. In: Proceedings of the Tenth National Conference on Artificial Intelligence. pp. 223–228 (1992)
13. Lemaire, B.: Models of high-dimensional semantic spaces. In: 4th International Workshop on Multistrategy Learning (MSL) (1998)
14. Mitchell, T.M.: Machine learning. MacGraw-Hill (1997)
15. Platt, J.C.: Fast training of support vector machines using sequential minimal optimization. Advances in Kernel Methods 12, 185–208 (1999)
16. Polat, K., Güneş, S.: A novel hybrid intelligent method based on c4.5 decision tree classifier and one-against-all approach for multi-class classification problems. Expert Syst. Appl. 36, 1587–1592 (2009)
17. Quinlan, J.R.: Induction of decision trees. Machine Learning 1, 81–106 (1986)
18. Tang, Y., Xu, Y.: Application of fuzzy naive bayes and a real-valued genetic algorithm in identification of fuzzy model. Information Sciences 169(3-4), 205–226 (2005)
19. Thurau, C., Paczian, T., Bauckhage, C.: Is Bayesian Imitation Learning the Route to Believable Gamebots?, pp. 3–9. No. August (2005)
20. Togelius, J., De Nardi, R., Lucas, S.M.: Making racing fun through player modeling and track evolution. Optimizing Player Satisfaction in Computer and Physical Games p. 61 (2006)

21. Tutenel, T., Bidarra, R., Smelik, R.M., Kraker, K.J.D.: The role of semantics in games and simulations. Comput. Entertain. 6, 57:1–57:35 (2008)
22. Yannakakis, G.N., Maragoudakis, M.: Player modeling impact on player's entertainment in computer games. User Modeling 2005 3538, 74–78 (2005)
23. Zampa, V., Lemaire, B.: Latent semantic analysis for user modeling. Journal of Intelligent Information Systems 18, 15–30 (2002)