

**Conception d'un Tuteur Intelligent en langage
naturel français pour l'enseignement de la
modélisation conceptuelle des données d'un système
d'information**

Salem Yassine

Department. of Computer Science and Engineering
Ferhat ABBAS University - Setif- Algeria
salem.yassine@yahoo.fr

Kharchi Louisa

Department. of Computer Science and Engineering
Ferhat ABBAS University - Setif- Algeria
khlouisa2005@yahoo.fr

Saidi Mohamed

Department. of Computer Science and Engineering
Ferhat ABBAS University - Setif- Algeria
Saidinacir@yahoo.fr

Abstract

In this paper is described the design of software for tutoring intelligent system (TI) whose designed to learning the conceptual model of data of an information system according to the Entity-relationship model (E/R) of Chen[6]. The goal is to permit to learners by resolution of exercises to have the know-how necessary of the elaboration of a conceptual model of data and by this to assimilate the base concepts of the E/R model.

The particularity of the tool is the using the understanding of language natural in the expert module for resolving the exercises while the majority of intelligent tutors that we have studied using it for analysis of dialogs with learners in interface. The strategy proposed makes easy the automatic language processing.

The tool is composed of an expert module and pedagogy module. The expert module is charged to generate the E/R model by resolving the exercises of a theme and the pedagogy module using the student model for to manage the progression learning of the students. The resolution of the set of exercises which are preliminary linked, definite and structured in series by the teacher gives a conceptual model of data corresponding to the theme. These themes with their exercises are in natural language French.

The student proposes a solution for every exercise submitted. The pedagogy module gives the explanation to student when he asked for it from the explanation table developed by the expert module. The student model is given in E/R formalism whose permits to represent the knowledge on the student and his pedagogy progress.

Keywords: automatic processing of natural language, design engineering, information system, intelligent tutoring system, The Entity-relationship model, understanding of natural language.

I Introduction

L'apprentissage du processus de la modélisation d'un système d'information (S.I.) d'une organisation, qui aboutit à un modèle conceptuel de données (MCD) reste un art difficile car :

- Il n'existe pas de démarche algorithmique pouvant générer, à partir des besoins d'une application, le modèle correspondant.
- Pour un même S.I., plusieurs perceptions du monde réel sont développées par les concepteurs, donc plusieurs solutions de modélisation.
- L'expérience dans le domaine n'est acquise qu'à travers l'étude d'un grand nombre de cas.

Une des trois solutions informatiques pour pallier cette difficulté consiste à se faire aider soit d'un système expert (S.E.), soit d'un système d'enseignement assisté par ordinateur (E.A.O.) ou bien d'un système d'enseignement intelligemment assisté par ordinateur (E.I.A.O.).

Ainsi, les systèmes d'E.I.A.O., appelés aussi tuteurs intelligents (T.I.) intègrent des techniques de l'intelligence artificielle (I.A.). Ils sont caractérisés par leur capacité élevée de résolution de problèmes, d'adaptabilité à l'apprenant, d'explication et de pédagogie [20].

Pour enseigner la modélisation des données d'un S.I. d'une organisation, nous devons être en mesure de :

- présenter d'abord les concepts du modèle;
- proposer des exercices et problèmes afin d'inculquer ces concepts à l'apprenant. Lui apprendre à reconnaître et formaliser les connaissances formelles nécessaires, à partir des besoins de l'organisation;
- fournir à l'apprenant, à sa demande, les explications sur la solution ;
- tenir compte de l'évolution de l'apprenant;
- exprimer les exercices en langage naturel comme cela se fait dans l'enseignement classique. Le logiciel n'en sera que plus convivial pour l'apprenant et plus efficace en rendement pédagogique.

Dans ce papier nous décrivons le module expert du tuteur intelligent en langage naturel qui déduit le MCD selon le modèle E/A [6] utilisé notamment dans la méthode Merise [27]. L'étudiant sera évalué par rapport à celui-ci.

II Etude de l'art

II.1 Les systèmes d'enseignement

Tous les systèmes cités sont typiques à notre TI mais aucun d'eux n'a fait du TAL.

Le système GUIDON[8] : avait pour objectif la construction d'un seul programme capable d'enseigner dans différents domaines.

Le système TIMCOD [2] : la validation de la solution de l'apprenant se fait par rapport au champ solution recueilli auprès d'experts pour déterminer les erreurs commises

Le système ARIADE [20] : sa mise en œuvre a été établie à des fins d'enseignement du diagnostic.

Système SACEPAF [19] : c'est un environnement du calcul élémentaire destiné aux élèves de première année élémentaire. Le tuteur expert pédagogique utilise un générateur de leçons et choisit la stratégie pédagogique suivant le profil de l'élève.

Le Tuteur MENTONIEZH [23] : l'évaluation de l'élève se base sur une interaction entre l'élève et le tuteur qui consiste en une série de questions/réponses. L'outil doit être utilisé en présence de l'enseignant. C'est donc un système dépendant.

II.2 Les outils TAL dans l'ingénierie des besoins

La Spécification de Besoins Informatiques (SBI) est une activité fondamentale dans l'ensemble des méthodes de conception et mise en œuvre des systèmes informatiques. Elle détermine en grande mesure la qualité du futur système. Le langage naturel est le moyen de spécification le plus répandu [21] qui se fait par la rédaction des textes comme les cahiers de charges, les cas d'utilisation ou les scénarios.

Les processus d'Ingénierie de Besoins (IB) ont déjà fait appel aux outils de Traitement Automatique de Langues (TAL). Ces outils TAL ont tous une spécificité en commun: faire profiter de la richesse sémantique du texte de SBI pour aider à mieux extraire, modéliser ou valider les besoins des intervenants dans le processus, souvent sous la forme d'un schéma, d'un certain aspect du futur système[16].

On distingue parmi les cinq aspects à modéliser en IB [5] : les modèles organisationnels, les modèles de comportement, les modèles du domaine, les modèles de besoins non fonctionnels, les modèles de données qui décrivent la disposition et les relations des données propres à l'environnement. Souvent, des schémas E/A sont utilisés. Ces modèles offrent des représentations intermédiaires de divers aspects du futur système.

II.3 Compréhension du langage naturel

Un système de compréhension du langage naturel nécessite successivement une analyse morphologique pour déterminer les morphèmes, une analyse lexicale pour identifier les mots, une analyse syntaxique pour avoir la structure syntaxique de la

phrase, une analyse sémantique pour aboutir à une représentation sémantique, une analyse pragmatique pour avoir la représentation pragmatique et enfin faire du raisonnement à l'aide de règles d'inférences et de déduction pour aboutir au sens de la phrase [25].

En général, comme le remarque Gala Pavia [15], les traitements syntaxiques ne sont pas un but en soi : l'analyse syntaxique est ainsi vouée à la création d'outils permettant l'extraction d'information d'ordre linguistique qui pourra être exploitée par la suite par d'autres applications comme par exemple l'extraction d'information [1], la désambiguïsation sémantique [11], les systèmes de question-réponse [13].

II.4 Terminologie et stratégies des outils TAL

L'omniprésence de l'informatique produit une grande diversité de textes de SBI pour tous les domaines imaginables. La terminologie dans ces textes est complètement dépendante du domaine d'application du système: la spécification d'un système de gestion scolaire aura une terminologie très différente de la spécification d'un système de gestion commerciale.

La majorité des approches linguistiques d'aide à l'IB sont fondées soit sur des analyses syntaxiques, soit sur une sémantique dépendante du domaine d'application du système et ne traitent que quelques pages des textes de SBI [16].

Les systèmes TAL d'aide à l'IB partagent tous une même préoccupation : exploiter la richesse linguistique du texte de SBI. Différentes stratégies et approches se distinguent [16] :

- Stratégie TAL légère: Elle s'appuie exclusivement sur le traitement des formes de surface (pas d'analyse morphosyntaxique, pas de grammaire formelle, pas de traitement statistique) [18].

- Les stratégies TAL trop lourdes, qui mettent en jeu des grammaires formelles et des analyseurs morpho-syntaxiques, facilitent la représentation formelle du texte de SBI, mais la complexité ou la lourdeur informatique du traitement les rend inefficaces dans le traitement des textes de SBI de longueur réelle. [12], [24], [14]

- Les approches fondées sur des techniques statistiques : sont les seules à traiter des textes longs. Cependant ces approches se concentrent dans l'extraction d'information lexicale (termes nominaux, candidats à concepts, entités de la base des données) mais manquent des analyses linguistiques fines, indispensables pour la validation des besoins ou la construction des schémas. [22]

- Les approches fondées sur la prise en compte de la sémantique : on a essayé avec des outils de TAL de réduire la multiplicité des interprétations de textes de SBI. Cependant ces outils ne traitent pas le problème de la polysémie verbale, mais des unités linguistiques comme les anaphores, les déictiques, les démonstratifs et les adjectifs.

II.5 L'usage des langages contrôlés

Les langages contrôlés réduisent l'ambiguïté des spécifications et facilitent la représentation formelle du texte de SBI [5]. Les intervenants doivent être habitués

(voire formés) à s'exprimer par écrit avec des restrictions linguistiques. On trouve 15,9 % de documents écrits en langage contrôlé parmi les documents rédigés par les intervenants pour l'analyse de besoins [21].

II.6 Variation lexicale

Dans les textes de SBI, la terminologie change selon la langue de spécialité propre au domaine d'application du futur système. Le lexique qui décrit les entités que le système doit manipuler n'est stable que dans les limites de ce domaine [16].

III Modélisation des données d'un SI selon le modèle E/A [6]

Le modèle E/A suggère de voir toute réalité comme composée d'entités ayant entre elles des associations. Il repose sur les concepts d'entité, d'association, de propriété et de cardinalité.

IV Processus de modélisation du tuteur

IV.1 Démarche pédagogique

Afin de réduire l'ambiguïté des spécifications et faciliter la représentation formelle du texte on fera usage d'un langage naturel contrôlé. Un thème contiendra donc les règles de gestion, le dictionnaire des données et des phrases qui n'entrent pas dans la construction du MCD mais qui informent sur le domaine d'application. C'est à partir du thème que sont extraits les exercices.

Les exercices sont structurés par série. Chaque série poursuit un objectif bien défini de façon à inculquer les concepts du modèle E/A à l'apprenant. Tous les exercices de la série doivent être traités car chacun d'eux contribue à l'élaboration de la solution. Lorsque la troisième série est maîtrisée, les concepts du modèle E/A sont considérés comme acquis de même que le savoir-faire de l'expert.

Ainsi, chaque exercice de la première série permet à l'élève de découvrir une association et les entités qu'elle lie, ceux de la deuxième affectent les propriétés aux entités et associations. Enfin, ceux de la troisième déterminent les cardinalités des associations. Cette démarche est conforme à celle d'un expert. L'expert se basera sur les règles décrites dans [26] pour faire cette construction. Nous avons rajouté la règle de validation suivante :

Un descripteur n'apparaît qu'une seule fois dans le MCD : soit dans une entité, soit dans une association soit comme une entité ou une association.

La première série est considérée comme un pré requis pour les deux autres.

Il s'agit d'un apprentissage incrémental par les exercices.

IV.2 Session d'apprentissage

L'apprenant doit résoudre séquentiellement l'ensemble des exercices de la série. L'évaluation de la solution de l'apprenant pour un exercice se fait par rapport à la solution élaborée par le module expert du TI. La solution du TI est enregistrée avec

le raisonnement qui l'a produite. Des explications sur la solution seront fournies à l'apprenant à sa demande.

Le modèle de l'élève, formalisé selon le modèle E/A, enregistre l'évaluation et la progression de chaque apprenant.

IV.3 Description de l'énoncé des exercices

Dans le modèle E/A, l'analyse du discours montre qu'il y a des déductions à faire selon la structure grammaticale de la phrase pour déterminer les «entités», les «associations», les « propriétés », les « cardinalités » ainsi que la possibilité de rattacher les « propriétés » aux «entités » et «associations».

De l'étude d'un recueil d'exercices proposés aux étudiants, nous avons dégagé les modèles de phrases les plus fréquemment utilisées qui sont faciles à comprendre par l'apprenant et à analyser par l'outil.

V Description du tuteur

V.1 Objectifs

Le tuteur est axé sur la pragmatique pour développer chez l'apprenant les mécanismes pratiques et suffisants à la mise en œuvre des concepts du modèle E/A.

Les exercices d'un thème représentent des règles de gestion et décrivent les contraintes à respecter par le concepteur. En ajoutant les propriétés organisées dans un dictionnaire de données, on dispose du matériel de base pour l'édification d'un MCD [10].

V.2 Architecture

Le tuteur est organisé autour des deux modules principaux :

- Le module expert, qui fait l'objet de cet article, résout les exercices et enregistre les résultats avec leurs explications. Il utilise une stratégie TAL légère.
- Le module pédagogue gère la session d'enseignement. Il utilise un modèle de l'élève qui rassemble les informations sur le profil de chaque élève.

A ces deux modules, il faut rajouter l'interface utilisateur.

La spécification des fonctionnalités de chaque module est ainsi mieux définie. L'évolution et la maintenance du système deviennent locales et ne remettent pas en cause la cohérence globale lors d'une modification.

V.3 Module expert

Chaque exercice est défini par le numéro de thème, le numéro de série, le numéro dans la série, son énoncé et sa note. Le module expert résout les exercices et fournit l'explication de la solution, en se basant sur la compréhension du LN. Celle-ci consiste à découvrir les entités et les associations avec leurs propriétés, les cardinalités et la traduction de l'ensemble dans une représentation interne, indépendante de toute langue, connue sous le nom de représentation de sens et qui correspond dans notre cas au modèle conceptuel E/A. Notre système de compréhension est schématisé par la figure n°1.

Le module expert dispose d'une table des associations, d'une table des entités et d'une table des explications qu'il complète au fur et à mesure qu'il résout les

exercices du thème. L'explication est sauvegardée pour la présenter à l'apprenant en cas où il la demande

Nous avons privilégié un langage contrôlé. En effet, l'exercice qui correspond à une règle de gestion est explicité d'une manière très simple généralement par le triplet « sujet, verbe, compléments ». Le sujet et les compléments sont le plus souvent des groupes nominaux. Le verbe est à la forme passive ou active, toujours à la troisième personne du singulier ou du pluriel du présent de l'indicatif. Nous n'avons, cependant, pas traité les nombres cardinaux dans les phrases puisqu'ils sont équivalents au singulier et pluriel des groupes nominaux lesquels sont suffisants pour déterminer les cardinalités des associations du MCD.

Cette simplification induit que toutes les formes possibles des mots n'apparaîtront pas toutes dans les phrases à analyser entraînant ainsi un vocabulaire réduit, peu ambiguë et facile à comprendre par l'élève et à analyser par le module.

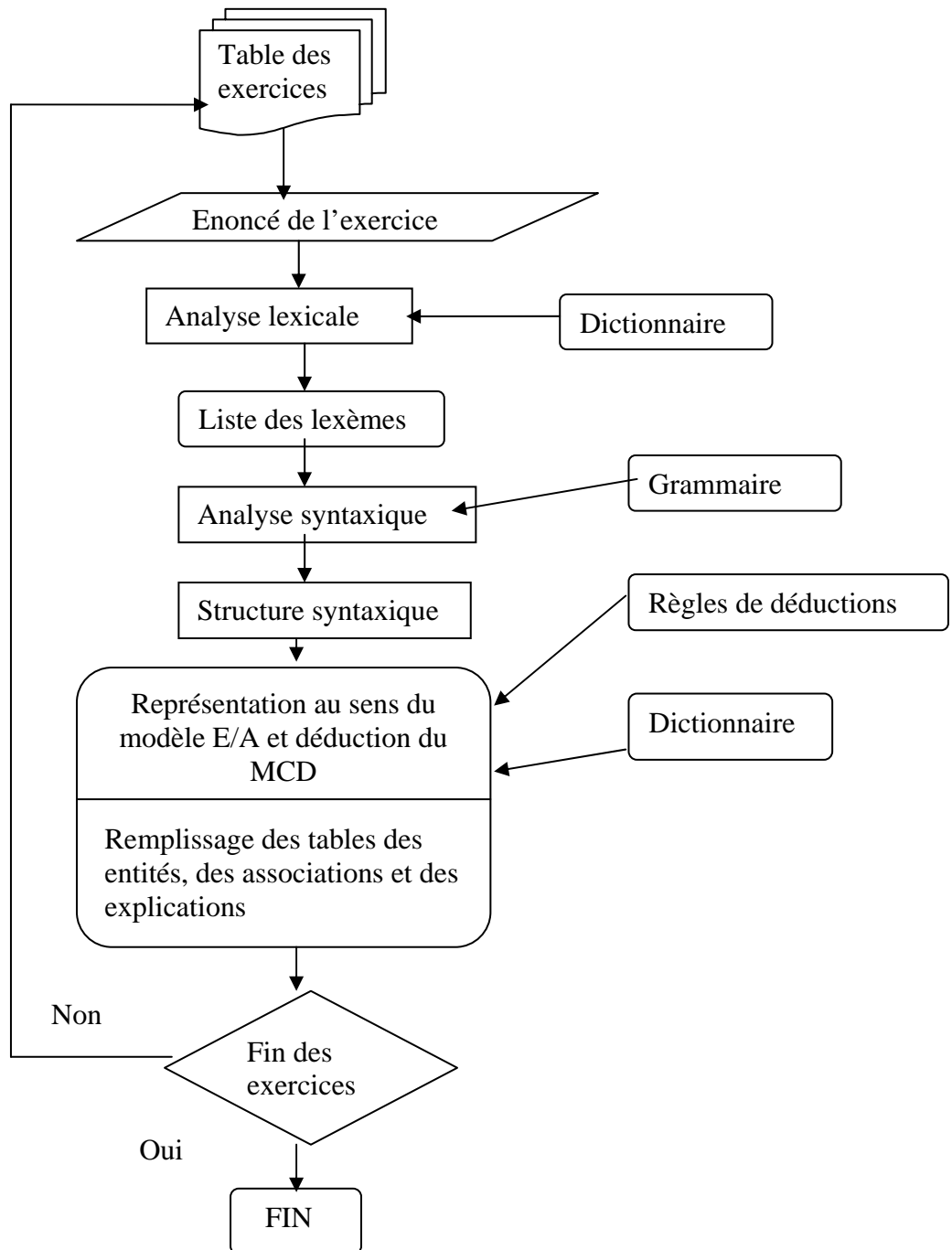


Figure n°1 : Architecture du module expert

V.4 Analyseur lexical

Il dispose dans le dictionnaire de la forme fléchie des mots pour accélérer la recherche.

V.4.1 Dictionnaire

Il est constitué de:

- Un dictionnaire indépendant du domaine d'application qui renferme les constituants de base du français tels que les déterminants, les conjonctions, les adverbes, les auxiliaires être et avoir, etc. Nous rajoutons, pour les besoins de la conception, les verbes qui traduisent une dépendance fonctionnelle tels que appartenir, dépendre, déterminer, ...
- Un dictionnaire des mots propres au domaine d'application dont on veut établir le modèle conceptuel.

V.4.2 Liste de lexèmes

L'analyseur lexical produit comme résultat une liste de lexèmes. Chaque lexème est muni d'informations extraites du dictionnaire et utilisées lors de l'analyse syntaxique, telles que genre, nombre, le nom du verbe, ...ainsi que d'un champ « affectation » qui indique si la donnée a été placée ou non dans une entité ou association par le module expert.

Exemple : « un enseignant enseigne plusieurs modules »

Le résultat de l'analyseur lexical sera :

Mot	Forme Canonique	Catégorie Lexicale	Champ1	Champ2	Champ3	Affe ctati on
un	un	article	masculin	singulier	indéfini	
enseignant	enseignant	nom	masculin	singulier	-	
enseigne	enseigner	verbe	transitif	singulier	enseignement	
plusieurs	plusieurs	adjectif indéfini	-	pluriel	indéfini	
modules	module	nom	masculin	pluriel	-	

V.5 Analyseur syntaxique

L'analyse syntaxique se charge de reconnaître la structure de surface de la phrase à traiter. Elle permet de distinguer le sujet et les compléments du verbe. Pour mener à bien cette analyse, il nous faut deux éléments complémentaires : la grammaire et la stratégie [3].

V.5.1 Description de la grammaire

La grammaire utilisée est une grammaire formelle de type 2 [7] suffisante dans un domaine restreint. Elle est décrite par le quadruplet :

$G = (V_n, V_t, R, S)$ avec

$V_n = \{\text{catégories lexicales, catégories syntaxiques}\}$;

$V_t = \{\text{mots utilisés}\}$;

$R = \{\text{règles de production}\}$;

S est l'axiome c'est-à-dire la phrase.

Nous avons défini dans la catégorie syntaxique le « mot composé » construit à partir de mots simples. Nous avons construits quatre mots composés : Nom à Nom, Nom de Nom, Nom Nom, Nom adjectif.

V.5.1.2 Description des règles de R

Les règles de grammaire devront valider les phrases suivantes :

Phrase := Gp_nom, Gp_verb / Pronom indéfini, Gp_verb

Pour pouvoir déduire les cardinalités des associations, lors de l'étape de représentation de sens, ont été précisés :

Adjectif_seul := « seul » / « seule » / « seuls » / « seules »

Adverbes := « au moins » / « au plus »

V.5.2 Définition de la stratégie adoptée

Notre analyseur utilise une stratégie ascendante des données vers les règles. Les règles sont définies indépendamment de leurs traitements. Ceci permet au concepteur de modifier son ensemble de règles sans accéder aux procédures qui les manipulent. Le résultat de l'analyse est un arbre syntaxique

Exemple :

Soit la phrase : « Un enseignant enseigne plusieurs modules »

L'analyseur syntaxique produit l'arbre syntaxique suivant :

phrase(gr_nom(det(un, indéf, sing), nom (enseignant, sing)), gr_verb(verbe (enseigne, trans), gr_nom(det(plusieurs, indéf, plur), nom(module, plur)))).

V.6 Modèles d'exercices

Nous présentons dans des tableaux les modèles de phrases que doivent respecter les exercices proposés avec les résultats que doit délivrer le module expert.

Annotation utilisée

Cp = complément; **Det** = déterminant; **Prep** = préposition; **Nom-Ass** = nom de l'association; **Card** = cardinalité minimale et maximale de l'entité i; **[]** = optionnel; **/** = ou bien.

V.6.1 Première série : Détermination des entités et de l'association.

Dans cette série, la restriction est que le type du déterminant doit être indéfini pour reconnaître une entité. Les associations étant déterminées par le verbe, l'enseignant doit utiliser des verbes différents dans les exercices.

Modèle de phrase	Résultat	Exemple
Sujet verbe Cp1 [Cp2]	Nom_Ass = verbe/verbe + prep Entité1 = sujet Entité2 = Cp1 Entité3 = vide / Cp2	*un enseignant enseigne plusieurs modules dans un département
« on » verbe Cp1 Cp2 [Cp3]	Nom-Ass = verbe Entité1 = Cp1 Entité2 = Cp2 Entité3 = vide / Cp3	* on affecte une et une seule salle à un enseignant pour une classe
Sujet verbe Det « fois » / Sujet verbe Det « fois » Cp1 Remarque : Det doit être au pluriel, le verbe peut être à la forme passive	Nom-Ass = verbe / être verbe(forme passive) Entité1 = sujet Entité2 = nom déduit du verbe Entité3 = vide / Cp1	* un employé s'absente plusieurs fois * un employé est sanctionné plusieurs fois dans un service
« on » verbe Cp1 Det « fois » / « on » verbe Cp1 Det « fois » Cp2	Nom-Ass = verbe Entité1 = nom déduit du verbe Entité2 = Cp1 Entité3 = vide /Cp2	*on sanctionne un employé plusieurs fois * on sanctionne un employé plusieurs fois dans un service

V.6.2 Deuxième série : Affectation des propriétés

Dans cette série, trois types de modèles ont été définis, chacun conduisant à une déduction. Le premier type affecte une propriété à une entité, le deuxième reconnaît une propriété comme étant une entité et le troisième exprime une dépendance fonctionnelle désignant l'association à laquelle sera affectée la propriété.

Modèle de phrase	Résultat	Exemple
Type 1		
Sujet verbe Cp1	Entité = sujet Propriété = Cp1	* un enseignant possède un grade
Sujet verbe (à la forme passive) prep (à) Cp1	Entité = Cp1 Propriété = sujet	* un grade est affecté à un enseignant
« on » verbe Cp1 prep (à) Cp2	Entité = Cp2 Propriété = Cp1	* on affecte un grade à un enseignant
Type 2		
Sujet verbe Cp1	Nom-Ass = verbe Entité1 = Sujet Entité2 = Cp1	* un enseignant a plusieurs échelons
Type 3		
Sujet verbe Cp1 Cp2 [Cp3] Remarque : Verbe = dépendre / être déterminé[e]	Nom-Ass = verbe Propriété = sujet Entité1 = Cp1 Entité2 = Cp2 Entité3 = vide /Cp3	* une note dépend du module et de l'étudiant
Sujet verbe Cp1 prep(par) Cp2	Nom-Ass = verbe Propriété = Cp1 Entité1 = Sujet Entité2 = Cp2	* un étudiant a une note par module

V.6.3 Troisième série : Enseignement des cardinalités

Un exercice permet de déduire la cardinalité minimum et maximum d'une seule entité de l'association à la fois. Le nombre de phrases pour déterminer les cardinalités de l'association doit être égal à la dimension de l'association. Dans un but pédagogique, la construction de la phrase sera simple et unique. Les nombres cardinaux n'ont pas été traités puisqu'ils sont équivalents au singulier et au pluriel des groupes nominaux qui suffisent pour la conception.

Modèle de phrase Sujet verbe Cp1		Résultat	Exemple
		Nom-Ass = verbe Entité1 = Sujet Entité2 = Cp1	
Det	du Det du Cp1		
sujet			
singulier	singulier	Card1 = 1, 1 Card2 = vide	* un enseignant est affecté à un département
singulier	singulier	Card1 = 0, 1 Card2 = vide	* un étudiant a une ou aucune bourse
singulier	pluriel	Card1 = 1, n Card2 = vide	* un enseignant enseigne plusieurs modules
pluriel	singulier	Card1 = vide Card2 = 1, n	* plusieurs fonctions sont attribuées à un enseignant
singulier	numérique « au plus un »	Card1 = 0, 1 Card2 = 1, n	* un employé possède au plus un véhicule
singulier	numérique « au moins un »	Card1 = 1, n Card2 = vide	* un étudiant est inscrit dans au moins un module

Remarques :

1) Pour le type de phrase « On verbe Cp1 Cp2 » ou « on » est le pronom indéfini, les mêmes règles s'appliquent sur le Cp1 et le Cp2 qui deviennent respectivement le Sujet et le Cp1.

Exemple : on affecte un enseignant à un département.

2) La cardinalité minimale égale à zéro s'obtient par la présence de l'adjectif indéfini « aucun » ou « aucune » dans le déterminant ou bien le déterminant égal à « au plus un ».

V.7 Module pédagogique

Le module pédagogique gère l'apprentissage de l'apprenant. Il utilise le modèle de l'élève qui tient compte de ses réponses aux exercices et de ses demandes d'explications. Il consiste en un ensemble d'actions pédagogiques qui aident l'apprenant à progresser. Comme par exemple :

Règle 1 : Si l'apprenant demande des explications alors afficher l'explication élaborée par le module expert.

Règle 2 : Si l'apprenant indique la fin de la série alors évaluer la série.

V.7.1 Dictionnaire des synonymes

Un dictionnaire de synonymes, caché à l'étudiant, servira au module pédagogique pour l'évaluation de l'étudiant dans le cas où il proposera une solution avec des données synonymes.

V.7.2 Le modèle de l'élève

Notre choix s'est porté sur le modèle de connaissances partielles (overlay model) étant donné que les connaissances de l'élève sont considérées comme un sous-ensemble des connaissances de l'expert. En effet, notre objectif consiste à faire découvrir et faire assimiler les connaissances de l'expert qui manquent à l'élève mais non à chercher les erreurs pour en effectuer des diagnostics complexes.

Le modèle de l'élève sera un modèle E/A qui tiendra compte des notes obtenues dans les exercices et de la progression de chaque élève.

V.8 Description des interfaces

Pour mener à bien la mission du module pédagogique, une interface apprenant sera développée. Un ensemble d'actions pédagogiques de cette interface ont été définies : identifier l'apprenant, proposer des thèmes à l'apprenant, appel au module expert pour la résolution, présenter les séries à l'apprenant, comparaison de la solution apprenant avec celle du système, enfin noter et évaluer l'apprenant.

L'interface de l'enseignant servira pour saisir les thèmes, les séries d'exercices, le dictionnaire de données.

L'apprenant aura à remplir des tableaux identiques à ceux élaborés par le module expert. La solution apprenant sera considérée comme juste ou fautive par rapport à la solution du module expert. Les explications seront affichées s'il y a une différence entre les deux solutions et/ou si l'apprenant le demande.

VI Etude de cas

Les séries sont extraites à partir du thème. A titre d'exemples sont écrits par l'enseignant les exercices suivants dans chacune des séries :

Première série :

1-1-1 Un magasin vend plusieurs articles.

Deuxième série :

1-2-1 Un article a un prix et une désignation.

Troisième série :

1-3-1 Un article est vendu par plusieurs magasins.

Le module expert construit le MCD avec des explications au fur et à mesure de l'analyse des séries.

A la fin de la résolution de la première série, le système aura créé la table des associations, la table des entités et celles des explications. Au fur et à mesure de l'analyse des exercices de la deuxième série, le module expert placera les propriétés soit dans les entités ou les associations quand une règle lui aurait permis de faire cette déduction tout en mettant à jour la « marque d'affectation » dans le dictionnaire. La marque d'affectation indique si la propriété est déjà placée dans la table des

entités ou celle des associations. Avec les exercices de la troisième série, le module expert complète la table des associations par les cardinalités. Ainsi, en considérant les exercices ci-dessus, les tableaux produits par le module seront :

Table des entités

Nom entité	propriétés
Magasin	
Article	désignation, prix

Table des associations (lire verticalement)

Num-ass	1
Nom-ass	vendre
Propriétés	
Entité1 Card-min, card-max	magasin 1,n
Entité 2 Card-min, Card-max	article

Table des explications

Num-exo	Explication
1 1 1	« Vendre » détermine l'association qui existe entre « magasin » et « article ».
1 2 1	Le verbe « a » indique une dépendance fonctionnelle. « désignation » est attribut de « article ». Il se trouve dans le dictionnaire de données. « prix » est attribut de « article ». Il se trouve dans le dictionnaire de données.
1 3 1	Une occurrence de « article » participe à l'association « vendre » plusieurs fois donc 1,n

VII Conclusions

Pour enseigner les concepts de base du modèle E/A nous avons conçu un tuteur intelligent qui analyse et interprète les textes descriptifs des problèmes écrits en LN. Les exercices rédigés en langage naturel contrôlé rendent l'outil plus efficace et plus naturel. Les règles de déduction que nous avons élaborées sont issues de nos expériences d'enseignant et indépendantes du domaine à modéliser. Les explications

affichées à l'apprenant sont extraites à partir de ces règles pour lui apprendre à distinguer dans un texte ce qui est nécessaire dans la conception. Plus les phrases sont simples mieux elles sont compréhensibles et facilitent l'abstraction que l'apprenant doit en faire pour concevoir son MCD.

L'originalité de cet outil est qu'il fait de la compréhension du langage naturel pour déduire le modèle E/A. Le fait que les textes soient proposés par des enseignants, que tout exercice respecte une syntaxe, a un sens et un but dans la démarche de conception montre que seule une analyse lexico-syntaxique sans analyse sémantique du texte suffit pour aboutir à une représentation du schéma E/A.

Le modèle de l'élève de type overlay utilisé par le module pédagogique répond bien à notre besoin.

Le tuteur est essentiellement axé sur la pragmatique. Il vise à développer le savoir faire et le savoir analyser dans la modélisation conceptuelle des données. Ses avantages sont :

- Il pourra être utilisé comme un outil intelligent d'aide à la conception de MCD selon le modèle E/A de SI.
- Il fera bénéficier l'apprenant de notre expérience en matière d'enseignement de la modélisation et d'encadrement de projets.
- Notre stratégie TAL facilite la représentation formelle du texte et simplifie le traitement informatique. Elle optimise le traitement du LN par l'élimination des phases d'analyse morphologique, sémantique et pragmatique puisque les formes du mot varient très peu et les exercices ne peuvent être dépourvus de sens sachant qu'ils sont proposés par l'enseignant. De plus, le but n'est pas de détecter des erreurs linguistiques mais plutôt conceptuelles. Les fonctions syntaxiques des différents composants de la phrase sont suffisantes pour déduire la représentation au sens E/A.

Actuellement la représentation du MCD se fait textuellement sous une forme tabulaire prédéfinie pour décrire les entités et associations. Nous envisageons de traduire les tableaux des entités et des associations déduits en graphisme pour être en conformité avec le formalisme de CHEN.

Il est tout à fait possible de prolonger ce travail pour en déduire automatiquement le modèle relationnel [9] par l'application des règles de passage proposées par le groupe CODASYL [17].

Nous comptons reprendre les résultats de ce travail pour établir les diagrammes de classes d'UML [4].

REFERENCES

- [1] Appelt D., Hobbs J., Bear J., Israel, D. et Tyson, M.: « FASTUS: a finite-state processor for information extraction from real-world text ». In *Proceedings of IJCAI-93*, Chambéry, 1993.
- [2] Benslimane A, Feki T, Marciano J P. : « TIMCOD : Un outil d'apprentissage de la modélisation conceptuelle de données ». Congrès INFORSID Clermont-Ferrand. France. 1992

- [3] Berrard-Dugourd Anne : « Rôle d'une analyse syntaxique dans un système de compréhension du langage naturel ». Thèse de doctorat d'état. Université Paris VII. France. 1990
- [4] Booch G., Rumbaugh J., Jacobson I.: « Le guide de l'utilisateur UML. » Editions Eyrolles 2001
- [5] Burg J.F.M. : «Linguistic instruments in requirements engineering. » IOS Press, Amsterdam. 1997
- [6] Chen P: « The Entity-relation ship model Towards a Unified View of data.» ACM TODS, Vol. 11, 1976
- [7] Chomsky N.: « Syntactic structures ». The Hague : Mouton. 1957
- [8] Clancey W J. : « Guidon ». In Journal Computer Based Instruction. Vol. 10 N°1 et N° 2. 1983
- [9] Codd E.F.: « A Relational model of Data for Large Shared Data Banks communication », ACM, vol N° 6, 1970
- [10] Collongues A.: « Merise 2 : Etudes et Exercices.» Editions Dunod. 1989
- [11] Dini, L., Di Tomaso V. et Segond F. : Ginger II, an example-driven word sense disambiguator. In *Computers and Humanities, special issue*, 1999.
- [12] Fabbrini F., Fusani M., Gnesi S., Lami G. : « An automatic quality evaluation for natural language requirements ». Seventh International Workshop on requirements Engineering Foundation for Software Quality.(REFSQ'2001)
- [13] Ferret O., Grau B., Hurault-Plantet, M., Illouz, G. et Jacquemin, C.: « Document selection refinement based on linguistic features for QALC, a Question Answering system ». In *Proceedings of Recent Advances in Natural Language Processing, RANLP-2001*, Tzigov Chark, Bulgarie, 2001.
- [14] Fliedl G., Kop C., Mayr H.C. : «From scenarios to KCPM dynamic schemas : aspects of automatic mapping », Proceedings of the 8th Internatinal Conference on Applications Natural Language and Information Systems (NLDB'2003), (A. Diisterhiift and B. Thaiheim, eds.), GI Edition, Allemagne, pp. 91-105.
- [15] Gala Pavia N. : « Un modèle d'analyseur syntaxique robuste fondé sur la modularité et la lexicalisation de ses grammaires ». Thèse de doctorat. Orsay France 2003
- [16] : Garcia-Flores J. : « Annotation sémantique des spécifications informatiques de besoins par la méthode d'Exploration Contextuelle : une contribution de méthodes

linguistiques aux conceptions de logiciels ». Thèse de doctorat. Laboratoire LALICC. Université Paris IV. Sorbonne. 2007

[17] Gardarin G.: « Bases de données ». Editions Eyrolles. 2001

[18] Gervasi V., Nuseibeh B. : « Lightweight validation of natural language requirements », *Software Practice and Experience*, N° 32, p. 113-133, 2002

[19] Laskri M.T., Bouhadada T. « SACEPAF : A learning environment of elementary calculus intelligently aided by a computer ». *Current Trends in computer Science and Information System, CTCSIS' 96*, Jordan, July, 9- 10, 1996.

[20] Lecllet D. : « Une approche par plans et par modélisation du domaine appliquée à l'enseignement de la rhumatologie. » *Le système ARIADE, Apprentissage de la Rhumatologie Intelligemment Assistée par Ordinateur* ». Thèse de doctorat. Université de technologie de Campiegne 1993

[21] : Mich L., Franch M., Novi P. : « Requirements analysis using linguistic tools : Results of an on-line survey ». *Rapport technique 66, Department of Computer Management Sciences. Università de Trento. 2003*

[22] Natt Och Dag J., Regnell B., Carlshamre P., Andersson M., Karlsson J. : « Evaluation Automated support for requirements similarity analysis in market-driven development », *Seventh International Workshop on Requirements Engineering Foundation for Software Quality (Switzerland), June 2001.*

[23] Py. Dominique, Carpentier F. G. « E.I.A.O. » INRIA. Interaction homme-machine, images, données, connaissances. 3A : bases de données, bases de connaissances, systèmes cognitifs. *Rapport d'activité. 1998*

[24] Rolland C., Proix C.: « A natural language approach for requirements engineering » *Avanced Information System Engineering (Lecture Notes in Computer Science), Springer Verlag, Paris, 1992*

[25] Sabah Gerard : « L'intelligence artificielle et le langage : Représentation des connaissances ». Editions Hermes. Paris 1989

[26] Schneider M: « Bases de données: Première partie ; Modèles et Systèmes ». CUST Université Blaise Pascal France cours 1988-1989

[27] Tardieu H., Rochfeld A., Colletti R. « La Méthode Merise. Principes et outils ». Les éditions d'organisation. Paris 1986

Received: October, 2008