Natural Language Processing in Serious Games: A state of the art.

Davide Picca[1], Dominique Jaccard[2], Gérald Eberlé[3]
[1,3] *University of Lausanne, {davide.picca, gerald.eberle}@unil.ch*
[2] *University of Applied Sciences of Western Switzerland, Dominique.Jaccard@heig-vd.ch*

Abstract

In the last decades, Natural Language Processing (NLP) has obtained a high level of success. Interactions between NLP and Serious Games have started and some of them already include NLP techniques. The objectives of this paper are twofold: on the one hand, providing a simple framework to enable analysis of potential uses of NLP in Serious Games and, on the other hand, applying the NLP framework to existing Serious Games and giving an overview of the use of NLP in pedagogical Serious Games. In this paper we present 11 serious games exploiting NLP techniques. We present them systematically, according to the following structure: first, we highlight possible uses of NLP techniques in Serious Games, second, we describe the type of NLP implemented in the each specific Serious Game and, third, we provide a link to possible purposes of use for the different actors interacting in the Serious Game.

Keywords: *E-learning, Natural Language Processing, Serious Games, Computer Assisted Learning;*

1. Introduction

In the last decades, Natural Language Processing (NLP) has obtained a high level of success and reached a popularity peak with IBM Watson [21] during the Jeopardy Challenge TV show organized by IBM. Over the same time frame, Serious Games have been further developed and applied to an increasing variety of fields including education, public policy, management, and health care. In this paper, we focus on serious games developed for educational purposes.

Interactions between NLP and Serious Games are already in place, considering that several Serious Games include NLP techniques. NLP is potentially a very interesting tool for Serious Games given the large percentage of Serious Games including communication aspects and provided that the majority of our communicative capabilities passes through linguistic information in the form of either spoken language (audio or video content) or written language (e.g., email or chat exchanges).

The objectives of this paper are twofold: on the one hand, providing a simple framework to enable analysis of potential uses of NLP in Serious Games and, on the other hand, applying this framework to existing Serious Games and providing an overview of the use of NLP in pedagogical Serious Games.

The proposed framework provides a systematic and structured overview of NLP techniques used in Serious Games and links them with their possible purposes of use:

- Why is NLP used in Serious Games (i.e., what are the objectives of using NLP in Serious Games)?
- What kind of NLP is implemented in Serious Games?
- Which actors may use results of NLP techniques in Serious Games?

In order to provide an overview of the possible uses of NLP, we selected 11 Serious Games leveraging on NLP methods. The selection was based on the availability of the full description of the games in articles dating back at least 5 years. As English has been the most studied language from the NLP point of view, the selection was restricted to English articles. Still, we mention other Serious Games using NLP in languages like French, Portuguese or Dutch.

The paper is organized as follows. In section 2, we present the overall framework of possible uses of NLP in Serious Games. In section 3, we present a synthesis of current NLP employed in the serious games that we have considered in this paper. In section 4, we present and analyze 11 Serious Games using NLP and in section 5 we conclude with some reflections on future works and explorations.

2. NLP in serious games: A simple framework

Although the application of NLP techniques to Serious Games has started some years ago, no paper has yet specifically analyzed how NLP methods were integrated into such games. This lack of academic investigation on the applications of NLP is motivated by the fact that, in general, NLP elements are employed as means to achieve other pedagogical or scientific objectives, but they never constitute objectives in themselves. Thus, in order to give an overall view of the application of NLP to Serious Games, it is essential to understand *what are the objectives* of using NLP, *what kind of NLP* and *which actors* take advantage of the NLP techniques implemented in Serious Games. Since most of the articles analyzed in this paper do not focus on NLP techniques explicitly, but rather on their applications, NLP modules in themselves are only briefly mentioned.

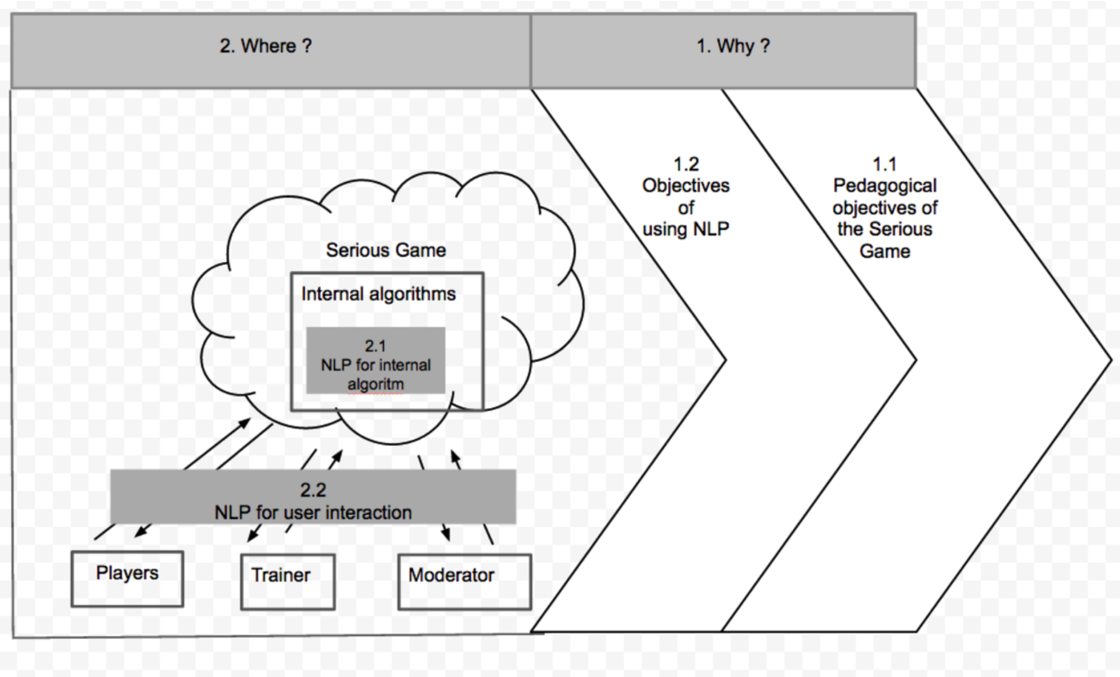

Figure 1 The pedagogical objectives with purposes of using NLP

As shown in Figure 1, pedagogical objectives (1.1) are the purpose of any pedagogical Serious Game. Hence, the game should be designed and developed focusing on achieving those pedagogical objectives.

The objectives of using NLP (1.2) should be coherent with pedagogical objectives and game concept. NLP may then be used in different parts of the Serious Game. For instance, NLP may be used both for or for improving games' internal algorithms (2.1) or for improving human interactions with the game (2.2).

For example, one use of NLP for improving the internal algorithms of the serious game (2.1) may be the use of a text written by the player to change the state of a finite state machine or to modify the value of variable used in one of the game's algorithms. We typically distinguish between "internal algorithm" and "user interaction." When using NLP for an internal algorithm, the user may not be aware that the Serious Game was altered as a result of what he has written.

Differently, when NLP is used for user interaction, actors obtain a direct feedback from the system depending on the texts written by players and have the opportunity to interact with the system depending on the feedback received. In order for NLP to support human interaction with the games, three main actors have been identified.

- Players: In Serious Games used for education players are characterized as the learners, the students. They may play the game individually or in groups.
- Trainers: It may be the teacher or the animator of the game. He may interact with the game in order to adapt the game depending on players' actions or on pedagogical event appearing outside the game.

- Moderators: In games that would be played by a very large amount of players, like in MOOCs (Massive Online Open Courses), the moderator may manage the learning community. Differently from the teacher, the role of the moderator does not entail any teaching activities or purposes. The moderator act as a game manager, ensuring that players or teams respect the rules of the game and act appropriately. He may interact with the games and players based on general behaviors of players. The role of moderator may be played by the trainer or by another person.

Thus, this framework allows for three cases of interaction (Game-Administrator, Game-Trainer, Game-Players) and one use of NLP without human interaction. Serious games may also offer Players-Players interaction, but those interactions usually do not ask for NLP support, as all players understand natural language.

3. Main NLP techniques employed

In this paragraph we provide a general description of the main NLP techniques employed in Serious Games. The main goal of this section is to provide the reader with a general understanding of the main techniques without specifying the detailed technical parameters for each game.
Among the several techniques mentioned in the papers under review, we identified four main methods which we outline in the following sections. The majority of the methods is used for classification tasks, which is one of the dominant applications addressed by NLP methods.

3.1 Finite-state machine (FSM)

This algorithm is mainly used in I-FLEG and deLearyous games. It is an abstract machine (like the Turing Machine, whose states are determined in a particular finite number and which can assume only one state at the time. In each state a symbol is read and such reading triggers an event or condition that is called *transition.* A particular FSM is defined by a list of its states, and the triggering condition for each transition. In Serious Games scenarios, states are a combination of player-nodes and Non-Playing Character (NPC) nodes (as in the deLearyous game) or a learning path (as in I-FLEG).

3.2 Expectation–maximization (EM) and K-means algorithm

The **expectation–maximization** (**EM**) **algorithm** is a statistical method widely employed in classification tasks for finding Maximum Likelihood (ML) or Maximum A Posteriori (MAP) parameter estimates in a model depending on unobserved variables. This method is entirely based on an iteration process attempting to perform an expectation (E) step. This first step aims at formulating a function for the expectation of the log-likelihood evaluated using the current parameter estimates. Successively, log-likelihood found on the *E* step is maximized throughout the maximization (M) step. This algorithm is used in Eveil-3D, AutoMentor and Land Science for language model in classification task as the speech acts in AutoMentor.

3.3 Latent semantic analysis (LSA)

LSA is successful technique widely employed in NLP tasks. The main assumption of this technique is that similar words occur in similar contexts. Thus, LSA is a vectorial semantics in which a matrix containing word counts per document is constructed from a large corpus. In order to reduce matrix cardinality, a mathematical technique called Singular Value Decomposition (SVD) is used. Once the cardinality reduced, words are then compared by taking a similarity measure (often the cosine) between the two vectors. iSTART, AutoMentor and ARIES exploit LSA in order to catch semantic similarities in learners' corpus.

3.4 Support vector machines

Support vector machines are supervised learning methods generally used for classification tasks. SVM works by projecting a set of training examples into a multidimensional space. Successively, the SVM model attempts to find a hyper-plane classification in such a multidimensional space maximizing the distance (i.e., the margin) from the closest training examples. SVM are used in DeLearyous and Mission Rehearsal for classification tasks.

4. Overview of the current use of NLP in pedagogical serious games

The proposed framework is used to analyze *why*, *what kind* and *how* NLP is used in the 11 selected Serious Games. All the Serious Games under review are used for pedagogical or training purposes. We classify games in different categories depending on the kind of competencies they aim to develop: reading and speaking competencies, soft skills and behavioral competencies, hard skills and scientific competencies,

4.1 Games using NLP for the development of reading and speaking competencies

4.1.1 Istart & istart-ME

iSTART is an ITS (Intelligent Tutoring System) designed to improve students' reading comprehension (by using reading strategies). This program consists of three modules:
1. An introduction module presenting the reading strategies,
2. A demonstration module showing the strategies in action, and
3. A practice module.
In iSTART-ME, students are presented with scientific texts and asked to type their own self-explanation. It is an extension of the previous iSTART [42] serious game, but improved by adding three generative programs: Coached Practice, Showdown and Map Conquest.
Coached Practice is an extension of the practice module developed for ISTART. An NPC called Merlin guides the learner and encourages him/her to give feedbacks or explanations. The student's reward are represented by points based on the quality of the self-explanation provided. In Showdown the learner's self-explanation is pitted against the self-explanation provided by an opponent NPC and the player with the highest score wins the round [33]. Map Conquest is also set up as a duel between the learner and two opponent NPCs. In addition to points, learners also win dices, in a number dependent on the quality of the self-explanation, which are used to conquer a territory on the map.

Game's pedagogical objectives
This game has been designed to improve the reading strategies of students. In particular it trains students to achieve a better and quicker understanding of the texts they read.

Objectives of using NLP
Analyzing the self-explanation provided by the student and giving him/her a feedback represents the critical part of this game.

NLP techniques used
As a first step, the program uses a metacognitive filter to eliminate sentences or part of sentences with no explanatory content. The second step consists in applying a combination of word-based and LSA-based algorithms to categorize students' answers according to four categories:
• 0 (inadequate, i.e., vague or irrelevant),
• 1 (barely adequate, i.e., sentence-focused by restatement or paraphrase of the sentence),
• 2 (good, i.e., local-focused by including concepts from the previous sentences), and
• 3 (very good, i.e., global-focused by using prior knowledge).
The word-based algorithm uses a list of content words (e.g., nouns, verbs, adjectives, adverbs) which are extracted from answers thanks to the CohMetrix toolkit (for further information about this toolkit, please refer to http://cohmetrix.com/, as of 2014-11-25). In addition, the word-based algorithm takes into account the length of the self-explanation by comparing the number of words and content words in the self-explanation text. On the other hand, the LSA-based algorithm is used sentence by sentence in the original scientific text. For further details about the LSA, refer to Section 3.3. The vector space for the LSA is built by taking three key benchmarks, i.e., the words in the title of the passage, the words in the text sentence and the words in the two prior sentences. The LSA rating is finally computed as a weighted sum of the cosines between the self-explanation sentence and the three benchmark vectors. The overall rating of the explanation, according to the four categories above, results from the weighted sum of the ratings relative to the word-based and the LSA-based algorithms [42,43].

4.1.2 Eveil-3D (Environnement Virtuel pour l'Enseignement Immersif des Langues)

This Serious Game is a computer-assisted, foreign language learning system developed by different institutions, under the leadership of PH-Karlsruhe (Pädagogische Hochschule Karlsruhe). The targeted students are adolescent low-proficiency non-native speakers, with a CEFR grade of at least A2. It is specifically dedicated to French and German students learning respectively German and French. The virtual scenario comprises a model of the Strasbourg Cathedral in which students are fully immersed. Students move in the cathedral interacting with an NPC using the targeted language [7]. Moreover, they wear 3D glasses and headphones and use a microphone. Thereby, students can interact verbally and their gestures are tracked, through a smartphone as emitting device and an IR camera as receiving device.

Game's pedagogical objectives
This game allows a full immersion in another language environment, as if the students were living in another country.

Objectives of using NLP
In this game a verbal automatic speech recognizer is trained and used to accept input from the students.

NLP techniques used
The fact that the students are adolescents complicates the process as they exhibit more variability in several domains as for example the spectral domain. Therefore, the acoustic models were adapted by MAP adaptation. For further details about the MAP algorithm, refer to Section 3.2. Since only a small corpus of near-speech and text is available to learners, language models were augmented with similar data from out-of-domain sources, using an EM algorithm found in the SRILM Toolkit [59] for the similarity detection. The language models are 4-gram case-insensitive language models with either modified Kneser-Ney or Witten-Bell smoothing, both discounting algorithms found in the SRILM Toolkit. The final language model is a weighted mixture of language models trained on the reduced out-of-domain sources, the weights being computed to minimize the perplexity on a subset of the games dialogs [68].

4.1.3 I-fleg (interactive french language learning game)

The I-FLEG game is a game integrated in the Second Life world. It is played by connecting to the Allegro Island which is populated with colored houses. Each house has an identical content and each room in each one of the houses is dedicated to a particular lexical field. The learner uses a first person perspective avatar to enter into a house and begins to play by clicking on an object to trigger exercises or by entering text into a text field. This Serious Game is designed to learn French as a second language for an audience at A1-A2 level (beginners) [1]. It differs from previous computer-assisted language learning programs by relying extensively on artificial intelligence and NLP techniques [16]. This game is comparable to an adventure game aiming at teaching vocabulary (on specific topics) and grammar features. The system also includes an English grammar and lexicon [2].

Game's pedagogical objectives
This game has been designed to learn certain aspects (especially, the vocabulary and the grammar) of the French language, as a second language.

Objectives of using NLP
As it is integrated in Second Life, this game benefits from 3D graphics, virtual reality and NLP technologies provided by the environment. The test exercises are produced in a non-deterministic way and their content is dependent on the learner's profile and the goal to be reached.

NLP techniques used
As mentioned above, clicking on objects in a room triggers exercises. These exercises and their solutions are generated on the fly from a knowledge base, whereby the knowledge associated with each object is encoded in OWL (Web Ontology Language). The knowledge object consists of an axiomatic description taken from the Wordnet hypernym structure (T-Box) and from an instance of the object (A-Box) associated to a set of facts describing the object (color, form, material, etc.). As the generated exercises are dependent on the learner's language level, the teaching goal, the exercise type and the (sequential) generation of exercises is subject to syntactic constraints. To deal with

syntactic constraints, the ontology is extended to an upper model by incorporating a linguistic ontology: the T-Box axioms are associated with linguistic concepts as part-of-speech and predicate argument structures [6]. In order to form natural language sentences, the concepts in ontology must be associated to words in a lexicon. This is accomplished by including a Linguistic Information Repository (LIR) model [53] in the program. This model associates multilingual information with ontologies to provide syntactic and morphological lexicons [17].
Another characteristic of the I-FLEG game is that it permits two types of learning: free learning and situated, interactive learning. In the first type of learning, the learner is free to follow his own path through the learning space after he/she has selected a learning goal. The program simulates the behavior of a human tutor and, by using a Finite State Automaton (FSA), it determines the next question. For further details on the Finite State Automaton, refer to Section 3.1. The learner's answer is classified into a dialogue act using a trained Logistic-Regression classifier (taken from the Mallet toolkit). Then, the FSA generates the following question, which is extracted randomly from an FSA-filtered corpus. For the second type of learning, i.e., the interactive learning, the learning path is defined through the trained corpus. It is presented to the learner in the form of grammar exercises (a shuffle, a fill-in-the-blank or a transformation exercise). To generate the exercise, the program uses the linguistic information (syntactic tree, morpho-syntactic information, lemmas) to build generated sentences [25].

4.2 Games using NLP for the development of soft skills and behavioral competencies

4.2.1 BOSS (BOrder Security System)

BOSS is a risk management training system developed at Macquarie University, Australia. In this game, the student is exposed to dangerous and critical situations to learn how to react promptly. The learner follows a scenario whose logic is contained in a knowledge base. He interacts within the scenario as an avatar immersed in a 3D virtual environment. The conversation with other NPCs is conducted through verbal input/output. To achieve this goal, the game includes a speech recognition module, a speech synthesis module and an emotion recognition module.

Game's pedagogical objectives
This game is designed for students who need to develop their risk management skills and who need to learn how to react appropriately in critical circumstances.

Objectives of using NLP
The three above-mentioned modules are used to manage the conversation between the NPCs and the player.

NLP techniques used
The game itself was developed with Vizard, a virtual reality software toolkit that offers the construction of an immersive 3D virtual environment and the creation of avatars. As some functionalities were not included in Vizard, some new modules (e.g., a scenario controller written in C++) were developed. This controller allows the creation of scenarios using the *lua* scripting language ("lua" means "moon" in Portuguese). A further module is the knowledge database, which is filled with a technique called Multiple Classification Ripple Down Rules, where a human expert creates production rules incrementally during the system training [55]. No further detail on the implementation could be found in the published literature about the speech recognition module, the speech synthesis module or the emotion recognition module [54].
BOSS was also used as a prototype system for generating and detecting deceptive language for a virtual agent [20]. In this context, two sets of scenarios were designed: the first one with neutral words and sentences and the second one with deceptive words and sentences. A set of 31 people was asked to participate to the game, and then to answer a few questions. These questions were designed so as to determine if participants detected which scenarios were written with deceptive wording. To set up the dialogues in the scenarios, the authors made an extensive use of the General Inquirer database. This database is a word list with around 200 tags attached to each word. The tagging concerns all kinds of attributes from positive or negative outlook to cognitive orientation classes. For an extensive list of the tags see http://www.wjh.harvard.edu/~inquirer/homecat.htm.

4.2.2 DeLearyous

DeLearyous is a Serious Game which aims at improving students' communication skills. Learners interact with an NPC in Dutch written natural language free text format [3]. The virtual character dialogues with the learner, taking into account his emotional status, which is classified in four emotion classes following the Leary's Rose or the Interpersonal Circumplex [63].
As a first step, a series of dialogues with a professional communication coach were transcribed to form the base of arguments for a given scenario. Then, the sentences formed by the players and the coach were annotated by several human annotators to position every sentence on the Interpersonal Circumplex (AKA Leary's Rose). It has been observed that the agreement between annotators was only fair, meaning that placing the sentences in the Circumplex quadrants is a difficult task even for humans.

Game's pedagogical objectives
This game has been designed to improve students' communication skills.

Objectives of using NLP
NLP is used in order to allow a natural language conversation between the learner and an NPC. To classify the player's position in Leary's Rose, the program must determine two axis of behavior: the dominant-submissive position and the opponent-cooperative position.

NLP techniques used
The deLearyous game is constructed as a pipeline of several modules, each module output being the input of the next module. The first module is a text input module, an editor that accepts free text from the player. Hereby, sentences are annotated with linguistic information using Frog, a Dutch parser. Features vectors are then created using an n-gram approach, which are compared to the training set of features vector using a Support Vector Machine classifier. The second module is an NLP module, whose objectives consist in predicting the position of the player's sentences in the Circumplex quadrants and determining which topic was intended by the player. In order to understand what the player intended to express, the words contained in his sentences are first lemmatized to standardize their form. Then, thanks to an extended keyword matching technique [64], the overlapping between the player's words and a benchmark vocabulary is maximized. To maximize the keyword-benchmark coverage, authors expanded the manually-set list of keywords with a set of keywords from a wordnet-based database named Cornetto, a lexical semantic database for Dutch, including synonyms, antonyms and other related words.
The third module in the pipeline is the scenario engine. The core of this engine is an FSM (for further details about the Finite State Automaton, refer to Section 3.1.) whose nodes are a combination of player nodes and NPC nodes; the relationships between nodes are manually created links between player's nodes and NPC nodes. The NPC's answer to the input text of the player is determined by the prediction of the emotional class and the intended topic found by the NLP module through the FSM relationships. As a result of the free text input, it is possible that the answer could not be determined. In that case, the scenario engine asks the player for more information. The fourth module in the pipeline is an audio module containing a database of pre-recorded answers, taking into account the emotional status shown by the NPC, a task too difficult to be obtained with a text-to-speech program. The last module consists in a 3D rendering module displaying the animation of NPC [64].

4.2.3 Façade

Façade is a one-act interactive drama. The learner is invited to a dinner during which a marital conflict takes place. The student's objective is to reconcile the couple. The game is designed into blocks of narrative stories (called dramatic beats) and one of the challenges is to reconstruct a real-time dramatic performance adapted to the learner's interactions. These interactions are made of written utterances which are mapped into a pre-defined set of discourse acts. The NPCs (a couple named Grace and Trip) respond with a pre-defined context-appropriate response, depending on the utterance category of the student (praise, criticism, provocation, etc.) [41,18,15].

Game's pedagogical objectives
This game is designed to train the students to find arguments in a difficult situation.

Objectives of using NLP
NLP is used to improve dialogue efficiency between the player and the NPC. NLP enables pragmatic dialogues between the player and the NPCs, with no emphasis on the syntax or the semantics of the input sentences.

NLP techniques used
The game is self-adaptive in the fact that it uses the learners' discourse to modulate its own responses. The program extracts the speech act of the player, then, in a second phase, the effect of the discourse act determines the behavioral and/or verbal response given by the NPCs.
The input sentences are considered as surface texts. To convert surface texts into discourse acts, the program uses production rules written in a custom language. These rules are manually written and consist of a template pattern and a discourse act. The template pattern consists of a combination of regular and occurrence expressions. An occurrence expression, unlike a regular expression, is insensitive to the positions and order of words in the sentences. Not only words can be used in the template pattern comparison, but also positional facts, a sentence fragment with a begin and an end position. This is an important notion as it emphasizes the meaningful part of a sentence. The template pattern can also contain a retraction operator "-", which is placed in front of any term. If the rule matches, then the retracted terms are removed from the internal representation of the sentence. This mechanism avoids another rule to be triggered with another meaning of the retracted terms. Internally, the learner's input sentences are represented by word occurrence facts, one per word. The rules are applied on these facts giving intermediate phrase occurrence facts. These latter facts are tested against the authored rules to give the discourse act or the positional facts. This mechanism allows for better efficiency in testing the input sentences. This efficiency is further enhanced by using the Rete [23] matching algorithm from the Jess API (Application Programming Interface). A further improvement in the rules treatment is the use of salience. Rules with salience are weighted rules, allowing preferred rules to be applied first [40].
It would be possible to overcome the huge effort of pre-defining the NPC responses by using a deep QA (Question-Answer) capability, such as the one designed for the IBM Watson system, and a subset of context-appropriate corpus [30,69].

4.2.4 FearNot! (Fun with Empathic Agents Reaching Novel Outcomes in Teaching)

FearNot! is a story telling application, an interactive video game trying to teach children strategies to prevent bullying and social exclusion [13]. This game enables children to explore physical and relational bullying issues and to develop strategies through empathic interactions with NPCs [3]. Children's empathy is triggered by the different characteristics of NPCs such as their appearance, behavior and emotion [29,39]. The story's episodes are generated by the bullying behavior of one NPC, while another NPC plays the role of the victim. Between two episodes, the victim NPC asks the children for advice on how to cope with bullying and changes his behavior in the next episode. The chosen anti-bullying strategy is not guaranteed to work, so the children can explore the strategies space.

Game's pedagogical objectives
The game aims at developing children's ability to face day-to-day bullying situations and at develop their ability to explore different possibilities of reactions to such situations.

Objectives of using NLP
Classifying speech acts and extracting semantic information generally represents a difficult task. Since this game is played by children, who usually express themselves with simply sentences, the process is simplified Consequently, the input domain will be constrained to a more restricted number of words.

NLP techniques used
Children provide advice to the victimized NPC with written free text input, which is then elaborated by a language processor. The NPC's behavior is driven by affective agents, which are built on the FAtiMA (FearNot Affective Mind Architecture) sub-system that uses OCC-based (Ortony, Clore & Collins cognitive model) cognitive appraisal along with coping behavior at both reactive and predictive levels [5,29]. For a detailed description of the FAtiMA sub-system please refer to [19].
In this program, the dialogues are composed either by speech acts between independent NPCs during the episodes or between the child and the victim NPC during the advice periods. As the scenario is restricted to bullying, the number of speech acts can be summarized within three categories: *help*,

confrontation and *socializing*. The NPC-to-NPC dialogues are reasonably well defined, beginning with the selection of a speech act and ending with the selection of an utterance. The selection of the speech act is determined by the emotional status of the NPC, and the utterance is selected from a template database. The child-to-NPC dialogues are more difficult to deal with as they are based on free text input. The authors decided to use a technique already used for the ELIZA program [67]. In practice, they rely on keyword analysis to determine the intended user speech act. As the language domain is restricted, the conversation subject is centered on the last scene, where the NPC is leading the conversation. Considering that the number of possible answers from the child user is also limited, the likelihood of determining the true intended speech act is higher. To generate believable dialogues between the various NPCs, speech acts must be structured in sequences. For this purpose, the program defines FSMs, one for each category of speech acts. For further details about the Finite State Automaton, refer to Section 3.1. Each FSM includes the language actions belonging to the given category and pre-defined potential answer elements. As already stated, for the user-to-NPC dialogues a different strategy with keywords recognition has been implemented. This latter strategymust cope with several difficulties like misspelling, syntax errors, idiomatic utterances, and eventually must recognize the child's speech act.

The problem with speech acts is that they do not contain any semantic information, they are not unique and there is no proper mapping to the syntax. Therefore, classifying utterances into speech acts requires the intervention of a human expert. In a closed domain as provided here, the problem can be resolved through the use of micro-grammars [26]. Micro-grammars are structural features in a sentence that can correlate to speech acts. These features are classified in three types: words and collocations (combinations of words), prosody (the voice tone) and conversational structure (preceding and current context). As a matter of fact, there is no prosody in the case of textual dialogs. To compensate for the lack of semantics, some language action has been added to each speech act. Language actions have been separated in two categories: NPC language actions and utterances (e.g., ask for advice, express reproach to user, beg for help, etc.) and child user answers (e.g., give advice, no answer, justification, etc.). The NPC-to-NPC language actions have been implemented in XML as rules templates. The rules can have variables, which are set at run time and may be recursive. The user-to-NPC dialogs identify speech acts by applying pattern matching on extracted words and collocations as well as on the conversational structure. [4].

4.2.5 Mission Rehearsal Exercise

Initially, this project was conceived as a military virtual reality-based training system for the US Army and it is now used to teach leadership skills in the context of high stakes social situations. The learner is placed in a situation in which several urgent tasks arise contemporarily. He/she is confronted with several NPCs and engages in a multi-modal interaction. The NPCs involved in the various scenarios have been designed to behave in a way which closely resembles human beings. The created NPCs are autonomous and able to understand the natural language, interpret human gestures, show emotions on their faces, reason about tasks, and generate new sentences. Moreover, they have a complete perception of their environment, the objects in the virtual world and the other interfacing beings (i.e., humans and NPCs) [31,61,60].

Game's pedagogical objectives
This game presents scenarios of high emotional tension situations, where the learner must decide how to cope with these situations.

Objectives of using NLP
In this game the authors ambition to create virtual NPCs with a behavior closely resembling that of real humans. To achieve this goal they integrated a great number of AI and NLP technologies.

NLP techniques used
The program includes the following components. First, there is a speech recognizer, which is designed in an unsupported mode, with a limited vocabulary and a finite-state grammar. Collected utterances are fed into the Viterbi decoder using a Hidden Markov Model (HMM) with Gaussian Mixture states and a MAP model. For further details about the Finite State Automaton, refer to Section 3.1. The weights of the MAP model have been trained on several sets of utterances [65]. The speech recognizer output (perfect or imperfect due to out-of-vocabulary words) is the input to a hybrid semantic parser. The parser is responsible for extracting the meaning of the speech in the form of internal shallow semantic frames. A frame or meaning element is a slot-value pair, where the value can be a lexical item or another frame element and which may be grouped further into an

ontology of classes (Addressee, Name, Location and Squad). Several statistical parsing methods are used to produce a best-guess. The first method is a simple conditional probability model representing the probability of finding a given slot-value pair having seen a given word or n-gram in the input. The probability associated with each of three types of n-grams (unigram, bigram and trigram) is added to produce the weight of each meaning element in the set of candidates meaning elements. Finally, the set is filtered through heuristic-based filters (elements with the highest weight) to obtain a complete frame. This method has been called a voting method. A second parsing method involves the use classifiers. In this context, the meaning elements are considered as classes and the feature set consists of unigrams, bigrams and trigrams from the training set. The feature values are defined as the term frequency times the inverse document frequency (tf*idf) for each n-gram. Hence, the final frame is found by selecting the preferred top n classes of input sentence feature vector. The selection is performed using two methods: a Maximum Entropy classification and a Support Vector Machine classifier. The filtering of the candidates is the same as for the voting method. A third parsing method uses the Language Model method. In this model the words are considered as statistically distributed and if the distribution can be estimated, then the candidate meaning elements are the top most probable slot-values pairs.[8] The semantic parser's output is the input to a dialogue management system. This system is a layered model, with each layer containing an information state and a set of dialog acts describing the changes of the information state. The layers are: contact, attention, conversation (participants, turn, initiative, grounding, topic, rhetorical), social commitments (obligations, restrictions) and negotiation. A detailed description of these layers can be found in [62]. The dialogue management system also interacts with a task planner, an action selector and an emotion model to generate a content selection. If the NPC program decides to speak, the sequence of the language production includes sentence planning, realization and ranking. The sentence planner receives its input from the dialogue manager in an impoverished form. This module uses a set of production rules to enrich the input with details concerning the events and objects involved. The realization phase is where the lexical process occurs. Each information state and event has several valid realizations in the lexicon. The set of realizations gives rise to a forest of possible trees, where each node preserves a link to the frames. In the ranking phase, all the trees of the forest are examined and ranked based on the nodes information and the emotional status. To mimic human beings more closely, an emotion model is used to model the role that emotions play in influencing decision-making, behavior and dialogues. The tree with the highest rank is selected as the final candidate [22]. The output speech is augmented with communicative gestures and sent to a rendering and a synthesizer module.

4.2.6 PlayMancer

The PlayMancer project is composed of a series of serious mini-games used in cognitive-behavioral therapy sessions, for patients with behavioral and addictive disorders (e.g., pathological gambling, eating disorders, etc.). The virtual environment proposes several islands, which serve as scenarios. The mini-games aim at improving patients' relaxation techniques and planning skills, so they develop new confrontation strategies against their disorder. Besides the traditional joystick, keyboard or mouse input devices, the program manages other kinds of components and devices. These other components are speech, touch, biosensors and motion-tracking devices. The PlayMancer platform has been constructed in a modular way, integrating existing components or newly developed components around the RavenClaw/Olympus framework [14]. The speech interface counts as part of the existing components. This interface was completed with a particularly emphasized emotion-recognition component [36].

Game's pedagogical objectives
The mini-games have been designed to immerse the patient in various situations where he/she is confronted with his/her pathological disorder. The ultimate objective is to teach patients different ways to overcome their disease.

Objectives of using NLP
The overall objectives of the techniques used is to combine the speech, body posture and emotion state of the patient to define his/her behavioral status and adapt the game interaction accordingly.

NLP techniques used
The speech interface includes several components. The voice activity detection serves as input to a speech recognizer and this latter component transforms the incoming word flow into transcribed utterances that are mapped into high-level concepts by a speech-understanding component. The

dialogue flow manager processes the concepts, including those coming from various other modalities, to generate feedback concepts that are transformed by a natural language generator into correct sentences. Finally, these sentences generate the output of a text-to-speech synthesizer. All these components are connected to each other through the Olympus hub [14].
The speech recognition system is based on the Sphinx-III decoding engine developed at Carnegie Mellon University (CMU). This component uses an acoustic model based on a three-state HMM, each state being modeled by a mixture of 8 continuous Gaussian distributions. The language model is a 3-gram words model, which was constructed with the CMU Language Model Toolkit [36] and in which the probabilities were extracted from the mini-games scenarios. The language-understanding module is based on the CMU Phoenix system. This module uses a flexible frame-based parser, which employs a semantic grammar. Semantic entities are slots containing the concepts extracted from the input sentences. The system operates by taking sub-strings and patterns and matches these against the constructed grammar. The grammar is constructed by concatenating domain-independent concepts (like “yes”, “no”, “hello”, etc.) and domain-dependent concepts introduced by the game authors. Possible slots are registered in parallel and passed to a confidence annotation program. The Phoenix system also takes into account some ill-formed utterances like restarts, repeats, ellipsis, anaphora or indirect references [32].
The confidence annotation system is the CMU Helios module. This module analyzes all the slots provided by Phoenix and uses features from different sources to compute the probability of correctly understanding the user’s intention. For more details please see http://wiki.speech.cs.cmu.edu/olympus/index.php/Helios. The interpretation with the highest score is forwarded to a dialog manager.
The RavenClaw dialogue manager consists of two parts: a dialogue tasks specifier and a dialogue engine. The first component integrates all the domain-dependent dialogue logic in the form of a tree of dialogue agents. Each dialogue agent holds pre-conditions, triggers and completion criteria (success or failure). The dialogue agents also encapsulate the concepts behind the dialogue tasks. The dialogue engine is a domain-independent component, which takes the tree of dialogue agents as input source to construct a dialogue stack. This component also manages conversational issues like error handling, request for help, suspending and resuming dialogue, turn-taking and timing behaviors, etc. It is the core of the RavenClaw system [10]. A more complete description of this module can be found in [11].
The output of the dialogue manager is sent to the CMU Rosetta language generator. It uses the same grammar rules as the Phoenix component to express in spoken language the dialogue tasks determined by the dialogue manager. Concatenating domain-independent templates with domain-dependent templates given by the games, authors generate the sentences.
The last component in the chain of modules is the speech synthesizer. PlayMancer uses the FestVox speech synthesizer. For detailed information and documentation please see http://festvox.org/ [9].

4.3 Games using NLP for the development of hard skills and scientific competencies development

4.3.1 AutoMentor and Land Science

AutoMentor is designed to simulate and replace a human mentor to guide single users and groups of users through the Land Science game. The Land Science game is used by middle school students to simulate urban and regional planning, with the objective of combining ecological and economic viewpoints [56,57,48].

Game’s pedagogical objectives
The ultimate goal of this game is to help learners to think like STEM (science, technology, engineering, mathematics) professionals.

Objectives of using NLP
AutoMentor replaces a human mentor by using several NLP techniques trained by human experts.

NLP techniques used
As discussed in [66], AutoMentor uses several analysis modules to interact with students, more specifically:
• A speech act classifier to classify the input into several categories (e.g., question, request, feedback, etc.).

• A newness and relevance module to identify if a new topic or an off-topic occurred.
• An epistemic network analyzer to identify which skill and knowledge occurred.
• A state transition network to navigate in the storyboard.

Thanks to the speech act classifier, speech acts are automatically assigned to different categories (e.g., question, request, feedback, etc.). In practice, students' chat data are extracted, processed using a leading tokens model (i.e., a model taking into account lexical and positional information) and then categorized with the help of two clustering algorithms. Clustering groups data according to similarity based on Euclidian and Manhattan distance. The two clustering algorithms used in this specific work are the K-Means and EM algorithms, As a final step, the automatically determined categories are compared with the categories as defined by a human expert. This comparison reveals that automatic classification leads to a more restricted number of categories than human classification.

The other noteworthy analysis module presented in this paper is the epistemic network analyzer. It is based on the epistemic frame hypothesis, i.e., the hypothesis that the mentor's vision is represented by means of his way of talking, listening, writing, reading, acting, believing, valuing and feeling. Each of these behavioral manifestations can be allocated to one of the following five categories: Skills, Knowledge, Identity, Values and Epistemology (the SKIVE elements).

To generate the epistemic network analyzer, the author exploits NLP techniques levering on the fact that linguistic phenomena are representative of the human mentor's behavior. Using written comments produced by human experts, the author investigates two crucial aspects, and more specifically: 1) the categories arising from the mentor's evaluations and suggestions and 2) the alignment between these categories and the SKIVE elements of an epistemic frame. Human experts constructed the following two sets of corpora:

1. The journalism practicum corpus, which is composed by a collection of student journalists' comments on news stories; and
2. The game corpus, which includes copyedit comments from a game called science.net designed for middle school students playing the role of a scientific reporter.

These two corpora are further segmented in the two most important speech acts for a mentor, namely, evaluation (e.g., "The report is fine") and suggestion (e.g., "Improve the first sentence") acts. Subsequently, an NLP analysis is conducted on the journalism practicum corpus to extract the fundamental clusters of the comments written by the human mentor. An LSA (for further details about the LSA refer to Section 3.3) is performed to compute similarity matrices for both the evaluation and suggestion acts comments. Similarity is defined by the cosine of two vectors representing two expressions in the LSA space. The LSA space used for computing the similarity was the TASA (Touchstone Applied Sciences Associates) corpus of 37'651 documents and 92'409 terms (as of 2014-11-12). An SVD is conducted on the similarity matrices, leading to a size reduction of large, sparse matrices. To infer which comment category is relevant for a particular context, a principal components analysis is applied to the matrices, followed by a varimax rotation to remove dependencies across categories. Thanks to a binary logistic regression, it is predicted which epistemic frames should be used in the epistemic network analyzer module (see below). The analysis on the journalism practicum corpus shows that a modest number of principal components speech acts is transferred to the game corpus. These results have been used to implement the AutoMentor program [27].

4.3.2 Operation ARIES! (Acquiring Research, Investigation and Evaluative Skills) / ARA (Acquiring Research Acumen)

"The aliens have invaded the Earth! The Fuaths of the planet Thoth in the Aries constellation are among us and try to destabilize the humans to steal their values by publishing flawed articles. The student has been hired by the Federal Bureau of Science (FBS) to detect the Fuaths as physically they look like humans."

This is the scenario behind Operation ARIES!, a Serious Game about scientific reasoning. This game combines a fantasy storyline, multimedia presentations and conversational interactions [24, 38] and the player's objective is to detect and resolve inconsistent information about scientific methods. The game is divided in three modules: the Training module, the Case Studies module and the Interrogation module. The Training module consists in a conversation between the learner and two pedagogical NPCs, a teacher NPC and a student NPC, about articles found in an eBook entitled "The Big Book of Science". In the Case Studies module the student must evaluate small reports from the psychology, biology or chemistry domains. He or she is confronted with three NPCs: a teaching NPC, a FBS agent NPC and a defector alien NPC. In the Interrogation module the student is confronted with three NPCs: a teaching NPC, a FBS agent NPC and a suspect NPC. The student's goal is to question the suspect NPC to determine if he is an alien or a human. The questions are first directed to the FBS agent NPC, which filters the questions to avoid any misclassification of the sentence by the program [45]. Operation ARA is an extension of Operation ARIES! that adds new features and games, and both serious games rely on AutoTutor [48].

Game's pedagogical objectives
This game trains students to detect inconsistent and incoherent scientific reasoning and to argue to restore the truth.

Objectives of using NLP
The program has been designed to use natural language conversations between the learner and several NPCs, relying on a written form for the learner and on a written and verbal form, combined with emotional facial expression, for the NPCs.

NLP techniques used
The conversations are based on an implicit, given curriculum script that describes the questions on the relevant topics, the expected answers, hints, prompts, anticipated misconceptions, common errors and their corrections. The script's texts constitute the base for the analysis of the student's answer. This analysis is conducted by using a combination of LSA and regular expressions. The LSA is based on a corpus extracted from the ARIES eBook, where each word in the eBook is assigned a number based on the relative frequency of the word in the texts. The LSA calculates the similarity of the learner's answer with the correct answer, the misconceptions descriptions or the common error sentences. This is achieved by transforming the pairs of sentences (the student's answer and one of the curriculum entry for the topic) into two vectors representing both bags of words. The similarity is then valued by computing the vector cosine between the two vectors. A result of 0 means no similarity and a value of 1 means full similarity. When the similarity value exceeds a given threshold, the answer is considered to be valid against the bag of words. To mitigate the LSA's insensitivity to the order of words in the sentence (i.e., negative answer and out-of-domain sentences are not detected), the authors introduced in the program regular expressions containing semantic information like words, word stems, synonyms, and metacognitive expressions ("I don't understand!") [28,12].

5. Synthesis and Conclusions

The review of 11 Serious Games and their interaction with NLP techniques presented in this paper allowed us to tackle each one of the above questions. The main conclusions and open points for further investigation can be summarized as follows. In the introduction, we stated three main questions, which are:

- Why is NLP used in Serious Games (i.e., what are the objectives of using NLP in Serious Games)?
- What kind of NLP is implemented in Serious Games?
- Which actors may use results of NLP techniques in Serious Games?

The framework and the table 1 (at the end of the paper) based on the framework intend to present a synthetic answer to these questions with the main objective of clearly showing that all the empty cells in the table let suppose that there is still a big potential of new possible use of NLP in serious games. So it is important to remark the high number of empty cells, because they are a clear signal that further considerations should be done on the exploitation of NLP for the different roles, opening a significant research opportunity.
With respect to the first question, we have concluded that NLP is typically used for text classification or speech tasks. In light of the current advancements in NLP, it appears that NLP is still only employed for relatively simply and well-established tasks. The introduction of more advanced applications such as adaptive-learning or cognitive-based machine learning techniques has not yet taken place, mostly because the interaction between Serious Games and NLP techniques is still in its early stages.
As a result of our review, we have been able to assess the type of NLP used in the context of Serious Games (i.e., question 2). We observed that vast majority of papers combining NLP with Serious Games uses outdated NLP techniques developed 20-30 years ago instead of exploiting more recently developed methods. In fact, most of the statistical techniques explained in Section 3 date back at least to the 1980s. Since then, NLP modeling has progressed dramatically; in particular, recent research in cognitive computing has led to the development of new computation techniques, which

outperform the existing algorithms. Since cognitive computing aims at simulating human thought processes in a computerized model, this model can be extremely useful to create Serious Games that are capable of solving problems just requiring a partial human assistance. In the current age of Big Data, a lot of data could be exploited to improve the quality of learning along pedagogical aspects, content aspects or psychological aspects; data collected from learners, trainers and moderators could be combined so as to provide a system that is better aligned and adapted to the learner's necessities. We believe that this possibility for improvement should already be considered and partially incorporated during the games development phase. In future research, we will analyze this idea by including statistical aspects, NLP and AI techniques to improve the responses of a series of games developed by one of the author.

With respect to question 3, it has emerged from our analysis that NLP techniques are mostly used to support players-game interactions. The fact that NLP algorithms are generally used to support players' interactions with the game may be explained by various factors. In the first place, pedagogical Serious Games incorporating learning objectives have players (i.e., students) at the center of the learning process. Secondly, NLP is mostly used for players' interaction because pedagogical Serious Games still replicate the framework of traditional games, in which the trainer usually does not exist.

Nonetheless, the Serious Games in which the trainer represents the main part of the learning process and delivers the pedagogical concept (e.g., classroom orchestration), NLP is used to support the trainer in detecting particular or non-usual student behavior in the game. In this context, NLP provides the trainer with a tool to monitor students' behavior and production, allowing him to adapt the lesson.

The use of NLP to support the moderator's activities is extremely rare. Although, e-Learning and MOOCs are expanding rapidly, current e-Learning and MOOCs are mainly based on "old traditional pedagogical concepts." We can expect e-Learning and MOOCs to follow the same change in pedagogical concept as has been observed for classroom-based courses, with a general move toward active pedagogies. If active pedagogies, like Serious Games, are used in MOOCs, the presence of a moderator managing the game community and interacting with the game based on the particular behavior of teams or players may be useful. The activity performed by the moderator could be supported by NLP techniques, which, for example, provide an overview and a synthesis of students' production of free texts.

References

[1] Amoia M., Brétaudière T., Denis A., Gardent C., Perez-Beltrachini L., et al. A serious game for second language acquisition in a virtual environment. Journal on Systemics, Cybernetics and Informatics (JSCI), 10(1):24–34, 2012.

[2] Amoia M., Gardent C., Perez-Beltrachini L., et al. A serious game for second language acquisition. In CSEDU (1), pages 394–397, 2011.

[3] Arnab S, de Freitas S., Bellotti F., Lim T., Stanescu I., Brisson Moreno-Ger A., Pereira G., Romero M., Ninaus M., Earp J., et a. D2.2: Game and Learning alliance, The European Network of Excellence on Serious Games. Public Deliverable, GALA project, 2012.

[4] Aylett R, Louchart S., and Pickering J., A mechanism for acting and speaking for empathic agents. In Autonomous agents and multi-agent systems workshop, 2004.

[5] Aylett R., Vala M., Sequeira P., and Paiva A., Fearnot!–an emergent narrative approach to virtual dramas for anti-bullying education. In Virtual Storytelling. Using Virtual Reality Technologies for Storytelling, pages 202–205. Springer, 2007.

[6] Bateman J.A., Upper modeling: A general organization of knowledge for natural language processing. In Proceedings of the Workshop on Standards for Knowledge Representation Systems, 1990.

[7] Bernert-Rehaber S. and Schlemminger G., Immersive 3d-technologien optimieren das Fremdsprachenlernen: "eveil-3d - Lernen in virtuellen Welten". Babylonia, 3:44–49, 2013.

[8] Bhagat R., Leuski A., and Hovy E., Statistical shallow semantic parsing despite little training data. In Proceedings of the Ninth International Workshop on Parsing Technology, pages 186–187. Association for Computational Linguistics, 2005. http://dx.doi.org/10.3115/1654494.1654514

[9] Bohus D., Raux A., Harris, T.K., Eskenazi M., and Rudnicky, Alexandre I. Olympus: an open-source framework for conversational spoken language interface research. In Proceedings of the

workshop on bridging the gap: Academic and industrial research in dialog technologies, pages 32–39. Association for Computational Linguistics, 2007.

[10] Bohus D. and Rudnicky Alexander I., Ravenclaw: Dialog management using hierarchical task decomposition and an expectation agenda. 2003.

[11] Bohus D. and Rudnicky Alexander I., The Ravenclaw dialog management framework: Architecture and systems. Computer Speech & Language, 23(3):332–361, 2009. http://dx.doi.org/10.1016/j.csl.2008.10.001

[12] Cai Y., Forsyth C., Germany M.-L., Graesser A. and Millis K., Accuracy of tracking student's natural language in operation aries!, a serious game for scientific methods. In Intelligent Tutoring Systems, pages 626–627. Springer, 2012. http://dx.doi.org/10.1007/978-3-642-30950-2_93

[13] Catalano C.E., Luccini M. A., Mortara M., Guidelines for an effective design of serious games. International Journal of Serious Games, 1(1), 2014. http://dx.doi.org/10.17083/ijsg.v1i1.8

[14] Conconi A., Ganchev T., Kocsis O., Papadopoulos G., Fernández-Aranda F., and Jiménez-Murcia, S., Playmancer: A serious gaming 3d environment. In Automated solutions for Cross Media Content and Multi- channel Distribution, 2008. AXMEDIS'08. International Conference on, pages 111–117. IEEE, 2008.

[15] Cruz-Cunha, M. M., Handbook of Research on Serious Games as Educational, Business and Research Tools, volume 1. Information Science Reference, 2012.

[16] Cruz-Lara S., Denis A., and Bellalem N., Linguistic and multilingual issues in virtual worlds and serious games: a general review. Journal For Virtual Worlds Research, 7(1), 2014.

[17] De Beaugrande R.A. and Dressler, W.U.. Introduction to text linguistics. Longman London, New York, 1981.

[18] Denis A., Falk I., Gardent C., Perez-Beltrachini L., et al. Representation of linguistic and domain knowledge for second language learning in virtual worlds. In LREC-The eighth international conference on Language Resources and Evaluation-2012, pages 2631–2635.

[19] Dernoncourt F., Of the Use of Natural Dialogue to Hide MCQs in Serious Games. francky.me, (1), 2012.

[20] Dias J., Mascarenhas S. and Paiva, A., Fatima modular: Towards an agent architecture with a generic appraisal framework. In Proceedings of the International Workshop on Standards for Emotion Modeling, 2011.

[21] Dras M., Richards D., Taylor M., and Gardiner M., Generating and detecting deceptive language in virtual agents. In International Workshop on Interacting with ECAs as Virtual Characters, page 38, 2010.

[22] Fleischman M. and Hovy E., Emotional variation in speech-based natural language generation. In International Natural Language Generation Conference, Arden House, NY, volume 2, page 4, 2002.

[23] Forgy C.L., Rete: A fast algorithm for the many pattern/many object pattern match problem. Artificial intelligence, 19(1):17–37, 1982. http://dx.doi.org/10.1016/0004-3702(82)90020-0

[24] Forsyth C. M., Graesser A. C., Pavlik P. Jr, Cai Z., Butler H., Halpern D., and Millis K., Operation aries!: Methods, mystery, and mixed models: Discourse features predict affect in a serious game. JEDM- Journal of Educational Data Mining, 5(1):147–189, 2013.

[25] Gardent C., Lorenzo A., Perez-Beltrachini L., Rojas-Barahona L. M., et al. Weakly and strongly constrained dialogues for language learning. In Proceedings of the 14th annual SIGdial Meeting on Discourse and Dialogue SIGDIAL 2013, pages 357–359, 2013.

[26] Goodwin C., Transparent vision. Interaction and grammar, 13:370, 1996.

[27] Graesser A., Cai Z., Wood J., Hatfield D., Bagley E., Nash P., and Shaffer D., Comments of journalism mentors on news stories: Classification and epistemic status of mentor contributions. Intelligent Tutoring Technologies for Ill-Defined Problems and Ill-Defined Domains, page 21, 2010.

[28] Graesser A.C., Wiemer-Hastings P.,Wiemer-Hastings K., Harter D., Tutoring Research Group Tutoring Research Group, and Natalie Person. Using latent semantic analysis to evaluate the contributions of students in autotutor. Interactive Learning Environments, 8(2):129–147, 2000. http://dx.doi.org/10.1076/1049-4820(200008)8:2;1-B;FT129

[29] Hall L., Lutfi L. Binti S., Nazir A., Hodgson J., Hall M., et al. Games based learning for exploring cultural conflict. In: AISB 2011 Symposium: AI & Games, 6-7 Apr 2011, York, UK.

[30] Hennigan B., Making the case for nlp in dialogue systems for serious games. 8th International Conference on Natural Language Processing, 1st Workshop on Games and NLP. Kanazawa, Japan. 2012

[31] Hill R. W. Jr, Gratch J., Marsella S., Rickel J., Swartout W. R., and Traum D. R.. Virtual humans in the mission rehearsal exercise system. KI, 17(4):5, 2003.

[32] Issa S. and Ward W., Cmu's robust spoken language understanding system. In Proceedings of Eurospeech, volume 93, 1993.

[33] Jackson J. T., Boonthum C., and McNamara D. S., istart-me: Situating extended learning within a game-based environment. In Proceedings of the Workshop on Intelligent Educational Games at the 14th Annual Conference on Artificial Intelligence in Education, pages 59–68, 2009.

[34] Jackson J. T, Dempsey K. B., and McNamara D. S. Game-based practice in a reading strategy tutoring system: Showdown in istart-me. Computer games, pages 115–138, 2012.

[35] Johnson W.L., Serious use of a serious game for language learning. International Journal of Artificial Intelligence in Education, 20(2):175–195, 2010.

[36] Johnson W.L and Valente A., Tactical language and culture training systems: using ai to teach foreign languages and cultures. AI Magazine, 30(2):72, 2009.

[37] Kostoulas T., Mporas I., Kocsis O., Ganchev T., Katsaounos N., et al., Affective speech interface in serious games for supporting therapy of mental disorders. Expert Systems with Applications, 39(12):11072–11079, 2012. http://dx.doi.org/10.1016/j.eswa.2012.03.067

[38] Laamarti F., Eid M., Saddik A.E., An overview of serious games. International Journal of Computer Games Technology, Volume 2014 (2014), Article ID 358152, 15 pages. http://dx.doi.org/10.1155/2014/358152

[39] Lester J., Lobene E., Mott B., and Rowe J., Serious games with gift: Instructional strategies, game design, and natural language in the generalized intelligent framework for tutoring. Design Recommendations for Intelligent Tutoring Systems, 2:205–215, 2014.

[40] Lester J., Mott B., Rowe J., and Sabourin J., Learner modeling to predict real-time affect in serious games. Design Recommendations for Intelligent Tutoring Systems, pages 199–208, 2013.

[41] Mateas M. and Stern A., Natural language understanding in façade: Surface-text processing. In Technologies for Interactive Digital Storytelling and Entertainment, pages 3–13. Springer, 2004. http://dx.doi.org/10.1007/978-3-540-27797-2_2

[42] Mateas M. and Stern A., Structuring content in the façade interactive drama architecture. In AIIDE, pages 93–98, 2005.

[43] McNamara D. S., Boonthum C., Levinstein I.B., and Millis K., Evaluating self-explanations in istart: Comparing word-based and lsa algorithms. Handbook of latent semantic analysis, pages 227–241, 2007.

[44] McNamara D. S., Jackson J. T, Boonthum C, Deng Y., and He X., istart-me: A natural language game-enhanced comprehension strategy tutor. Journal of South China Normal University, 6, 2012.

[45] Meinedo H., Caseiro D., Neto J., and Trancoso I., Audimus a broadcast news speech recognition system for the european portuguese. In Nuno Mamede, Isabel Trancoso, Jorge Baptista, and M das Gracas Volpe Nunes, editors, Computational Processing of the Portuguese Language, volume 2721 of Lecture Notes in Computer Science, pages 196–203. Springer Berlin Heidelberg, 2003.

[46] Millis K., Forsyth C., Butler H., Wallace P., Graesser A., and Halpern D., Operation aries!: A serious game for teaching scientific inquiry. In Serious games and edutainment applications, pages 169–195. Springer, 2011. http://dx.doi.org/10.1007/978-1-4471-2161-9_10

[47] Mote N., Sethy A., Silva J., Narayanan S., and Johnson W.L., Detection and modeling of learner speech errors: The case of arabic tactical language training for american english speakers. In Proceedings of the InSTIL Symposium on NLP and Speech Technologies in Advanced Language Learning Systems, 2004.

[48] Neto J. P., Silva R., Madeiras Pereira J., and Fernandes J., Solis' curse-a cultural heritage game using voice interaction with a virtual agent. In Games and Virtual Worlds for Serious Applications (VS-GAMES), 2011 Third International Conference on, pages 164–167. IEEE, 2011.

[49] Nye B.D., Graesser A.C., and Hu X., Autotutor and family: A review of 17 years of natural language tutoring. International Journal of Artificial Intelligence in Education, 24(4):427–469, 2014. http://dx.doi.org/10.1007/s40593-014-0029-5

[50] Oliveira L. C., Paulo S., Figueira L., and Mendes C., The inesc-id blizzard entry: Unsupervised voice building and synthesis. In Blizzard Challenge Workshop, 2008. See: http://www.festvox.org/blizzard/index.html

[51] Paulo S. and Oliveira L., Multilevel annotation of speech signals using weighted finite state transducers. In Speech Synthesis, 2002. Proceedings of 2002 IEEE Workshop on, pages 111–114. IEEE, 2002. http://dx.doi.org/10.1109/wss.2002.1224384

[52] Paulo S., Oliveira L. C., Mendes C., Figueira L., et. al., Dixi–a generic text-to-speech system for european portuguese. In Computational Processing of the Portuguese Language, pages 91–100. Springer, 2008. http://dx.doi.org/10.1007/978-3-540-85980-2_10

[53] Pellegrini T., Ling W., Silva A., Correia R., Trancoso I., Baptista J., and Mamede N.J., Overview of computer-assisted language learning for european portuguese at l2f. In CSEDU (2), pages 538–543, 2012.

[54] Peters W., Espinoza M., Montiel-Ponsoda E., and Sini M., Multilingual and localization support for ontologies. Technical report, Technical report, 2009.

[55] Richards D. and Porte J., Developing an agent-based training simulation using game and virtual reality software: experience report. In Proceedings of the Sixth Australasian Conference on Interactive Entertainment, page 9. ACM, 2009. http://dx.doi.org/10.1145/1746050.1746059

[56] Richards D and Taylor M., Incremental human-based training of an agent-based training system. In Proc. Agents Learning Interactively from Human Teachers (ALIHT), Workshop at 9th International Conference on Autonomous Agents and Multiagent Systems, Toronto Canada, May, 2010.

[57] Rus V., Moldovan C., Niraula N., and Graesser A. J.. Automated discovery of speech act categories in educational games. International Educational Data Mining Society, 2012.

[58] Shaffer D. W. and Graesser A.. Using a quantitative model of participation in a community of practice to direct automated mentoring in an ill-defined domain. Intelligent Tutoring Technologies for Ill-Defined Problems and Ill-Defined Domains, pages 61–68, 2010.

[59] Silva A., Mamede N., Ferreira A., Baptista J., and Fernandes J., Towards a serious game for portuguese learning. In Serious Games Development and Applications, pages 83–94. Springer, 2011. http://dx.doi.org/10.1007/978-3-642-23834-5_8

[60] Stolcke A., et al. Srilm-an extensible language modeling toolkit. In INTERSPEECH, 2002.

[61] Swartout W.R., Lessons learned from virtual humans. AI Magazine, 31(1):9–20, 2010.

[62] Swartout W.R., , Gratch J., Hill R. W.Jr, Hovy E., Marsella S., Rickel J., Traum D., et al. Toward virtual humans. AI Magazine, 27(2):96, 2006.

[63] Traum D and Rickel J., Embodied agents for multi-party dialogue in immersive virtual worlds. In Proceedings of the first international joint conference on Autonomous agents and multiagent systems: part 2, pages 766–773. ACM, 2002. http://dx.doi.org/10.1145/544862.544922

[64] Vaassen F. and Daelemans W., Emotion Classification in a Serious Game for Training Communication Skills. In Proceedings of Computational Linguistics in the Netherlands 2010.

[65] Vaassen F, Wauters J., Van Broeckhoven F., Van Overveldt M., Daelemans W., and Eneman K., Delearyous: Training interpersonal communication skills using unconstrained text input. Proc. of ECGBL, pages 505–513, 2012.

[66] Wang D. and Narayanan S.S., A confidence-score based unsupervised map adaptation for speech recognition. In Signals, Systems and Computers, 2002. Conference Record of the Thirty-Sixth Asilomar Conference on, volume 1, pages 222–226. IEEE, 2002. http://dx.doi.org/10.1109/acssc.2002.1197181

[67] Wang J., Li H., Cai Z., Keshtkar F., Graesser A., and Shaffer D. W.. Automentor: artificial intelligent mentor in educational game. In Artificial Intelligence in Education, pages 940–941. Springer, 2013. http://dx.doi.org/10.1007/978-3-642-39112-5_154

[68] Weizenbaum J.. Eliza—a computer program for the study of natural language communication between man and machine. Communications of the ACM, 9(1):36–45, 1966. http://dx.doi.org/10.1145/365153.365168

[69] Winebarger J. T., Stüker S., and Waibel A., Adapting automatic speech recognition for foreign language learners in a serious game. In Proceedings of the 3rd Workshop on Games and NLP (GAMNLP-14), October 3rd, 2014, North Carolina State University, Raleigh, NC, USA

[70] Zagal J. P., Tomuro N., and Shepitsen A., Natural Language Processing in Game Studies Research: An Overview. Simulation & Gaming, 43(3):356–373, October 2011. http://dx.doi.org/10.1177/1046878111422560

Table 1 Synoptic table for Serious Games and NLP

	Games' pedagogical objectives	Objectives of using NLP	NLP Techniques Internal algorithm	Player input to game	Game output to player	Trainer input to game	Game output to trainer	Moderator input to game	Game output to moderator
AutoMentor and Land Science	Help learners to think like STEM professionals	Simulate a human mentor	Speech classifier Epistemic network analyzer Behavioral classification by clustering LSA	Written	Written	Written comments	N/A	Written	Written
BOSS	How to deal with suspect immigrants as a border officer	Use verbal conversation	Production rules Speech recognition Speech synthesis Emotion recognition	Verbal	Verbal	Written	N/A	N/A	N/A
deLearyous	Improve your interpersonal communication skills with the aid of a NPC coach	Classify the player's emotional state	Text input classifier Conversation options module Pre-recorded audio module	Written	Verbal Gesture	Written	N/A	N/A	N/A
Eveil-3D	Foreign language learning	Speech recognition	Speech recognition Audio module	Verbal Gesture	Verbal	N/A	N/A	N/A	N/A

Façade	Reconcile a NPC couple while you are invited for dinner	Extract speech act and determine NPC behavior	Surface text rules Discourse act classifier Production rules Regular expressions Occurrence expressions	Written	Written	Written	N/A	N/A	N/A
FearNot!	How to react as a victim of a bullying behavior	Classify speech acts and extract semantic information	Discourse act classifier Production rules	Written	Written Gesture	Written	N/A	N/A	N/A
I-FLEG	Learning French as a second language	Analyze text input and generate learning activities	LTAG grammar Morpho-semantic lexicon Surface realization	Written	Written	Written	N/A	N/A	N/A
iSTART / iSTART-ME	Improve your reading comprehension and strategies	Analyze self-explanation text and give feedback	Discourse classifier Assessment algorithm	Written	Written	N/A	N/A	N/A	N/A

Mission Rehearsal Exercise	Acquire basic language and cultural behavior in a foreign country	Analyze verbal and gesture input and produce verbal and gesture output	Speech recognition Semantic parser Dialog manager Sentence planner realization Ranking Text-to-speech synthesizer	Verbal Gesture	Verbal Gesture	N/A	N/A	N/A	N/A
Operation ARIES! / ARA	Learn scientific methodologies with NPC coaches	Use natural language conversation	Assessment algorithm Regular expressions Curriculum scripts	Written	Written Verbal	N/A	N/A	N/A	N/A
PlayMancer	Learn to confront and overcome compulsive disorders	Analyze verbal and gesture input	Speech recognition Speech understanding NL generator Text-to-speech synthesizer	Verbal Gesture	Verbal	N/A	Written	N/A	Written
REAP Pictórico	Learn Portuguese as a second language	Provide natural language explanations	Text-to-speech synthesizer	Written	Written Verbal	N/A	N/A	N/A	N/A

Solis Curse	Test your knowledge while visiting a museum	Use natural language conversation	Speech recognition Speech act classifier NL generator Text-to-speech synthesizer	Verbal	Verbal	N/A	N/A	N/A	N/A
The Tactical Language and Culture Training	Acquire basic language and cultural behavior before being immersed in a foreign country	Use natural language conversation and provide feedback	Speech recognition Speech act classifier Dialog manager Text-to-speech synthesizer	Verbal	Verbal	N/A	N/A	Verbal	Verbal