

Intelligent Model Conception Proposal for Adaptive Hypermedia Systems

Mehdi TMIMI

Laboratory of Computing and
Interdisciplinary Physics, ENS,
USMBA
Fez, Morocco

Mohamed BENSLIMANE

Transmission and Treatment of
Information Laboratory, EST,
USMBA
Fez, Morocco

Mohammed BERRADA, Kamar

OUAZZANI

Laboratory of Computing and
Interdisciplinary Physics, ENS,
USMBA
Fez, Morocco

Abstract—The context of this article is to study and propose solutions for the major problems of adaptive hypermedia systems. In fact, the works and models proposed for these systems are made according to the tradition of studying first theories and rules, then modeling and designing a system that implements them. As a result, adaptive hypermedia systems designed reflect and support only the elements and information that were studied during the design phase. Also, these systems require a huge amount of data to power their architecture in order to start operating. This famous problem is called “cold start” and until now represents a challenge. So, in this paper, we will propose an intelligent and flexible model inspired by human nature and that proposes a promising solution to these problems concerning hypermedia adaptive systems.

Keywords—Adaptive hypermedia; artificial intelligence; deep learning; learner model; domain model; adaptation model; brain; neuron

I. INTRODUCTION

Adaptive systems have been of great importance in today's world, and are constantly improving by using advances in Internet technologies, psychology, artificial intelligence and education [1].

These adaptive systems can be defined as a learning management system (LMS) or even an individual learning platform that adapts the teaching to individual learner differences, such as cognitive abilities, learning styles, emotional states, etc.

The adaptive hypermedia is one of the adaptive learning systems that has emerged in response to the earlier traditional learning systems based on the "one design for all" [2] and "just on the web" approach. Its main characteristic is to give learners an active role [3] in building their knowledge and skills by adapting all the structural, visual and contextual aspect of the learning process.

To achieve these ends, three interrelated models are required:

1) *Learner model*: describes learner characteristics such as: skills, knowledge, learning styles, etc. [4]

2) *Adaptation model*: describes the set of construction and

presentation rules that are responsible for constructing the content to be delivered to the learner.

3) *Domain Model*: describes how the information content of the application is structured [5].

These three models mentioned above represent the core, the resources and the fundamental elements for the functioning of the adaptive hypermedia, these elements are first collected from scientific and academic works then connected and structured in a logical conception that we call a model.

After studying and proposing new conceptions for each one of the three models, which we published in several papers [6] [7], we intended to start the development of our hypermedia adaptive. However, we first had to deal with two major problems, which are: conceptual flexibility and cold start.

As the days go by, scientific research brings us new elements and relationships that influence the learning process and that need to be added manually and then applied by our adaptive hypermedia system. Unfortunately, and as known, such frequent conceptual changes are often costly in terms of performance and time and risky in terms of data integrity.

Also, the works and references that form the basis of our models may not be fully applicable and depend on the context.

Concerning the "cold start" problem, the main challenge has always been how to collect data about learners without disturbing them with questionnaires, especially in the early interactions with the system when learners can lose motivation and get bored if we welcome them with pre-tests and quizzes.

To overcome these problems, we drew inspiration from human nature (the learner) itself. And we propose for adaptive hypermedia a scalable and flexible system that mimics human complexity with the ability to learn and make decisions automatically.

II. INSPIRATION AND PROJECTION

At the early stage of our reflection, we thought of a smart model that learns and automatically makes decisions. In fact, that kind of model looks similar to the work done on artificial intelligence and deep learning. But we did not take that path and we just observed ourselves and thought deeply about human nature, which is very complex in itself.

A. Inspiration

We; as humans; all our memories, experiences, abilities, and knowledge are managed mainly by our brain.

The human brain is very organized; it is composed of several parts that each has specific roles while being complementary to each other.

This complementarity is done by neurons. In fact the total number of neurons was estimated at 85 billion [8], but recent studies [9] estimated a total of 105 billion neurons in the human cerebellum alone [10].

In Fig. 1, we show the internal composition of a neuron

and how the communication between different neurons is done. In fact, each neuron has a cell body, dendrites and an axon. The communication between two neurons is done by chemical and electrical processes, the electrical nerve messages that arrive at the end of the nerves are found in a neuron, and trigger a secretion of chemical molecules that will deposit on the next neuron, where they will be translated again into an electrical message in the other neuron [11].

Finally, neurons have the ability to learn and depending on the type of learning, the connections between the neurons involved are either “significant or stronger” or “less and weaker” (as shown in Fig. 2).

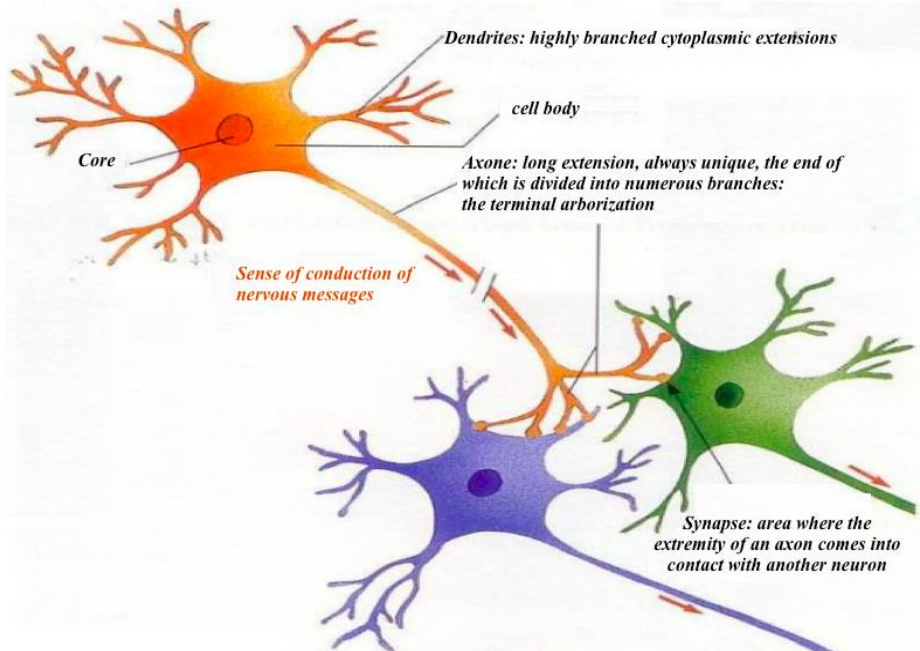


Fig. 1. The Different Elements of a Neuron and how the Communication is Done [11].

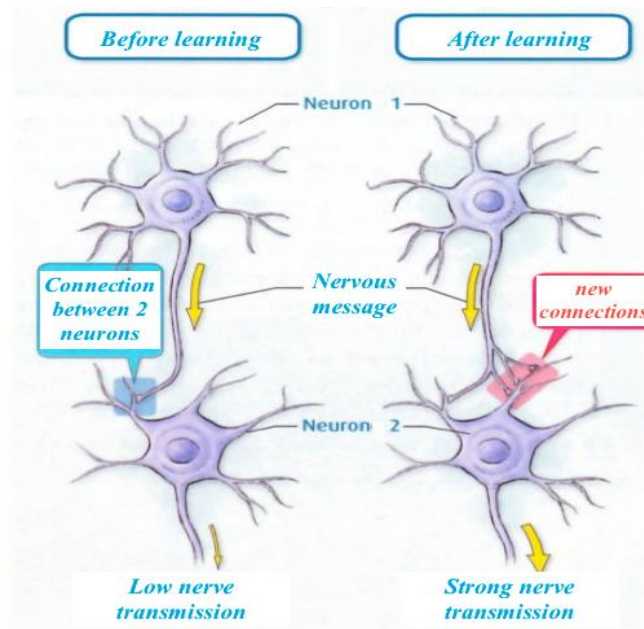


Fig. 2. Neurons before and after Learning [12].

B. Projection

The above was just some definitions and explanations of how the human brain works, let's now do a Reformulation of this complexity while making a projection on our field of research.

Since each neuron represents an information, we will follow the same logic considering that any element of our system will play the role of a neuron. This element refers to concrete and unique information within our system and will be defined as a couplet (element name, element value).

Concerning the communication between the neurons, we will implement it as logical and reflexive relationships between the elements, where they can play both the role of inputs (like neuron dendrites) or outputs (like neuron axons) for these relations.

Finally, and as we mentioned earlier concerning the learning ability of neurons, we thought of a coefficient mechanism that will indicate the strength of connections and relationships between elements.

III. PROPOSITION

In the following and as shown in Fig. 3, we will present our intelligent model based on the projection discussed above and designed with UML2 class diagram.

We will first present the different components and relationships of our model, then we will explain the constraints and rules necessary for its operation and finally we will detail

how to implement them while presenting the main use case and a future extension of our model once it is in action.

A. Components and Relations

Our model is composed of three classes, which are:

- 'Element': this class represents an information within our system and it is characterized by a unique identifier, name, value and a description. The identifier [id property] will uniquely reference this information. Also, each instance of the element class can simultaneously play many 'input' and 'output' roles in different relations.
- 'Relation': This class is characterized by a unique identifier, name, rule and a date. The rule property defines the actions and operations to be performed for all the results of this relation, the allowed values are: {Update, Delete and no Action}. Also, each instance of the Relation class can have one or more Result, and an instance of result class concerns one and only one Relation.
- 'Result': this class is characterized by a unique identifier, coefficient, state and a date. The coefficient property; which is an integer value; varies from 1 to ∞ depending on the number of times that the system has approved this result. Finally the state property; which is a string value; describes the state of this relationship. The allowed values are: {current, historic}.

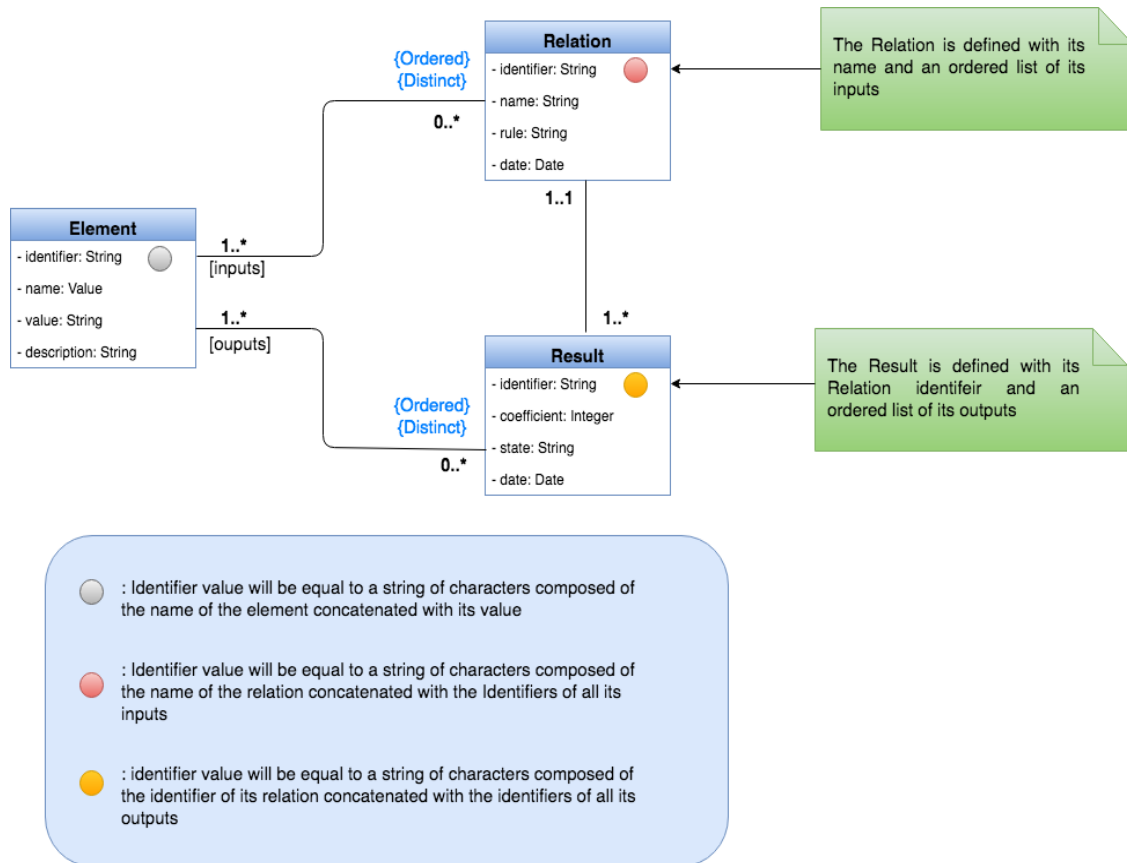


Fig. 3. Class Diagram of Our Proposed System.

B. Constraints and Rules

All the constraints and rules that we will explain are developed to ensure the uniqueness of some components of our model.

- The first rule concerns the uniqueness of the element, i.e. it is impossible to have two elements with the same identifier.
- The second rule concerns the uniqueness of a relation, which is defined by its name and an ordered and distinct list of its inputs. This rule will make it impossible and forbidden to have two relations that have the same names and the same Inputs. In Fig. 4, we present a concrete explanation of this rule where we have a 'Relation: {r1}', which has "define" as a name and two inputs elements e1 and e2. By involving the constraint cited above the 'Relation: {r4}' can not be created and the system will consider it as a 'Relation: {r1}'
- The third rule concerns the uniqueness of a Result, which is defined by its relation name and an ordered and distinct list of its outputs. This rule will make it impossible and forbidden to have two Results that have the same Relation and the same Inputs. In Fig. 5, we present a concrete explanation of this rule; Whenever the system approves an existing Result, it will be translated by incrementing the coefficient of the 'Result: {res1}' instead of creating a new 'Result: {res3}'.

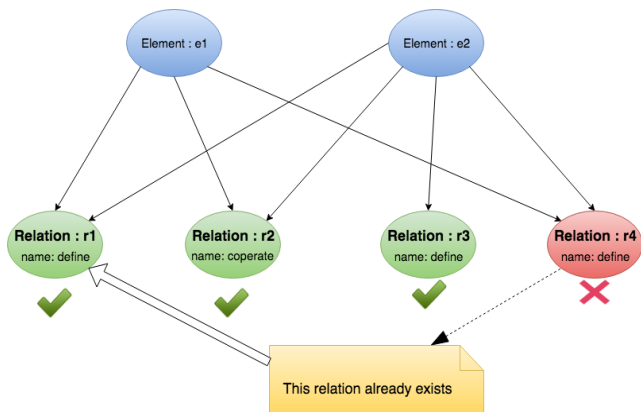


Fig. 4. Diagram Expressing the Constraint of the Relation Identifier.

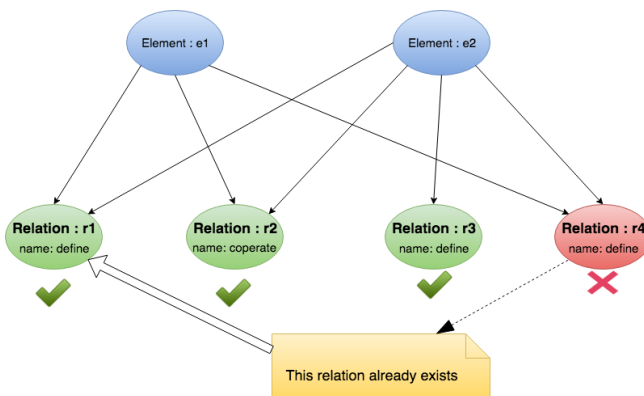


Fig. 5. Diagram Expressing the Constraint of the Result Identifier.

C. Implementation Rules

The major difficulty in expressing the rules explained above is summarized in how to ensure the uniqueness of an element, a relation and a result?

In fact, we used the relational data logic model (MLD-R) for this purpose and we worked on algorithms that will ensure the generation of the primary keys.

In the following, we will present for each rule the mechanism of generation of its identifier.

- Concerning the uniqueness of the element, the identifier will equal to a string of characters composed of the element name concatenated with its value. **Element Identifier** = Element name + Element value
- Regarding the uniqueness of the relation, the identifier will equal to a string of characters composed of the Relation name concatenated with a sorted and distinct sequence of the input element identifiers.

$$\text{Relation Identifier} = \text{Relation name} + \text{Element1 Identifier} + \text{Element2 Identifier} + \dots + \text{ElementN Identifier}$$

We added {ordered} and {distinct} constraints to the list of input element in the class diagram (Fig. 3) to ensure that we will not have in any case, two Relations that have the same name and the same list of input element and also to prevent the same element to play multiple input roles for the same relation.

Also, at the moment when the system approves or triggers a Relation, our system will first generate the identifier dedicated to this Relation. Then, from this generated identifier, the system will look for this relation, and following the result of this query our system will decide either to create a new relation or just use the existing one.

- Concerning the uniqueness of the Result, the identifier will equal to a string of characters composed of the Relation identifier concatenated with a sorted and distinct sequence of the outputs elements identifiers.

$$\text{Result Identifier} = \text{Relation Identifier} + \text{Element1 Identifier} + \text{Element2 Identifier} + \dots + \text{ElementN Identifier}$$

We added {ordered} and {distinct} constraints to the list of output element in the class diagram (Fig. 3) for the same purpose explained above.

Also, when the system approves or triggers a Result, our system will first generate the identifier dedicated to this Result. Then from this generated identifier, the system will check if we already have a Result with this identifier, and following this query our system will either decide to create a new Result and initialize its coefficient to 1 or just use the existing Result while incrementing its coefficient.

D. Main Use Case and Future Extension

Before discussing the main use case, we will first show the location of our intelligent model within the global architecture of adaptive hypermedia.

In order to do that, we show in Fig. 6 the global architecture proposed by the Munich reference model [13],

which consists mainly of three models (user / learner, domain and adaptation model).

Unlike this architecture proposed by Munich, we will introduce our intelligent model while keeping the same architecture composed of the three models.

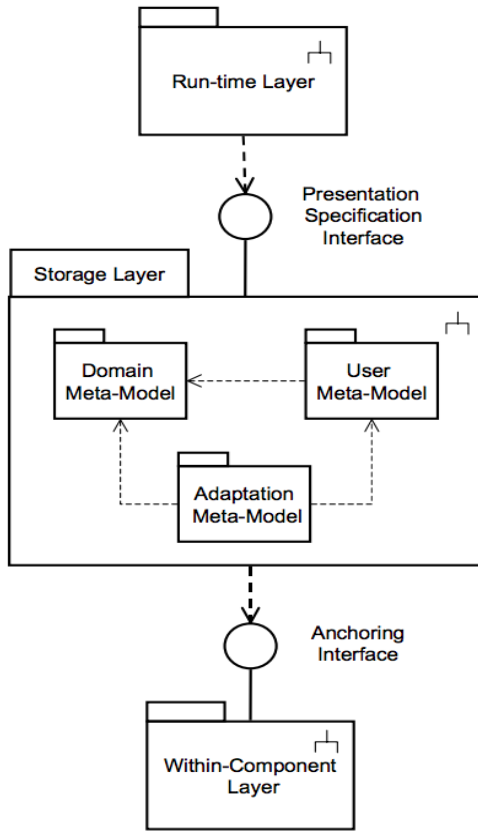


Fig. 6. Munich's Architecture of Adaptive Hypermedia Applications [13].

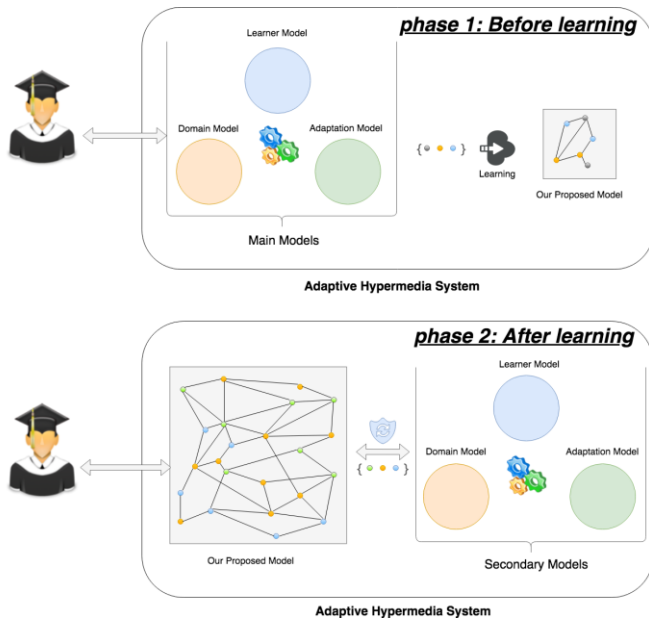


Fig. 7. Explanatory Diagram of the Two Phases of Our Intelligent Model within the Adaptive Hypermedia.

In fact, we have planned two phases for our system:

The first is the learning phase of our intelligent model, during this phase the adaptive hypermedia is based primarily on the three models while our intelligent model is simply observing and learning by defining and connecting the different elements of these three models (Fig. 7). In fact, we imagine our proposed model as a baby that needs to be educated first, and over time it will begin to apply what it has learned.

The second phase represents the final phase where our intelligent model takes a step forward and becomes the main model that drives the adaptive hypermedia while the three models become secondary (Fig. 7) and considered as its support.

Moreover, the mechanism of the coefficients that we have implemented in our system was made to imitate the consciousness and the subconsciousness of the human being.

So, whenever we force our system to trigger a result or supply it with data, the system is considered to be in a consciousness mode while the subconsciousness mode is reserved for autonomous decisions and behaviour made by our system.

Regarding the problem of cold start, we thought from the beginning to reorient the main challenge of "how to collect data on learners without disturbing them with questionnaires?" to "how to make decisions using the minimum of data?"

Indeed, from all the previous explanations, we can clearly see that our system forms a network of connected elements that continues to grow while having full flexibility in terms of structure and depth.

As a result, the bigger it grows, the less input data is required to make decisions.

Finally, we thought of a future extension of our model. This extension will be based on our current model to generate a network of abstract elements. By abstract element we refer to the definition or Meta data of the element.

This extension will present all our global rules and theories established which are based on the experiences learned by our system.

IV. CONCLUSION AND PRESPECTIVES

Our intelligent system will operate side-by-side with existing models within the hypermedia adaptive systems while enabling existing ones to become more dynamic and flexible by overcoming the two main conceptual flexibility and cold-start issues.

Also, the extension of the scope of our intelligent model is still at the theoretical phase, therefore we intend to work on its implementation in order to have a unique and independent model that can replace all other models on which adaptive hypermedia is based.

Finally, we plan in our next work to develop the global architecture of our adaptive hypermedia by linking all our proposed published models (learner model, adaptation model,

domain model and our proposed intelligent model) and then develop and test our system, which is the final goal of our research.

REFERENCES

- [1] Kara, N., & Sevim, N. (2013). Adaptive Learning Systems: Beyond Teaching Machines. *Contemporary Educational Technology*, 4(2), 108–120.
- [2] G. Kaya and A. Altun, "A Learner Model for Learning Object Based Personalized Learning Environments," *Commun. Comput. Inf. Sci.*, vol. 240, pp. 349–355, 2011.
- [3] A. Behaz, M. Djoudi, "Approche de Modélisation d'un Apprenant à base d'Ontologie pour un Hypermédia adaptatif Pédagogique", In: CIIA, 2009.
- [4] D. Samia, B. Taher, and L. Yacine, "Une nouvelle approche pour l'adaptation d'un hypermédia pédagogique au profil cognitif de l'apprenant en utilisant XML," in *2nd Conférence Internationale sur l'Informatique et ses Applications (CIIA'09)*, 2009.
- [5] P. M. E. Wu, H.; Houben, G.J.P.M.; De Bra, "Supporting user adaptation in adaptive hypermedia applications," in *P. Vet, van der, & P. M. E. De Bra (Eds.), Proceedings Conferentie Informatiewetenschap 2000*, 2000, vol. Vol. 00-20, pp. 88–98.
- [6] Tmimi, M., Benslimane, M., Berrada, M., & Ouazzani, K. (2017). A Proposed Conception of the Learner Model for Adaptive Hypermedia. *International Journal of Applied Engineering Research*, 12(24), 16008–16016.
- [7] Tmimi, M., Benslimane, M., Berrada, M., & Ouazzani, K. (2018). Implemented and Tested Conception Proposal of Adaptation Model for Adaptive Hypermedia. Accepted paper for publication in the *International Journal of Emerging Technologies in Learning (IJET)*
- [8] Williams RW, Herrup K (1988) *The control of neuron number*. *Annu Rev Neurosci* 11:423–453.
- [9] Andersen BB, Korbo L, Pakkenberg B (1992) A quantitative study of the human cerebellum with unbiased stereological techniques. *J Comp Neurol* 326:549–560
- [10] Suzana Herculano-Houzel ,Roberto Lent : Isotropic Fractionator: A Simple, Rapid Method for the Quantification of Total Cell and Neuron Numbers in the Brain, *The Journal of Neuroscience*, March 9, 2005 • 25(10):2518–2521
- [11] *La 3D: Cerveau*, online in: "<http://tpela3d.e-monsite.com/pages/i-la-vision-de-l-homme/cerveau.html>".
- [12] Editeur Belin, *Svt, Iere s*, Manuel de l'élève (édition 2011) DUCO, ANDRE, 2011.
- [13] Koch, N., & Wirsing, M. (2002). The Munich Reference Model for Adaptive Hypermedia Applications An Overview of the Reference Model. In *2nd International Conference on Adaptive Hypermedia and Adaptive Web-based Systems* (pp. 213–222).