

**UNA PROCEDURA PER VALUTARE “AUTOMATICAMENTE” IL
LIVELLO DI APPRENDIMENTO DEGLI STUDENTI**

Roberto Pirrone

Università degli studi Palermo – Dipartimento Ingegneria Informatica
Viale delle Scienze – Edificio 6 piano 3 Palermo
pirrone@unipa.it, +39 0917027525

Vincenzo Cannella

Università degli studi Palermo – Dipartimento Ingegneria Informatica
Viale delle Scienze – Edificio 6 piano 3 Palermo
cannella@csai.unipa.it, +39 0917027526

Giuseppe Russo

Università degli studi Palermo – Dipartimento Ingegneria Informatica
Viale delle Scienze – Edificio 6 piano 3 Palermo
russo@csai.unipa.it, +39 0917027528

Résumé : Les Systèmes Tuteur Intelligent (STI) sont programmes basés sur l'ordinateur capable de présenter documents éducatives aux utilisateurs dans une manière flexible et personnalisée. L'évaluation automatique des étudiants est devenue un sujet d'intérêt émergent dans cette area de recherche. Le but de ce travail est la présentation de une procédure de évaluation capable de déterminer quelle que l'étudiant connaît sur a certain sujet, et de obtenir un chemin dans les matériels didactiques pour gagner la connaissance nécessaire à l'objectif.

Abstract: Intelligent Tutoring Systems (ITS) are computer programs which are able to present educational materials to users in a flexible and personalized way. Performing automatic assessment of the student has become an emergent topic of interest in this research area. The purpose of this work is to present an assessment procedure that is able to determine what the student knows about a certain query topic, and to derive a path in the learning materials to gain the knowledge needed to the goal.

Riassunto : Gli Intelligent Tutoring Systems sono programmi in grado di presentare documenti necessari all'apprendimento in una maniera flessibile e personalizzata. La valutazione automatica dello studente è diventato un campo di interesse emergente all'interno di questa area di ricerca. Lo scopo di questo lavoro è la presentazione di una procedura per la valutazione dello studente in grado di determinare il grado di conoscenza dello studente su una certa materia e di ottenere un percorso personalizzato all'interno del materiale didattico al fine di ottenere la conoscenza necessaria al raggiungimento dell'obiettivo.

Mot-clés : Système Tuteur Intelligent, évaluation, ontologie,

Keywords: Intelligent Tutoring Systems, assessment, ontology, natural language interaction

Parole Chiave: Intelligent Tutoring Systems, valutazione, ontologia, interazione in linguaggio naturale

1 INTRODUZIONE

Il processo di valutazione all'interno di sistemi informatici che effettuano un'interazione per aumentare le conoscenze degli attori coinvolti può essere visto come il problema di dare una misura alle competenze, alle credenze, alle motivazioni ed ai comportamenti presenti e futuri degli studenti che interagiscono con il sistema. In generale ciò comporta la definizione di un modello dello studente come elemento comparativo che tenga conto degli errori e delle variabilità che l'interazione comporta. La natura del problema della valutazione "automatica" comporta l'utilizzo di procedure che siano in grado di avere al loro interno una capacità di fronteggiare alcuni elementi di incertezza nell'interazione con gli studenti. Questo lavoro è particolarmente incentrato nella definizione di una procedura per la valutazione "automatica" delle risposte degli studenti all'interno di un Intelligent Tutoring System (ITS). Gli ITS sono strumenti che si affiancano ai sistemi tradizionali per aiutare nell'apprendimento di determinati concetti. Negli anni passati si sono avuti molti studi in questo campo (Wainer 1990), ma lo sviluppo di interazioni più complesse basate su interfacce ricche di contenuti ha naturalmente aumentato il livello di difficoltà nell'automazione del processo. Tutti e tre i principali paradigmi per gestire la variabilità sono stati utilizzati all'interno degli ITS: le reti bayesiane, la teoria della fusione delle informazioni di Dempster e Shafer e la logica fuzzy. Per una relazione dettagliata si veda (Jameson 1996). Comunque negli ultimi periodi l'utilizzo delle reti bayesiane ha preso il sopravvento sugli altri paradigmi. In questo lavoro si propone un approccio basato sull'utilizzo della Knowledge Space Theory (KST) (Falmagne et al. 1990), che è focalizzata sulla possibilità di definire possibili percorsi di apprendimento all'interno del dominio di interesse. Iniziamo adesso ad esaminare alcuni ITS presenti in letteratura: nel sistema BITS (Butz et al. 2004) l'utilizzo delle reti bayesiane definisce le relazioni di precedenza tra gli elementi di uno specifico dominio. Il sistema ANDES (Conati et al. 2002) aiuta lo studente nell'apprendimento della meccanica newtoniana. Questo sistema valuta le conoscenze dello studente in relazione alla sua capacità di rispondere ad un problema dato. Per ottenere tale scopo si utilizzano più reti bayesiane per memorizzare sia la conoscenza generale che quella specifica per un determinato problema. Questo procedimento è ottenuto automaticamente come risultato di uno strumento che risolve problemi di fisica basandosi sulle regole proprie del dominio. Il nostro sistema invece si basa sulla KST per definire lo stato di conoscenza dello studente e muove da questo per intraprendere le azioni necessarie ad interagire con lo studente. Il principio di funzionamento nella definizione dello stato di conoscenza basato sulla KST è stato affrontato in letteratura: Dowling e Hockemeyer propongono un approccio deterministico basato sulla definizione di diagrammi a stati finiti (Dowling 2002), mentre un approccio probabilistico di può trovare in (Cosyn 2000). I riferimenti visti precedentemente dimostrano che entrambi gli approcci possono essere utilizzati per la definizione di un dominio di conoscenza basato su elementi legati da relazioni di precedenza. In aggiunta a ciò il nostro lavoro propone l'utilizzo di una ontologia di dominio (Guarino 1995), all'interno della quale le relazioni tra gli elementi del dominio sono utilizzate sia per definire le relazioni strutturali tra gli elementi del dominio che quelle di precedenza necessarie per la definizione della valutazione del livello di apprendimento dello studente. In tal modo si tenta di definire l'attitudine dell'insegnante nella spiegazione dei concetti necessari alla comprensione di uno specifico argomento: in quest'ambito l'ordine di comprensione di alcuni concetti è dettato obbligatoriamente dalla natura dell'argomento trattato, mentre altri concetti possono essere affrontati in maniera facoltativa. Per definire tale organizzazione della base di conoscenza due predicati particolari sono stati introdotti all'interno dell'ontologia: il primo definisce le relazioni di "prerequisito stretto" mentre il secondo definisce quelle di "prerequisito debole". In questo modo si può definire un percorso di apprendimento personalizzato per lo studente. Il secondo legame vuole introdurre quindi gli elementi di variabilità presenti nel momento in cui un docente calibra la lezione sulle conoscenze specifiche della classe che ha davanti. La procedura di valutazione parte dall'analisi della domanda che lo studente pone al sistema per avere spiegato un argomento che intende approfondire. Il sistema aggancia la domanda ad un nodo dell'ontologia e da lì inizia la conversazione. Tale procedura è stata realizzata come componente di un ITS (Pirrone et al. 2005) avente lo scopo di approfondire lo studio del linguaggio di programmazione Java. Il resto del lavoro è organizzato nella maniera seguente: la prossima sezione introduce alcuni approfondimenti sulla KST. La sezione seguente presenta il sistema in generale, mentre i due successivi parlano dei principali moduli che la compongono più approfonditamente. La sezione seguente descrive

la procedura di valutazione dello studente. Nella seguente si effettuano alcune considerazioni sulle motivazioni che hanno portato alle scelte nella definizione del sistema anche in relazione all'evoluzione nei sistemi di apprendimento a distanza. Infine si presentano le conclusioni ed il lavoro a venire.

2 LA KNOWLEDGE SPACE THEORY

La Knowledge Space Theory (KST) è stata proposta da Doignon e Falmagne (Falmagne et al. 1990) (Albert et al. 1997) al fine di descrivere la struttura di un dominio di conoscenza. Nel formalismo definito dalla KST un *insieme di conoscenza* è rappresentato come insieme di oggetti che possono indifferentemente essere nozioni o capacità. Lo studente si considera come in grado di conoscere un determinato argomento se ad un certo momento ha la conoscenza di tutti gli oggetti che lo definiscono. Così come definito da Albert (Albert et al. 2002), un dominio è costituito da tutti gli oggetti legati da relazioni di precedenza. Si considera che uno studente “ha appreso” un dominio quando è in grado di rispondere correttamente a problemi relazionati agli oggetti componenti. In tal modo ogni studente si può definire a partire dal suo *stato di conoscenza*, cioè la collezione degli oggetti su cui, ad un certo istante, lo studente sa rispondere affermativamente. Ovviamente lo stato di conoscenza varia nel tempo. La collezione dei possibili stati all'interno di un dominio si chiama *spazio di conoscenza*. Dato un dominio di conoscenza Q si definisce il sottoinsieme dei possibili problemi in cui è possibile scomporre il problema di partenza tali che risulti: $Q = \{q_1, q_2, \dots, q_n\}$. Se è possibile definire un operatore di precedenza \leq tale che per ogni q_i, q_t risulti $q_i \leq q_t$ quando $i \leq t$, allora il problema che stiamo considerando è scomponibile secondo la KST e l'insieme Q sarà definito come un insieme supposto ordinato. In altre parole due oggetti q_i, q_t sono legati da una relazione di precedenza se e solo se si può desumere la conoscenza del concetto q_i a partire dal concetto q_t . Inoltre sarà possibile trovare una formulazione di un particolare stato K detto *spazio di conoscenza* (Knowledge Space) tale che risulti $K \subseteq Q$; K è uno stato $\Leftrightarrow (\forall q_1, q_2 \in Q, e q_1 \in K \Rightarrow q_2 \in K)$. Dato un insieme di problemi q_i ,

che rispondono alla relazione vista precedentemente, l'insieme $K \subset 2^Q$ degli stati di conoscenza è detto *struttura di conoscenza* (Knowledge Structure) del problema dato. Si può dimostrare che tutti i possibili spazi di conoscenza si possono ordinare a partire dalla relazione di precedenza (Hart 1995). Tale spazio di conoscenza K_s contiene ovviamente l'insieme vuoto \emptyset e l'insieme di tutti gli oggetti come elementi Q . L'insieme di tutti i possibili stati di conoscenza è definito a partire dalla relazione di precedenza definite per legare gli oggetti componenti. Una sequenza diretta di stati che, a partire da uno stato di partenza K , aggiunge un solo oggetto e quindi porta a definire un nuovo stato di arrivo K' si definisce come un componente di un possibile percorso di apprendimento. In questo modo si definisce un insieme di stati di conoscenza annidati che ricoprono il dominio considerato. L'oggetto q che aggiunto allo stato K definisce lo stato K' appartiene per definizione alla frontiera esterna di K ed a quella interna di K' . Per una più approfondita definizione degli aspetti della KST si può consultare ad esempio (Falmagne et al. 2003).

3 IL SISTEMA

Il sistema è scomponibile in due sottosistemi fondamentali (vedi Fig. 1): il primo contiene gli elementi atti alla valutazione delle competenze dello studente ed il secondo invece l'insieme dei componenti per la visualizzazione del materiale didattico. Ognuno di questi sottosistemi si presta ad una differenziazione degli elementi che lo definiscono. Il modulo di valutazione è composto dal modulo *chatbot* che si occupa del processo di interazione dello studente in linguaggio naturale, dal *modulo LSA* (Latent Semantic Analysis) che definisce la metodologia di comprensione dei termini della conversazione e dall'*ontologia* in cui vengono memorizzati i concetti del dominio di appartenenza. Il modulo di visualizzazione permette agli studenti la navigazione del materiale didattico, tale modulo è composto dalla *mappa* che è il front-end di visualizzazione e da un *modulo* che organizza i documenti del dominio basato sulle *SOM* (Kohonen 1990).

La prima interazione avviene tra lo studente ed il sottosistema di valutazione: lo studente pone al sistema una domanda circa un argomento che egli intende approfondire. Tale interazione avviene in

maniera intuitiva ed in linguaggio naturale da parte dello studente. Il modulo che si occupa della conversazione con lo studente è una versione modificata del chatbot detto “Program D” (Coursey 2004), una versione realizzata in linguaggio Java del chatbot conosciuto come ALICE (Alice). Questo chatbot è un interprete del dialogo che utilizza un formalismo di rappresentazione detto AIML (Artificial Intelligence Markup Language). A partire dalla conversazione il sistema proietta i termini più significativi all’interno di uno spazio di rappresentazione degli elementi ottenuto con la tecnica LSA (Laundauer et al. 1998). Anche i termini dell’ontologia sono proiettati nello spazio semantico e quindi dalla vicinanza tra i termini e i concetti dell’ontologia si ottiene il filo conduttore del dialogo. Il grado di vicinanza tra i termini ed i concetti esprime la vicinanza semantica tra gli elementi del dominio che si sta considerando. Dalla conversazione si ottiene un possibile percorso didattico tra i documenti che sono anch’essi mappati nello spazio semantico ed organizzati in cluster dal modulo della SOM. La mappa è organizzata a partire da una metafora spaziale. I documenti sono rappresentati come un insieme di costellazioni all’interno di un cielo stellato. I documenti semanticamente più vicini sono quindi più vicini tra loro ed anche le costellazioni vicine tra loro rappresentano gruppi di documenti che hanno un livello crescente di comunanza. Per gli scopi del presente lavoro si darà una più approfondita spiegazione delle metodologie che hanno portato alla creazione degli elementi del sistema nei prossimi paragrafi.

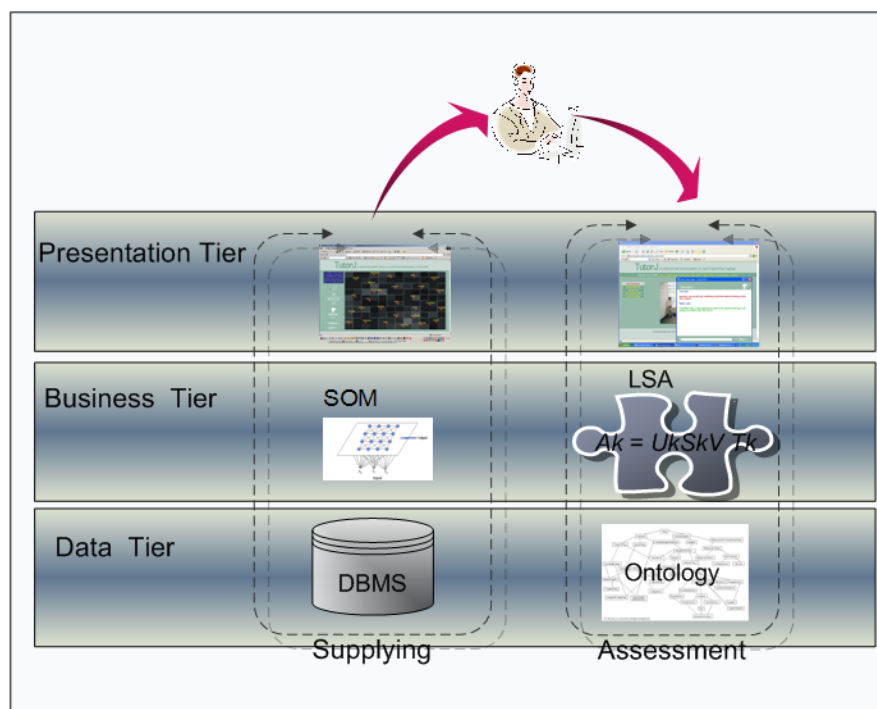


Figura 1: Rappresentazione del sistema

Il sistema è stato realizzato in java come una web application a più livelli in cui i vari livelli in senso orizzontale sono accomunati da una stessa funzione. Al livello più basso il livello dei dati in cui vengono conservati i dati, nel livello intermedio si definisce la logica con cui il sistema è stato realizzato mentre nel livello più alto si definisce la logica di presentazione e di interazione con gli utenti.

4 IL MODULO DI VALUTAZIONE

4.1 *Ontologia di dominio*

Tutte le informazioni e le strutture dei concetti del dominio per realizzare l’ontologia sono implementati all’interno della base di conoscenza di OpenCyc (Reed 2002). L’ontologia che definisce il linguaggio Java (concetti, relazioni, vincoli, regole e così via) sono state definite a partire dal documento ufficiale pubblicato dalla Sun Microsystem (Gosling et al 2003) che riporta le principali caratteristiche del

linguaggio. In un secondo tempo l'ontologia è stata validata da insegnanti di informatica con comprovata esperienza nell'insegnamento a livello universitario del linguaggio Java. In accordo a quanto esposto precedentemente le relazioni dell'ontologia sono state suddivise in due categorie fondamentali: le relazioni di struttura e quelle di navigazione. Le prime servono per esprimere i legami strutturali dipendenti dal dominio che si sta rappresentando. Ad esempio la preposizione (`#$iscomposed #Java #Statement`) è utilizzata per esprimere la relazione di composizione che unisce il concetto Java con una categoria di termini che tecnicamente si definiscono Statement. Tuttavia tali predicati da soli non sono sufficienti alla generazione di un percorso di apprendimento. È necessario quindi introdurre una nuova categoria di predicati. La seconda categoria di concetti definisce la spina dorsale del percorso di apprendimento ed è definita in relazione ai concetti di precedenza della KST. Le relazioni che definiscono tale categoria hanno il compito di definire i prerequisiti del dominio. In tale categoria rientrano due relazioni; la prima definisce i prerequisiti "forti" ed è stata definita ad hoc: la relazione in questione è la relazione "`#$isaPrerequisiteFor`" e gode della proprietà di transitività. La seconda relazione è presente all'interno della base di conoscenza di OpenCyc ed è utilizzata per la definizione dei prerequisiti "deboli". Tale relazione è la relazione "`#$conceptuallyRelated`" e assume un'importanza per definire dei percorsi alternativi all'interno della base di conoscenza. Le regole per la navigazione del dominio sono state inserite all'interno della base di conoscenza. Ad esempio la regola seguente è stata definita per definire i predecessori all'interno della base di conoscenza passando per livelli intermedi. In questo modo si tiene traccia degli elementi della base di conoscenza già visitati per non inserirli ulteriormente tra i prerequisiti trovati.

```
(#$implies
#$or (#$and
#$or (#$isa ?N ?Pr)
#$isaPrerequisiteFor ?Pr ?N )($gens ?Pr ?N) #$isaMethodOf ?N ?Pr)))
#$and (#$or (#$isa ?N ?Inter)($isaPrerequisiteFor
?Inter ?N) ($gens ?Inter ?N)($isaMethodOf ?N ?Inter)))
($Predecessore ?Pr ?Inter)) ($Predecessore ?Pr ?N ) )
```

4.2 LSA

Il modulo LSA è centrale nel processo di acquisizione delle domande dello studente. Tale modulo parte dalla scomposizione del corpo dei documenti con un iniziale preprocessing per eliminare le parole comuni (stop-words). La fase successiva è la creazione di una matrice cosiddetta delle co-occorrenze, $A(n \times n)$, in cui si definisce la cardinalità della presenza delle parole all'interno del documento. A questo punto per ridurre la complessità computazionale si effettua una trasformazione SVD (Single Value Decomposition) che approssima la matrice di partenza con una di ordine k ($k < n$) secondo la formula

$$A_k = U_k S_k V_k^T$$

In questo modo per ogni documento viene definita un'associazione che cattura la maggior parte del contenuto semantico ed elimina il rumore dovuto alla presenza nel corpo dei documenti di parole poco significative. Nel caso specifico l'insieme dei documenti di addestramento è rappresentato da documenti che descrivono i concetti presenti nell'ontologia. Questa scelta consente di mantenere un mapping dei documenti e dei termini dell'ontologia all'interno dello spazio semantico costruito dall'LSA. Anche la conversazione con lo studente viene immessa all'interno dello spazio semantico. In tal modo si crea un ambiente comune all'interno del quale i concetti e le definizioni date anche dagli studenti esterni sono valutati secondo la stessa metrica.

4.3 Il Chatbot

Il sistema utilizza per la conversazione una versione modificata del chatbot A.L.I.C.E. (vedi Fig. 2). La struttura della conversazione è modellata tramite l'AIML, un paradigma che gestisce il processo mediante il quale ad alcune categorie particolari di domande vengono assegnate specifiche risposte. Per far interagire il chatbot con la base di conoscenza si utilizza la tecnologia Cyn(Coursey 2004) in modo

da ottenere la definizione degli elementi in SubL (OpenCyc) che è il linguaggio d'interrogazione della base di conoscenza al cui interno è presente il modulo dell'ontologia.

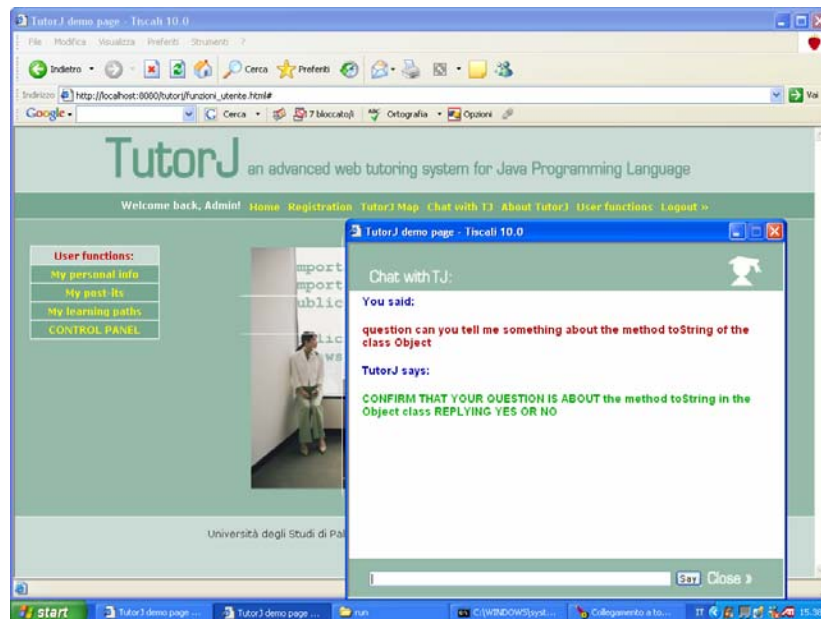


Figura 2. Il Chatbot

5 IL MODULO DI VISUALIZZAZIONE

5.1 La SOM

Le Self-Organizing Map, sviluppate dal prof. Teuvo Kohonen a partire dal 1979 sono orientate alla rappresentazione delle topologie associate a dati di ingresso, definiti in spazi con elevata dimensionalità, attraverso una mappa generalmente bidimensionale. Questo processo di organizzazione spaziale delle caratteristiche dei dati di ingresso è chiamato anche Feature Mapping e viene realizzato dalle SOM con una tecnica di apprendimento non supervisionato, da cui il nome che indica l'auto-organizzazione. Per quanto detto la SOM è anche nota in letteratura come Kohonen Feature Map. All'interno di una SOM viene definito il cosiddetto strato di Kohonen, costituito da una matrice di neuroni (Processing Elements) disposti nello spazio in maniera ordinata e collegati ad un insieme comune di ingressi. La posizione di ogni neurone sulla mappa è individuata dalle sue coordinate e la dimensionalità della matrice dello strato determina la dimensionalità della SOM.

5.2 La mappa

La mappa (vedi Fig. 3) visualizza i termini ed i documenti presenti nel dominio e agisce come "mappa concettuale", pur non fornendo alcun link di struttura tra i concetti. Questa GUI rappresenta ciascun documento come una stella ed una collezione di documenti come un cielo stellato. I documenti, rappresentati nello spazio semantico tramite la LSA e quindi divisi per classi dalla SOM. La SOM divide per classi i documenti, che sono rappresentati come vettori di uno spazio concettuale. In questo modo due documenti simili per il contesto semantico che trattano, vengono inclusi nella stessa classe. La SOM è stata utilizzata per ottenere i legami classici di tipo orizzontale tracciabili all'interno di una struttura ipertestuale, ma un'importante applicazione aggiuntiva è quella di ottenere automaticamente quei legami di associazione che mancano negli ipertesti: di solito i percorsi didattici sono predefiniti e quindi rigidi, ragione per cui chi naviga attraverso questa struttura ha poche possibilità di ottenere informazioni aggiuntive che non siano state previste al momento di costruzione del percorso di apprendimento. I legami di associazione invece permettono di esplorare in maniera differente la struttura considerata per cui la stessa informazione può essere utilizzata in contesti diversi in relazione al tipo di percorso che si sta effettuando. L'utilizzo della mappa nel sistema considerato consente la realizzazione di uno spazio concettuale continuo in cui una serie di documenti riguardanti

concetti differenti sono organizzati tra di loro. Lo spazio concettuale è costruito sopra la SOM tramite le proprietà che legano concettualmente i differenti modelli.

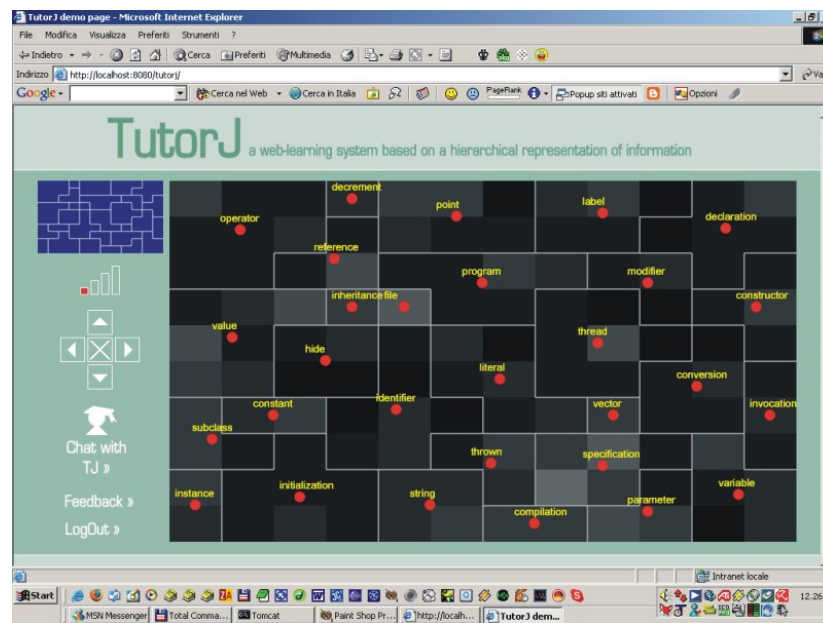


Figura 3. La mappa

Gli studenti possono navigare liberamente all'interno della mappa e visualizzare gli argomenti per loro più importanti oppure seguire il percorso di apprendimento definito all'interno del sistema tramite il modulo di conversazione. Il percorso viene visualizzato all'interno della mappa come una polilinea che congiunge gli elementi rappresentati dai termini di livello più alto e quindi più significativi.

6 LO SCHEMA DELLA CONVERSAZIONE

Lo scopo di questa procedura è l'ottenimento di un percorso di apprendimento che congiunga il goal item (ciò che è stato chiesto dallo studente) tramite segmenti che stanno sulla frontiera del knowledge state dello studente. Non è necessario conoscere l'intero knowledge space dello studente: basta continuare la conversazione fino al punto in cui lo studente risponde correttamente ad una domanda posta dal sistema. L'interazione parte dal punto in cui lo studente chiede al sistema uno specifico topic che vuole conoscere. Il chat-bot usa il modulo dell'LSA per comprendere l'oggetto della questione posta e per sicurezza chiede conferma. Se lo studente risponde negativamente il chat-bot chiede allo studente di riformulare la domanda utilizzando altri termini. Altrimenti la procedura ha inizio. Ogni argomento della conversazione è associato ad un nodo dell'ontologia. Il chatbot naviga l'ontologia per stabilire quali sono gli argomenti legati a quello chiesto al fine di arrivare al Knowledge State comprendente l'argomento richiesto. Un nodo A è considerato predecessore del nodo D se il concetto associato ad A è un prerequisito per la comprensione del concetto associato a D. Il modulo del chat-bot sceglie uno dei predecessori e si muove verso il corrispondente nodo dell'ontologia. Quindi allo studente viene posta una domanda circa il concetto associato e l'LSA analizza la risposta. Il comportamento del sistema è determinato dal livello di correttezza c della risposta. Tale livello è normalizzato tra zero (risposta completamente errata) ed uno (risposta completamente corretta). Nella tabella seguente viene riassunto il comportamento del sistema in base al grado di correttezza della domanda che viene sottoposto all'interno di quattro possibili intervalli.

Correttezza	Classificazione dell'argomento	Azione del chatbot
-------------	--------------------------------	--------------------

$0 \leq c < t_a$	Sconosciuto	Verifica dei predecessori
$t_a \leq c < t_b$	Incerto	Seconda domanda sull'argomento
$t_b \leq c < t_c$	Conosciuto	Argomenti concettualmente legati
$t_c \leq c \leq 1$	Perfettamente conosciuto	Nodo seguente

Tabella 1: tabella di corrispondenza

L'intera procedura è ripetuta tutte le volte necessarie al raggiungimento dell'intero percorso. Al termine della procedura il sistema determina un percorso coerente e lo mostra tramite l'interfaccia.

7 PRINCIPALI CONSIDERAZIONI SUL SISTEMA

Il sistema è stato sviluppato mettendo particolare attenzione ai seguenti aspetti:

1. **la gestione dell'interazione:** il sistema è stato pensato come in grado di gestire l'interazione con gli utenti in maniera innovativa. L'utilizzo dello strumento chatbot per interagire con gli studenti in cui si utilizza il linguaggio naturale rappresenta un aspetto innovativo e pone una nuova tipologia di problematiche sia di livello implementativo che anche di livello tecnologico non indifferenti.
2. **la gestione della conoscenza:** la realizzazione di un sistema che utilizza un'ontologia per gestire la conoscenza necessaria ha degli elementi di novità ed anche di possibile integrazione con altri strumenti che utilizzano gli stessi formalismi per la rappresentazione della conoscenza. Inoltre l'utilizzo di una base di conoscenza strutturata come OpenCyc permette di utilizzare conoscenze di base basate sul sensocomune che si possono integrare e rappresentare anche un aspetto implementativo avente un valore aggiunto di tipo strutturale e pedagogico.
3. **l'integrazione delle tecnologie:** l'utilizzo di tecnologie eterogenee e oggetto di ricerca scientifica in ognuno dei componenti del sistema dà una valenza prototipale elevata ed una importante base di partenza nella costruzione di sistemi che utilizzino una parte degli esperimenti realizzati e li integrino con sistemi più affermati per realizzare applicazioni scalabili su ampia scala.

Il sistema realizzato parte da considerazione pedagogiche sulla necessità di interessare lo studente nel percorso di apprendimento ed in tale ottica la scelta di un'interazione in linguaggio naturale, pur con tutte le problematiche di tipo implementative che comporta, è sembrata come essere l'elemento aggiuntivo che qualificava le scelte fatte in termini di cambiamento nelle modalità di fruizione e quindi di apprendimento. In definitiva la metodologia di costruzione del sistema rappresenta uno sforzo di applicazione ad un caso concreto, i sistemi per il tutoring di studenti, da affiancare alle tecniche tradizionali per la creazione di sistemi di interazione innovativi dal punto di vista tecnologico e che abbiano una elevata valenza di supporto nella gestione del processo di apprendimento a distanza. In tale ottica si sta iniziando a integrare il sistema all'interno di una piattaforma FAD di tipo tradizionale.

8 CONCLUSIONI

In questo lavoro si è introdotto una procedura che valuti la conoscenza di determinati argomenti all'interno di un ITS da parte di uno studente che lo utilizza come strumento di supporto per l'approfondimento di determinati aspetti di un dominio di conoscenza. Tale procedura è basata sulla KST. Il sistema è stato presentato nei suoi componenti fondamentali e si è definito anche l'interazione con lo studente ed il flusso

delle informazioni all'interno del sistema. Tale aspetto rappresenta senz'altro un aspetto innovativo anche nella gestione dell'innovazione che i componenti del sistema hanno anche dal punto di vista della ricerca scientifica. I sistemi di conversazione in linguaggio naturale, l'utilizzo di ontologie per la rappresentazione della conoscenza rappresentano punti di forza del sistema che vanno ad aumentare il valore innovativo del sistema presentato.

I prossimi passi di sviluppo sono incentrati come già accennato sull'integrazione del sistema all'interno di una piattaforma FAD tradizionale oltre che alla definizione di un modello dello studente utile per fissare alcuni elementi fondamentali che meglio potranno effettuare il processo di valutazione delle competenze dello studente. Il sistema realizzato è sicuramente in fase prototipale, ma da esso si sono presi alcuni spunti importanti per realizzare componenti innovative all'interno di applicazioni complesse.

9 BIBLIOGRAFIA

- H. WAINER, E.: Computerized adaptive testing: A primer. *Erlbaum, Hillsdale, NJ*, 1990
- JAMESON A. Numerical uncertainty management in user and student modeling: An overview of systems and issues. *User Modeling and User-Adapted Interaction* 5, 1996 pp.193–251
- FALMAGNE J., DOIGNON, J., KOPPEN, M., VILANO, M., JOHANNESSEN, L.: Introduction to knowledge spaces: How to build, test, and search them. *Psychological Review* 97, 1990 pp 201–224
- BUTZ, C., HUA, S., MAGUIRE, R.: A web-based intelligent tutoring system for computer programming. In: *IEEE/WIC/ACM International Conference on Web Intelligence (WI'04)*. 2004 pp 159–165
- CONATI, C., GERTNER, A., VANLHEN, K.: Using bayesian networks to manage uncertainty in student modeling. *User Modeling and User-Adapted Interaction* 12, 2002 pp 371–417
- COSYN, E., THIERY, N.: A practical procedure to build a knowledge structure. *Journal of Mathematical Psychology* 44 2000 pp 383–407
- DOWLING, C., HOCKEMEYER, C.: Automata for the assessment of knowledge. *IEEE Trans. on Knowledge and Data Engineering* 13 2001 pp 451–461
- PIRRONE, R., COSSENTINO, M., PILATO, G., RIZZO, R., RUSSO, G.: Discovering learning paths on a domain ontology using natural language interaction. In: *Proc. of the 18th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems IEA/AIE 2005, Bari, Italy* 2005 pp 236–240
- ALBERT, D., HOCKEMEYER, C.: Adaptive and Dynamic Hypertext Tutoring Systems Based on Knowledge Space Theory. *Artificial Intelligence in Education: Knowledge and Media in Learning Systems, Frontiers in Artificial Intelligence and Applications* 39 1997 553–555
- ALBERT, D., HOCKEMEYER, C.: Applying demand analysis of a set of test problems for developing adaptive courses. In: *Proc. of International Conference on Computers in Education*. 1. 2002 pp 69–70
- HARP, S., SAMAD, T., VILANO, M.: Modeling student knowledge with self-organizing feature maps. *IEEE Trans. on Systems, Man and Cybernetics* 25 1995 pp 727–737
- FALMAGNE, J., COSYN, E., DOIGNON, J., THIERY, N.: The assessment of knowledge, in theory and in practice. In: *International Conference on Integration of Knowledge Intensive Multi-Agent Systems*. 2003 pp 609–615
- COURSEY, K.: Living in CyN: Mating AIML and Cyc together with Program N (2004)
- LANDAUER, T., FOLTZ, P., LAHAM, D.: An introduction to latent semantic analysis. *Discours Processes* 25 1998 pp 259–284
- GUARINO N. Formal ontology, conceptual analysis and knowledge representation. *International journal of human-computer studies* 43 1995 pp. 625-640
- KOHONEN, T. The self-organizing map. *Proceedings Of The Ieee, Vol. 78, No 9, September* 1990
- OPENCYC. The OpenCyc SubL Application Programming Interface [en ligne]. Disponible sur : < <http://www.opencyc.org/doc/opencycapi>> (consulté le 18.02.2007)
- REED, S.AND LENAT, D. *Mapping Ontologies into Cyc*. In AAAI 2002 Conference Workshop on Ontologies For The Semantic Web, Edmonton, Canada, July 2002.
- ALICE. The A.L.I.C.E. bot [en ligne]. Disponible sur : < <http://www.alicebot.org>> (consulté le 10.03.2007)

JAMES GOSLING, BILL JOY, GUY STEELE, GILAD BRACHA: Java(TM) Language Specification, The (3rd Edition) *Prentice Hall PTR* 2005