Designing an intelligent dialogue system for serious games

Franck Dernoncourt
April 2012

Laboratory LIP6, 4 place Jussieu, 75005 Paris, France
franck.dernoncourt@gmail.com

Abstract

The objective of our work is to design a conversational agent (chatterbot) capable of understanding natural language statements in a restricted semantic domain. This feature is intended to allow a natural dialogue with a learner, especially in the context of serious games. This conversational agent will be experimented in a serious game for training staff, by simulating a client. It does not address the natural language understanding in its generality since firstly the semantic domain of a game is generally well defined and, secondly, we will restrict the types of sentences found in the dialogue.

Introduction

Whatever the type of learning, one of the ways to maintain attention and motivation of the player in a serious game is to allow him to interact with a virtual agent. Currently, this dialogue, whether in serious games or narrative video games (storytelling) as well as in most environments for human learning is achieved through the trees based on multiple choice questions (Thue et al. 2007). The dialogue is very constrained, therefore reducing the learning of the player, who must content himself with clicking on one of the possibilities, which ultimately decreases the motivation and the reflection. We believe that intelligent dialogue systems (also called advanced dialogue systems) may constitute a relevant answer to this problem. For example, if the business that we are interested in is a pharmacy or hospital, the dialogue between the simulated patient and the player, in this case a medical staff member, will help to get used to have dialogue with the patients and test their knowledge to solve usual, real-life situations that may be urgent and critical, where a mistake can be extremely serious, even fatal.

A dialog is a verbal activity which involves at least two interlocutors and is used to accomplish a task in a given communication situation. It is a coordinated sequence of actions (linguistic and non-linguistic) leading to a goal (Vernant 1992).

The idea of human-computer interaction based on natural language is not new: it began to emerge 50 years ago with the Turing test. Nevertheless, this issue, at the conceptual and practical level, remains topical. There are, for example, annual competitions like the Loebner Prize (Loebner 2003) or the Chatterbox Challenge whose objective is to mimic the human verbal interaction. However no program has achieved the level of a human so far (Floridi et al. 2009).

The history of natural language processing, which directly influenced conversational agents, reveals several epochs whose evolution is interesting:

1. a statistical-morphological approach between the years 1945-1955, characterized by the use of statistical methods involving the morphology of entries; these methods are making a successful comeback for machine translation,
2. a syntactic approach, marked by the use of formal grammar with Chomsky's linguistic work in 1955,
3. a semantic approach in the 1960s, epitomized by the first implementation of the chatterbot Eliza (Weizenbaum 1966),
4. a cognitive (Grice 1979), pragmatic (Moeschler and Reboul 1994) approach from the 1970s, coinciding with the emergence of knowledge representation. It was an era influenced by cognitive psychology with a focusing on mental operations or on the processes underlying the production of speech (Searle 1972).

These four approaches are now seen as complementary to each other and we get inspiration from them throughout our work.

In order to define performance criteria for conversational agents, we will consider the following four criteria pre-conditioning the development of an intelligent dialogue system proposed by (Rastier, 2001):

- learning (at least temporary integration of information about the user),
- inquiry (request for clarifications from the system),
- correction (suggested corrections to the question, when necessary),
- explanation (explanation by the system of a reply given above).

In the first part, we present the state of the art by focusing on AIML language, then we analyze in a second time how our approach overcomes the limitations of this language.

State of the art

Figure 1 shows an example architecture of a conversational agent. The user types a sentence that conversational agent converts to an abstract language, here AIML (Wallace 2003): this translation is used to analyze the content of the sentence and make requests via a search engine in a knowledge base. The response

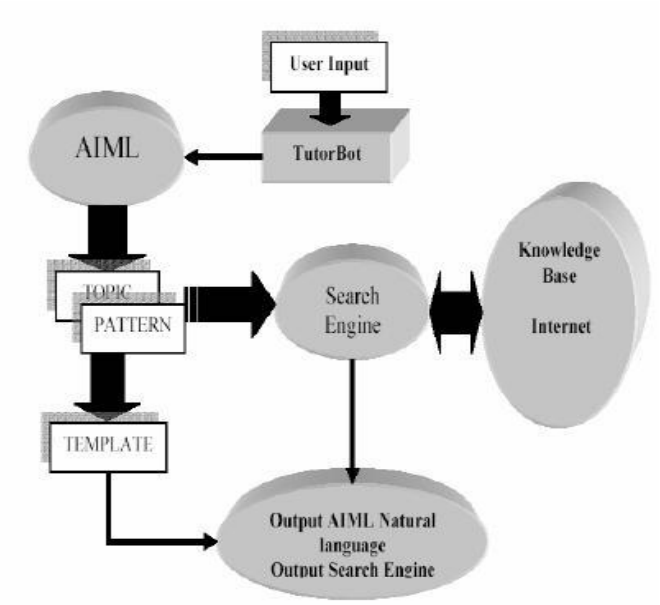

Figure 1 - Example of architecture of a conversational agent. Source: (De Pietro *et al.* 2005)

in natural language is generated through an abstract language, also AIML, and will be presented to the user.

However, this architecture is very rudimentary and rigid. For example, we must often update the knowledge base to include knowledge about the user, particularly in the context of a tutoring system that requires monitoring of the achievements of the user as well as his motivation.

Conversational agents fall into two main classes:

1. conversational agents for non-task-oriented conversation with the user on any topic with a friendly relationship such as ALICE (Wallace 2009),
2. task-oriented conversational agents, which have a goal assigned to them in their design.

The task-oriented conversational agents themselves are usually classified into two categories:

1. service-oriented conversational agents, e.g. guiding clients on an ecommerce site, such as the virtual assistant Sarah PayPal,
2. educational conversational agents, whose goal is to help the user learn.

Our work focuses on educational conversational agents (tutor bots). A number of educational conversational agents have been designed and implemented, such as (Zhang et al. 2009), (De Pietro et al. 2005), (Core et al. 2006), (Pilato et al. 2008) or (Cast et al. 2009).

Various architectures have been developed, here are the elements common to most of them:

• a knowledge base inherent in the field,
• a response manager,
• a storage structures of dialogues in the form of trees, especially in the case of educational conversational agents within a video game.

There is no standard powerful tool for understanding natural language which could manage a dialogue based on the context. This is one of the problems of the subject.

AIML is the language most often used by conversational agents, including educational conversational agents such TutorBot and TQ-Bot (Mikic et al. 2008). Although its simplicity and the relatively good performance of the conversational agents using it makes it attractive, AIML is however a very limited language that can be summarized in a simple pattern matching. Patterns of inputs (users' sentences) and outputs (responses of the conversational agent) is defined largely by expansion and a priori. For example, the well-known Alice bot, one of the best non-task-oriented conversational agents, winner in 2001, 2002 and 2004 of the Loebner contest and second in 2010, is in fact a simple list composed of tens of thousands of predefined questions/answers:

<category><pattern>Do I like cheese</pattern>
<template>How would I know if you like cheese or not?</template>
</category>

Approach

Our approach is modular in order to distinguish the different skills required to manage the dialogue and be able to develop independently the different modules, as shown in the flow chart shown in Figure 2.

Initially, the sentence entered by the user is orthographically and grammatically corrected if necessary. This step, although obvious, is neglected by many conversational agents like Cleverbot, yet one of the most successful conversational agents, as shown in these two examples from the official website http://www.cleverbot.com:

Example 1:
Human: Hi.
Cleverbot: How clever are you?
Human: Very clver.
Cleverbot: How clever?

Example 2:
Human: Hi.
Cleverbot: Are you a boy?
Human: Yes I am a boys.
Cleverbot: Are you a girl?

In Example 1, we see that the typo of the user who wrote "clver" instead of "clever" has clearly prevented the conversational agent Cleverbot from understanding the sentence and to propose a coherent dialogue. To resolve this problem, we use the free open-source spell checker Hunspell, used by major software like Google Chrome and LibreOffice. If we submit the reply "Very clver" to Hunspell, it shows us: "clver 7 0: clever, claver, clover, elver, caver, cover, cleaver." The first suggestion is the good one. Hunspell takes into account many parameters to order these suggestions, such as grammar, pronunciation, keyboard type used, etc. In

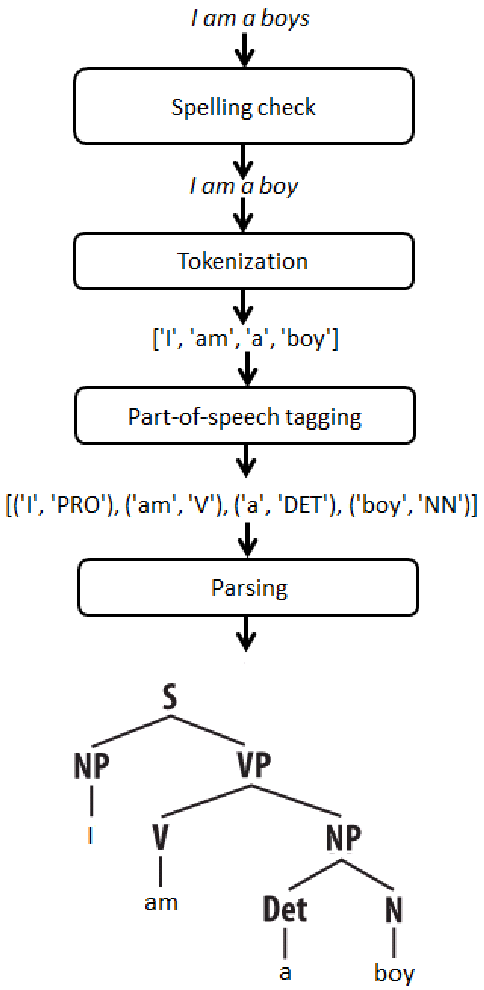

Figure 2 - Preprocessing flow

addition to Hunspell's order of suggestion, we can also make our own criteria, for example, give higher priority to the suggestions contained in the preceding words of dialogue. Example 2, which shows a grammatical error that misled the conversational agent, is also corrected by Hunspell. If we submit the phrase "Yes I am a boy" to Hunspell, it returns us: "* * * * + boy", the symbol * meaning that words do not need to be corrected.

In a second step, the user's sentence is analyzed lexically (tokenization): during this process the sentence is normalized by being broken into words. If simple heuristics based on regular expressions, i.e. on finite state automata, are sufficient to perform the lexical analysis of Western languages where words are usually separated by spaces, it does not suffice in some other languages such as Chinese. For example, the Chinese phrase (Mandarin) 看 上海 风景 can be segmented 看 / 上海 / 风景 (literally "look / Shanghai / landscape," ie, "look at the scenery of Shanghai") or 看上 / 海风 / 景 ("love / sea breeze / view", i.e. "love watching the sea breeze"). Seeing this sentence, a Sinophone would always segment the first way because the meaning from the second segmentation is less likely, as we see in the English translation. However, in other cases, segmentation can be ambiguous even for a Sinophone, like the phrase 学生会 组织 演出, which can be segmented in two different ways, namely 学生会 / 组织 / 演出 ("The student union/ organizes / a show ") or 学生 / 会 / 组织 / 演出, ("the student(s) / will (or can) / organize / a show"). In light of these two examples, we see that this kind of Chinese sentences makes it more difficult and underlines the need of using more complex heuristics, hence the existence of specific word breakers for the Chinese language such as the Stanford Chinese Word Segmentation. Thus, it is useful to distinguish this step of the analysis of the sentence as a step in itself, although in our case the treatment is simple because we apply it only on the English and French languages.

In a third step, we perform a grammatical labeling (part-of-speech tagging), whose objective is to classify words according to their grammatical function (nouns, pronouns, verbs, etc..). This classification is based on a dictionary and on the context in which the word appears. Grammatical taggers fall into two groups: stochastic and rule-based. An example of stochastic grammatical tagger is the unigram tagger, which classifies words only according to the probability of belonging to a class of words, calculated probability of a training corpus. For example, the Brown corpus (Francis and Kucera 1967), a grammatical unigram tagger correctly classified slightly over 80% of words (Bird et al. 2009), while the best taggers reach above 95%. This is a significant gain, but it shows that even a trivial stochastic tagger (unigram) presents correct results.Taggers are numerous grammatical English, but rare for the French. To our knowledge, there are only four directly in operational Cordial Analyzer, Tagg LIA, Stanford Tagger (available in French since January 6, 2012) and TreeTagger.

In a fourth step, we build the parse tree using a parser. This step allows us to detect among other things structural ambiguities, that is to say sentences with multiple parse trees. If the analysis of the context does not disambiguate, our conversational agent can ask a question via the user requesting it to clarify its sentence. The following excerpt from the film Animal Crackers (1930) shows a classic example of structural ambiguity:

> *Groucho Marx: While hunting in Africa, I shot an elephant in my pajamas. How an elephant got into my pajamas I'll never know.*

Figure 3 shows the two parse trees constructed from the segment underlined sentence, which means that this segment is structurally ambiguous. In the example of structural ambiguity given above, the second sentence, i.e. the context, removes the ambiguity by choosing the most unlikely meaning, hence the humorous nature of the transition. If the conversational agent fails to infer the meaning from context, it may ask who was wearing pajamas when firing.

These various steps of processing the sentences of the user are summarized in Figure 2, which shows the process applied to the user's sentence "I am a boy." Technically, these steps (except the first) are based on open-source library and free NLTK (Bird et al. 2009) which offers many features of language processing and has interfaces with databases such as WordNet (Fellbaum 2005) as well as with libraries and third party software such as grammatical tagger Stanford Tagger and automated prover prover9. Many corpora are also available via NLTK, which is very useful for implementing the training process and for testing, including performance tests.

Having established the initial steps in the processing of the sentence of the user, our work focuses on the reasoning engine, especially on the analysis of these intentions in the sentences of the user. These various data are then used to study the semantics of the sentence while calculating statistical data, especially via latent semantic analysis, which will be submitted to the reasoning engine. The latter launches queries to a database of domain-specific knowledge of the subject of dialogue such as the medical field if the dialogue simulates a conversation between a patient and a pharmacist.

In addition to the knowledge base, the reasoning engine must also take into account the educational goals of the game for this. We use educational data based on decision trees that are already used in computing environments for human learning. This allows us to reuse existing learning scenarios and to direct dialogue in order to complete the learning objectives. Moreover, these tree structures mitigate the problem of generation of the answer because the answers can be generated by the conversational agent based on predefined patterns and depending on the location of the dialogue in the tree of learning scenarios .

Thus we see that the critical point is the connection between the sentence of the user who in essence is expressed in an infinite space, the natural language, although semantically restricted by the context of the game, and the finite space corresponding to the tree of learning scenarios. To locate the user's sentence in this tree, the knowledge of the intentions is essential. Moreover, the recognition of intentions can increase the robustness of semantic analysis, as pointed out (Horvitz and Paek, 2001). The main objective of our work, now that the pre-treatment of sentences are implemented, is to design a system for recognizing intentions.

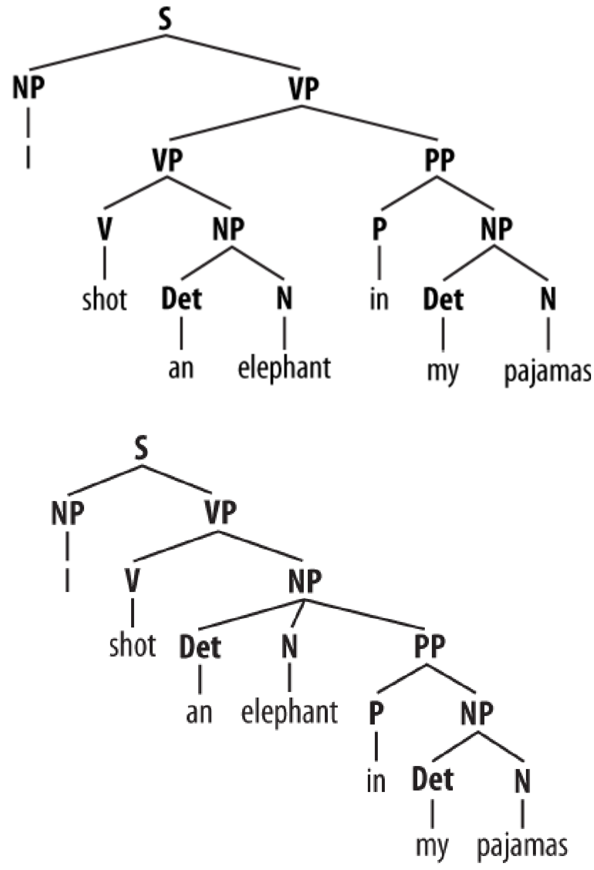

Figure 3 - Example of structural ambiguity. Source: (Bird *et al.* 2009)

The work on the recognition of intentions have begun about 30 years (Schmidt et al. 1978), (Wilensky 1983), and (Kautz and Allen 1986) are often considered to be the first papers in this field. Intention recognition systems are very similar to objective recognition systems, so much so that both types of systems are sometimes confused. The recognition of intentions leads to multiple applications ranging from natural language processing to computer intrusion detection and military strategy. Mechanisms of intention recognition have already been implemented as part of interactive stories, like LOGTELL (Karlsson et al. 2007).

As highlighted (Sabri, 2010), there are generally three major components in a system of intention recognition:

- a set of intentions among which the system chooses,
- knowledge about the relationship between actions and goals,
- an incoming stream of observed actions, which in our case corresponds to dialog acts.

Logic has often been used to implement systems for intentions recognition (Charniak and McDermott 1985), mainly based on the concepts of abduction and planning. The logical approach can be combined with statistical approaches (Pereira and Anh 2009) (Demolombe and Frenandez 2006). The sentence "Don't you think it's hot?" is an example highlighting the potential complexity of the analysis of intentions: the intention may be either a simple statement that indirect request to open a window or turn on air conditioning, or that the simple wish to continue the conversation. We see through this example that the research intentions can be very similar or identical to the research objectives.

Two major approaches have emerged to analyze the intentions (Raufaste et al. 2005): the classical approach, also known conventionalist approach, which seeks the intentions in the heart of the linguistic structure, and the intentionalist approach, who is based on the research on the context of intentions. These two approaches are complementary, as shown in Figure 4.

Finally, as shown (D'Mello et al., 2010), learning conversational agent can be enhanced when the modality is oral and not written. Therefore, we use Dragon NaturallySpeaking 11, which is the leader in speech recognition and edited by the firm Nuance, and the software AT & T Natural Voices ® Text-to-speech to transmit the responses of the conversational agent. Note that these software are not free.

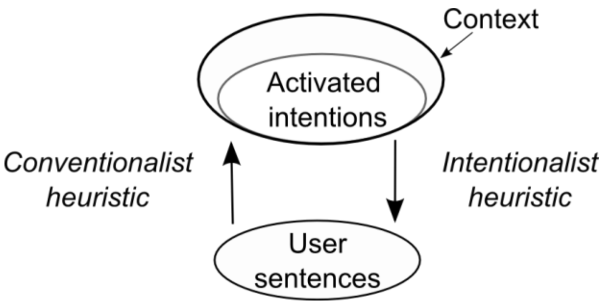

Figure 4 - Complementarity of the conventionalist and intentionalist approaches.
Source : (Raufaste *et al.* 2005)

Conclusions and perspectives

Our system will be implemented within the platform for serious games Learning Adventure (Carron et al., 2010), in which it will be evaluated through experiments with students.

The development of a dialogue between user and computer leads to potentially very many applications that are not limited to serious games. For example, oral or written interaction man-machine (Horvitz and Paek, 2001), designing bots, chat flooding, questions and answers systems, etc.

This theme is very timely as evidenced by IBM Watson (Ferrucci 2010), (Baker 2011), and the report by Gartner (Gartner 2011), whose following prediction shows the growing importance of this area:

By year-end 2013, at least 15 percent of Fortune 1000 companies Will use a virtual assistant to serve up Web self-service content to Enhance Their CRM offering and service delivery.

In addition, work on conversational agents have many common issues with document analysis, data mining, machine translation and the semantic web: all of these areas represent indirect applications of our work and interactions are considered with some of them.

Acknowledgments

I want to thank my supervisor Jean-Marc Labat for his invaluable advice and the DGA for its financial support.

References

Baker, S. eds. 2011. Final Jeopardy: Man vs. Machine and the Quest to Know Everything. Houghton Mifflin Harcourt

Bird, S., Klein, E., Loper, E. eds. 2009. *Natural Language Processing with Python*. O'Reilly Media.

Carron, T., Marty, J.C. and Talbot S. 2010. Interactive Widgets for Regulation in Learning Games. *The 10th IEEE Conference on Advanced Learning Technologies*, Sousse, Tunisia.

Core, M., Traum, D., Lane, H. C., Swartout, W., Gratch, J., Lent, M. V. and Marsella. S. 2006. Teaching negotiation skills through practice and reflection with virtual humans. *Simulation* 82(11):685–701.

D'Mello, S.K., Graesser, A. and King, B. 2010. Toward Spoken Human-Computer Tutorial Dialogues. *Human--Computer Interaction*. (4):289--323.

De Pietro, O., De Rose, M. and G. Frontera. 2005. Automatic Update of AIML Knowledge Base in E-Learning Environment. Proceedings of Computers and Advanced Technology in Education, Oranjestad, Aruba, August: 29–31.

Charniak, E., McDermott, D. eds. 1985. *Introduction to artificial intelligence*. Reading, Mass.: Addison-Wesley.

Fellbaum, C. 2005. *WordNet and wordnets*. In: Brown, Keith et al. (eds.), Encyclopedia of Language and Linguistics, Second Edition, Oxford: Elsevier, 665-670

Ferrucci, D. 2010. Build Watson: an overview of DeepQA for the Jeopardy! challenge. In Proceedings of the 19th international conference on Parallel architectures and compilation techniques (PACT 2010). ACM.

Demolombe, R., Fernandez, A.M.O. 2006. Intention recognition in the situation calculus and probability theory frameworks. In Proceedings of Computational Logic in Multi-agent Systems (CLIMA).

Floridi, L., Taddeo, M. and Turilli, M. 2009. Turing's Imitation Game: Still an Impossible Challenge for All Machines and Some Judges—An Evaluation of the 2008 Loebner Contest. *Minds and Machines*. Springer.

Fonte, F. A., Carlos, J., and Rial, B. 2009. TQ-Bot: An AIML-based Tutor and Evaluator Bot. Computer. *Journal of Universal Computer Science* 15(7), 1486-1495

Francis, W.N. and Kucera, H. eds. 1967. *Computational Analysis of Present-Day American English*. Brown University Press.

Jacobs, J. 2011. Hartner. http://www.gartner.com/DisplayDocument?id=1522020 , Virtual Assistant Vendor Landscape.

Grice, H.P., Kant, E. and Foucault, M. 1979. Logique et conversation. *Communication*, 30, 57-72.

Horvitz, E. and Paek, T. 2001. Harnessing models of users' goals to mediate clarification dialog in spoken language systems. *User Modeling* 3–13. Springer.

Kautz, H. and Allen, J.F. 1986. Generalized plan recognition. In Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI), 32-38.

Loebner, Hugh. 2003. Home Page of the Loebner Prize, http://www.loebner.net/Prizef/loebner-prize.html The First Turing Test.

Meier, E., Mast, M. and Luperfoy, S. 1996. Dialogue Processing in Spoken Language Systems. In ECAI'96, Workshop, Budapest, Hungary. Springer.

Fernando, A., Burguillo, C., Rodríguez A., Rodríguez, E. and Llamas, M. 2008. T-Bot and Q-Bot: A couple of AIML-based bots for tutoring courses and evaluating students. Frontiers in Education Conference. FIE. 38th Annual.

Karlsson, B., Ciarlini, A.E.M., Feijo, B., Furtado A.L. 2006. Applying a plan-recognition/plan-generation paradigm to interactive storytelling, The LOGTELL Case Study. *Monografias em Ciência da Computação Series* (MCC 24/07),

Moeschler. eds. 1994, *Dictionnaire encyclopédique de pragmatique*. Paris: Éditions du Seuil.

Pereira, L.M. and Anh, H.T. 2009. Elder care via intention recognition and evolution prospection. Procs. 18th International Conference on Applications of Declarative Programming and Knowledge Management (INAP'09), Évora, Portugal.

Pilato, G., Roberto P., and Riccardo R. A kst-based system for student tutoring. 2008. *Applied Artificial Intelligence* 22, no. 4 (4, 2008): 283-308.

Rastier, F. eds. 2001. Sémantique et recherches cognitives, *PUF* (2e éd).

Raufaste, É., Longin, D. and Bonnefon, J.F. 2005. Utilitarisme pragmatique et reconnaissance d'intention dans les actes de langage indirects. *Psychologie de l'Interaction*. 21--22.

Sadri, F. 2010. Logic-based approaches to intention recognition. Handbook of Research on Ambient Intelligence: Trends and Perspectives.

Schmidt, C., Sridharan, N. and Goodson, J. 1978. The plan recognition problem: an intersection of psychology and artificial intelligence. *Artificial Intelligence*, Vol. 11, 1978, 45-83.

Searle, J.R. and Pauchard, H. eds. 1972. reed. 2009. *Les Actes de langage*. Hermann.

Thue, D., Bulitko, V., Spetch, M. and Wasylishen, E. 2007. Interactive storytelling: A player modelling approach. In The Third Conference on Artificial Intelligence and Interactive Digital Entertainment (AIIDE), Stanford, CA, pp. 43–48.

Vernant, D. 1992. Modèle projectif et structure actionnelle du dialogue informatif. In *Du dialogue, Recherches sur la philosophie du langage*, Vrin éd., Paris, n°14, p. 295-314.

Wallace, R. 2003. *The elements of AIML style*. Alice AI Foundation.

Wallace, R. 2009. *Parsing the Turing Test, The Anatomy of A.L.I.C.E*. Springer.

Weizenbaum, J. 1966. Eliza. *Communications of the Association for Computing Machinery 9*. p. 36–45.

Wilensky, R. 1983. Planning and understanding. Addison Wesley, Reading, MA, 1983.

Zhang, H. L., Z. Shen, X. Tao, C. Miao, B. Li, and others. 2009. Emotional agent in serious game (DINO). In Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 2, 1385–1386.