

Modeling Action Game Domains Using Latent Semantic Analysis

Katia Lida Kermanidis
 Department of Informatics
 Ionian University
 Corfu, Greece
 kerman@ionio.gr

Kostas Anagnostou
 Department of Informatics
 Ionian University
 Corfu, Greece
 kostasan@ionio.gr

Abstract—Player modeling has attracted the interest of game designers recently, as a personalized game offers more satisfaction. In this paper we propose modeling the semantic space of the action game SpaceDebris, in order to identify semantic similarities between players. To this end we employ Latent Semantic Analysis and attempt to identify latent underlying semantic information governing the various gaming styles. The several challenging research issues that arise when attempting to apply Latent Semantic Analysis to non-textual data, that describe a complex dynamic problem space, are addressed, and the framework of the experimental setup is described.

player modeling, action games, latent semantic analysis, knowledge representation, semantic similarity

I. INTRODUCTION

Representing the knowledge of a specific domain, i.e. identifying the concepts that carry units of meaning related to it (domain “words”), as well as the semantic relations governing those concepts, is a wide and popular research area. Modeling domain knowledge is essential for developing expert systems, for intelligent prediction and decision making, for intelligent tutoring, user modeling, complex problem solving, reasoning etc. Mastering the semantics of a domain is to learn the “language” of the domain [12], i.e. to become exposed to various sequences of domain “words” in numerous contexts. This is similar to the way a foreign language learner learns vocabulary usage by reading, listening to, and writing texts in that language.

There are two possible ways for supplying domain knowledge [12]: by hand, making use of domain experts’ know-how, and automatically, by deriving the semantics from large corpora of “word” sequences. The first approach is more accurate, but domain-dependent, while the second is useful when no hand-crafted knowledge is available.

A widely used method for representing domain knowledge by statistical analysis of word usage is Latent Semantic Analysis (LSA). LSA is adopted from the field of Information Retrieval [11] and improves retrieval performance by taking into account automatically detected polysemy and synonymy relations between words. LSA

identifies these underlying semantic relations by exploiting the occurrence statistics of the words throughout the document collection: by reducing the dimensionality of the initial term-document matrix¹, hidden semantic similarities between words, between documents, and between words and documents surface, linking together words that may not even appear in the same document, or documents that may not share any common words.

LSA has been applied with significant success to other domains, like essay assessment in language learning [8], intelligent tutoring [7], text cohesion measurement [13], summary evaluation [18], text categorization [14]. Although all previously mentioned LSA applications have been performed on text corpora, some approaches have proposed its use in different non-textual knowledge domains like board game player modeling [22], complex problem solving [15], gene function prediction [3][4][5], web navigation behavior prediction [21], collaborative filtering [10], semantic description of images [2].

The present work proposes the application of LSA to a new domain, namely digital action games, in order to identify similarities among the playing techniques of various players. Thereby, the players’ profile can be constructed, and games can be adapted to individual players’ needs and preferences, offering more satisfaction.

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. 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.

Throughout the remainder of the paper we will address the research challenges that emerge when attempting to represent the semantics governing the SpaceDebris action

¹The matrix with rows representing index terms and columns representing documents, and each cell contains the number of occurrences of a term in a document.

shooting game [1]. The proposed use of the representation is player modeling: unsupervised grouping of players with similar gaming manners. Section 2 provides a bibliographic review of player modeling and categorization. Section 3 presents the basic properties of Latent Semantic Analysis, section 4 introduces the action game SpaceDebris, and finally section 5 describes the cognitive modeling process of the game domain, as well as its use for modeling players.

II. PLAYER PROFILING

Several game designers have recently been shifting their focus to the player rather than the game itself. Numerous attempts have been made to identify the gaming technique of each player (e.g. (in)experienced, aggressive, tactical, action player), aiming to adapt the game features to his individual preferences and needs. By personalizing the features of the game, the designer hopes to provide increased satisfaction and entertainment.

Player modeling has been achieved within an interactive storytelling game and the use of machine learning techniques [20][16], by estimating the statistical behavior (distribution) of player actions [19], by using graphical knowledge representation schemata like influence diagrams [17] and Bayesian networks [9]. Further references to player modeling can be found in [6]. In [1] SpaceDebris players are grouped into two clusters, using unsupervised learning, according to their playing style (aggressive or tactical).

Unlike previous approaches that either assign one of a set of predefined profiles to a player, or explore explicit actions and decisions made by the player, the present work proposes a knowledge model that attempts to

- identify the vocabulary of the game domain,
- represent complicated game states (action game states are hard to represent, as their definition is not straightforward like in board games), and
- detect hidden, underlying semantic relations between decisions made and actions taken and their context, as well as among domain “words”.

III. LATENT SEMANTIC ANALYSIS

As mentioned earlier, LSA is a mathematical/statistical method initially proposed for reducing the size of the term-document matrix in information retrieval applications, as the number of lexicon entries may reach several thousand, and the document collection may contain tens of thousands of documents or more. LSA achieves dimensionality reduction through Singular Value Decomposition (SVD) of the term-document matrix. SVD decomposes the initial matrix A into a product of three matrices and “transfers” matrix A into a new semantic space:

$$A=TS D^T \quad (1)$$

T is the matrix with rows the lexicon terms, and columns the dimensions of the new semantic space. The columns of D represent the initial documents and its rows the new dimensions, while S is a diagonal matrix containing the singular values of A . Multiplication of the three matrices will

reconstruct the initial matrix. The product can be computed in such a way that the singular values are positioned in S in descending order. The smaller the singular value, the less it affects the product outcome. By maintaining only the first few of the singular values and setting the remaining ones to zero, and calculating the resulting product, the initial matrix may be approximated as a least-squares best fit. The dimensions of the new matrix are reduced and equal to the number of selected singular values.

As an interesting side effect, dimensionality reduction reduces or increases the frequency of words in certain documents, or may even set the occurrence of words to higher than zero for documents that they initially did not appear in. Thereby semantic relations between words and documents are revealed that were not apparent at first (latent). It needs to be noted that LSA is fully automatic, i.e. the latent semantic relations are learned in an unsupervised manner. Another significant property is that LSA does not take into account the ordering of words within their context; documents are considered “bags of words”. Extensive information on LSA can be found in [11].

IV. SPACEDEBRIS

The videogame used for the purposes of data collection is based on SpaceDebris [1]. The action takes place within the confines of a single screen, with alien ships scrolling downwards. There are two types of enemy spaceships, the carrier which is slow and can withstand more laser blasts, and a fighter which is fast and easier to destroy. The player wins when he has successfully withstood the enemy ship waves for a predetermined time. The game environment is littered with floating asteroids which in their default state do not interact (i.e. collide) with any of the game spaceships. In order to do so, an asteroid has to be “energized” (hit by player weapon). Also floating are shield and life power-ups which the user can use to replenish his ship’s shield and remaining lives. The player’s ship is equipped with a laser cannon which she can use to shoot alien ships. The laser canon is weak and about 4-5 successful shots are required to destroy an enemy ship (except for the boss which requires many more). The laser can also be used to “energize” an asteroid and guide it to destroy an enemy ship.

V. MODELING SPACEDEBRIS

Several research challenges need to be addressed when attempting to model the domain of an action game like SpaceDebris using LSA.

A. Vocabulary Identification

In board-like games, like tic-tac-toe or chess, domain “words” are easy to identify. Boards may be viewed as grids of cells and each cell state (e.g. “X”, “O” or empty in tic-tac-toe) constitutes a “word” [12]. In action video games “words” are harder to identify. Should they represent player actions, enemy actions, the state of the context, scoring results, spare lives or ammunition, time parameters? In the firefighting microworld of [15] “words” are actions like appliance moves, or water drops. The definition of a game “word” depends on the intended use of the model. If the

intended use is behavior prediction, a “word” needs to model a player’s action, as the player’s sequence of actions (in a given context) defines his behavior.

In the present work, two approaches to representing “words” are considered. In the first approach, the game terrain is considered a grid, and the concatenation of the states of each cell in the grid constitutes a “word”. The state of each cell is determined by several factors, depending on the state of each game entity. For example a cell might be empty, it might contain an asteroid, it might contain an “energized” asteroid. It might also contain the player’s ship, the player’s ship firing a laser, the player’s ship being hit by a laser. A cell might also be in state that combines a number of states such as those described. Player or enemy actions are modeled implicitly through the related cell states. Further out-of-the-grid (non-spatial) information, like score, spare lives, spare shields, is modeled separately and each of these features is concatenated to the cell states to constitute a complete “word”. The cell size is of importance, as it affects the level of granularity. The smaller the cell size is, the more “generic” the “words” are. We will experiment with grid sizes 11x8 and 12x6, the first corresponding to the player ship’s size and the second to the largest enemy ship size, with a screen resolution of 1024x768 pixels. Vocabulary size using this representation of approximately 24 cell states reaches 2212 with a grid size of 11x8 and 1728 with grid size of 12x6. Vocabulary size is important, as too many “words” may result to too few cooccurrences and LSA will not work. A too small vocabulary may lead to too few similarities and, again, the method will not work [12]. Optimal vocabulary size is an open research issue and depends on the domain.

The second approach is more “holistic” and resembles that of [15]. Each “word” represents a player action, like *move to a location* or *fire*. However, unlike [15], each action in a “word” is accompanied by a concatenation of features that represent the state of the context in which the action took place. These features are

- the number of enemies very close to the player
- the number of enemies close to the player
- the total number of enemies on the screen
- the number of player lasers fired
- the number of enemy lasers fired
- the position of the player
- the number of life and shield upgrades performed
- the number of hit asteroids
- the number of visible asteroids
- the number of hit enemy ships
- the score value
- the number of available life upgrades
- the number of shields available to the player

“Word” examples using an NxM grid (ex. 1) and the “holistic” (ex. 2) approach are shown below. The first part (up to X_{NM}) of the string in ex. 1 consists of tokens, each token stands for one cell state (tokens are concatenated together with underscores). We use 16bit numbers, to denote the presence (1 or 0) of one of the 9 game entities (player, 2 types of enemies, 3 types of lasers, 2 types of upgrades, asteroid). The last three tokens encode out-of-the-grid

information, i.e. the score, the number of spare lives and spare shields respectively. In ex. 2 the first token is the player’s action (the player moves to location with coordinates (-286, -133)). Each of the following concatenated tokens is a value for each of the features listed above (e.g. 1 enemy is very close, 3 are close, there are 9 enemies on-screen, player has fired a laser, enemies have fired 3 lasers etc.).

2_1_0_..._X_{NM}_1000_3_100 (ex. 1)

move-286-133_1_3_9_1_3_..._X (ex. 2)

The “grid” representation takes into account long-distance semantic dependencies, i.e. the semantics of each cell (no matter how distant) participates in the domain knowledge. The “holistic” representation detects causality relations between the environment and the player’s reaction to it in a more straightforward way.

B. Game Session Representation

Game sessions play the role of documents in Information Retrieval. As documents are sequences of words that convey a specific meaning and are considered to satisfy a certain information need, game sessions are well-formed sequences of “words” in the game domain. Each “word” constitutes a complete description of a player’s action or of a description of the context (game environment) at a given moment.

One way to represent a game session is to take a sample of the game state at constant pre-defined time intervals (e.g. every 500 msecs) and register the sequence of “words” (“words” are defined using either the grid or the holistic approach) that describe the sample. Each sample represents a game state at the specified time point. The duration of the sampling time interval is very important. Small intervals may lead to consecutive states that are semantically identical (i.e. the player has not had enough time to make a decision or act, or the state of the context has not changed). Long intervals may lead to the loss of semantic information (i.e. player’s actions that occurred between the samples may be missed). We will experiment with various interval sizes in order to find the “optimal” sampling rate.

Another way to represent game sessions is through sampling events that are dynamically triggered by player’s actions. Every time the player acts, a game state sample is taken, and the player’s action and game context are recorded.

C. Reduction Rate

The rows of the resulting term-document matrix represent the “words”, and the columns represent game sessions. Each cell contains the frequency of occurrence of the “word” in the row in the column session. Applying LSA to the matrix, another research question arises: What is the optimal number of singular values that should be maintained? In Information Retrieval the number of dimensions of the latent semantic space is usually between 100 and 300 [12]. More research work needs to be done in order to determine the appropriate number of dimensions when it comes to non-textual domains. Our proposal includes the experimentation with various dimension

numbers and the research of their impact on modeling performance.

D. Experimental Setup for Measuring Semantic Similarity

As mentioned earlier, the extracted model will be used for identifying similar gaming techniques among players. A group of players will play the game for a given time frame. Players will at first be asked to familiarize themselves with the game by playing off the record for 4-5 minutes. After this introductory phase, game sessions will be recorded for every player. Each game session lasts an average of 3 minutes, and players will be asked to complete a specific number of games. The number of games needed for successfully identifying the player's gaming style will be experimentally explored. Each game session will constitute a feature vector, which is formed by the set of "words" representing it. Feature vectors both before and after LSA will be stored for comparative analysis of results.

To identify similar gaming techniques, the distance between vectors needs to be computed. Though several distance metrics have been experimented with, pairwise cosine similarity is the most popular measure [12]. Cosine similarity will link the most semantically similar vectors together, forming unsupervised clusters of similar gaming techniques. Clustering evaluation may be performed in two ways. Players may be asked to answer a short questionnaire before playing, where they will characterize their individual gaming style, choosing one or more from a set of pre-defined styles. Another way is to ask a game expert to identify the style of each individual player by looking at his actions and decisions throughout the game sessions. The matching degree of the cosine similarity and the expert's decision (and/or the player's questionnaire answers) will be measured before and after applying LSA, for detecting its impact.

VI. CONCLUSION

In this paper we have described a proposal for modeling the semantic space of a complex non-textual problem, i.e. an action game, using LSA. While the application of LSA to textual data is fairly straightforward, several research issues arise when the data involved are not textual, but represent players' actions and environmental (contextual) conditions. These research issues have been addressed and an experimental setup has been proposed for the novel use of the extracted model to player modeling.

REFERENCES

- [1] K. Anagnostou and M. Maragoudakis, "Data mining for player modeling in videogames", Proc. of the Panhellenic Conference on Informatics (PCI), Corfu, Greece, 2009.
- [2] R. Basili, R. Petitti, and D. Saracino, "LSA-based automatic acquisition of semantic image descriptions", Proc. of the Conference on Semantics and Digital Media Technologies, LNCS, vol. 4816, 2007, pp. 41-55.
- [3] B. Done, P. Khatri, A. Done, and S. Draghici, "Predicting novel human gene ontology annotations using semantic analysis", IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 7(1), January-March 2010, pp. 91-99.
- [4] Q. Dong, X. Wang, and L. Lin, "Application of latent semantic analysis to protein remote homology detection", Bioinformatics, vol. 22(3), Oxford University Press, 2006, pp. 285-290, doi: 10.1093/bioinformatics/bti801.
- [5] M. Ganapathiraju, N. Balakrishnan, R. Reddy, and J. Klein-Seetharaman, "Computational biology and language", Ambient Intelligence for Scientific Discovery, LNAI, vol. 3345, 2005, pp. 25-47.
- [6] B. Geisler, "An empirical study of machine learning algorithms applied to modeling player behavior in a first person shooter video game", MSc Thesis, 2002, University of Wisconsin-Madison.
- [7] A. C. Graesser, P. Penumatsa, M. Ventura, Z. Cai, and X. Hu, "Using LSA in AutoTutor: Learning through mixed-initiative dialogue in natural language", Handbook of Latent Semantic Analysis, T. Landauer, D. McNamara, S. Dennis, W. Kintsch Eds, 2007.
- [8] D. T. Haley, P. Thomas, A. de Roeck, and M. Petre, "A research taxonomy for latent semantic analysis-based educational applications", Proc. of the Conf. on Recent Advances in Natural Language Processing, Borovets, Bulgaria, 2005.
- [9] S. He, J. Du, H. Chen, J. Meng and Q. Zhu, "Strategy-based player modeling during interactive entertainment sessions by using Bayesian classification", Proc. of the 4th International Conference on Natural Computation (ICNC), November 2008, pp.255-261.
- [10] T. Hofmann, "Latent semantic models for collaborative filtering", ACM Trans. on Information Systems, vol. 22(1), January 2004, pp. 89-115.
- [11] T. Landauer, P. Foltz, and D. Laham, "An introduction to latent semantic analysis", Discourse Processes, vol. 25, 1998, pp. 259-284.
- [12] B. Lemaire, "Models of high-dimensional semantic spaces", Proc. 4th International Workshop on Multistrategy Learning (MSL), June 1998.
- [13] P. M. McCarthy, S. Briner, V. Rus, and D. McNamara, "Textual signatures: identifying text-types using latent semantic analysis to measure the cohesion of text structures", In A. Kao and S. Potet (Eds), Natural Language Processing and Text Mining, March 2007, pp. 107-122, doi: 10.1007/978-1-84628-754-1_7
- [14] P. Nakov, E. Valchanova, and G. Angelova, "Towards deeper understanding of the LSA performance", Proc. of the Conf. on Recent Advances in Natural Language Processing, 2003.
- [15] J. F. Quesada, W. Kintsch, and E. Gomez, "A computational theory of complex problem solving using the vector space model (part I): latent semantic analysis, through the path of thousands of ants", In J. J. Cañas (Ed.), Cognitive research with Microworlds, Granada, Spain, 2001, pp. 117-131.
- [16] D. Roberts, M. Riedl, and C. Isbell, "Opportunities for machine learning to impact interactive narrative", Workshop on Machine Learning and Games at NIPS, 2007.
- [17] G. Shahine and B. Banerjee, "Player modeling using knowledge transfer", Proc. of EUROIS GAMEON-NA Conference, Gainesville, Florida, 2007, pp. 82-89.
- [18] J. Steinberger and K. Jezek, "Using latent semantic analysis in text summarization and summary evaluation", Proc. of Conf. on Information Systems, Implementation and Modeling (ISIM), 2004, pp. 93-100.
- [19] R. Thawonmas and J. Ho, "Classification of online game players using action transition probability and Kullback Leibler entropy", Journal of Advanced Computational Intelligence and Intelligent Informatics, Special issue on Advances in Intelligent Data Processing, vol. 11(3), 2007, pp. 319-326.
- [20] D. Thue, V. Bulitko, M. Spetch, and E. Wasylishen, "Interactive storytelling: a player modelling approach", Proc. of the 3rd Conf. on Artificial Intelligence and Interactive Digital Entertainment (AIIDE), Stanford, California, USA, June 2007, pp. 43-48.
- [21] H. van Oostendorp and I. Juvina, "Using a cognitive model to generate web navigation support", International Journal of Human-Computer Studies, Elsevier, vol. 65(10), October 2007, pp. 887-897, doi:10.1016/j.ijhcs.2007.06.004.
- [22] V. Zampa and B. Lemaire, "Latent semantic analysis for user modeling", Journal of Intelligent Information Systems, Kluwer Academic Publishers, vol. 18, 2002, pp. 15-30.