

Caractérisation de Requêtes d'Assistance à partir de corpus

**François Bouchet
Jean-Paul Sansonnet**

LIMSI-CNRS

{bouchet, jps}@limsi.fr

MFI'07

31 mai 2007

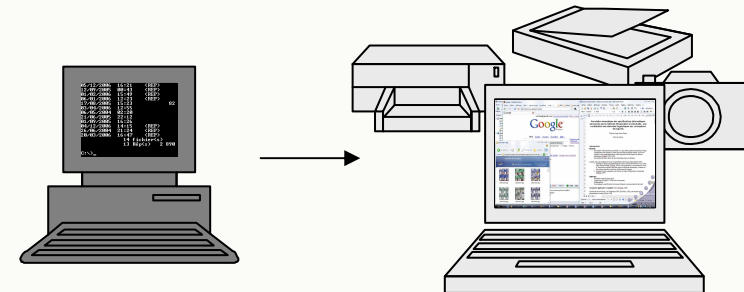
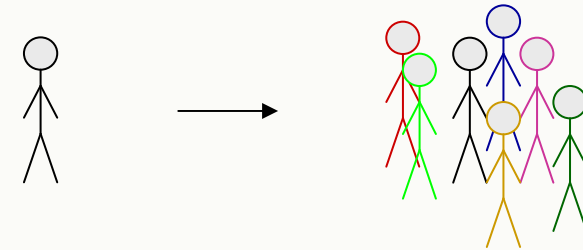


Plan d'ensemble

- Contexte
 - ◆ Le besoin d'assistance
 - ◆ Le projet Daft
- Le corpus Daft
 - ◆ Présentation générale
 - ◆ Catégorisation des activités conversationnelles
- Le langage DAFT
 - ◆ 3 niveaux : modalités, prédicats, références
 - ◆ Champs et types
 - ◆ Syntaxe générale et exemples
 - ◆ Application au corpus
- Conclusion et perspectives

Contexte : le besoin d'assistance

- Evolution des **utilisateurs** d'outils informatiques :
 - ◆ En **nombre** : la moitié des ménages français ont au moins un ordinateur [INSEE, 2005].
 - ◆ En **diversité** : informatique d'informaticiens → informatique pour tous.
- Evolution des **ordinateurs** (loi de Moore) et de :
 - ◆ Leur **champ d'application** : traitement de texte, multimédia, jeux vidéos, recherche d'infos...
 - ◆ Leur **champ d'interaction** : imprimante, scanner, PDA, appareil photo...
- Evolution des **logiciels** devenus :
 - ◆ Plus **nombreux** : dizaines de traitements de texte ou d'applications d'édition de photos.
 - ◆ Plus **complexes** : 45 boutons et 150 actions «de base » dans 9 menus (Word 2003).

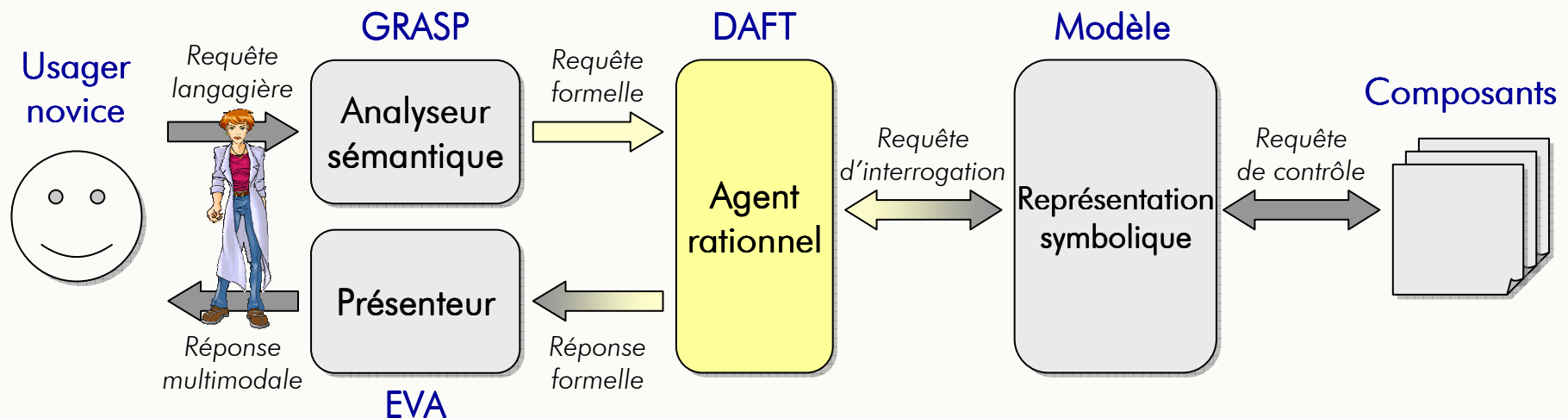


Inadéquation croissante entre le système et la représentation cognitive qu'en ont les utilisateurs.

Contexte : le projet Daft

- Objectifs :
 - ◆ Agents Conversationnels Animés dédiés à la fonction d'assistance :
 - Pour : des utilisateurs novices s'exprimant en langue naturelle non contrainte
 - Par : un raisonnement sur la structure et le fonctionnement des applications (aide contextuelle)

- Schéma d'architecture générale :



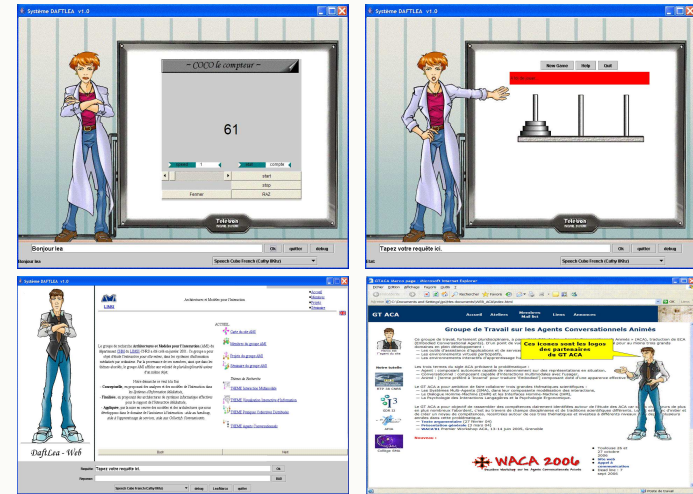
Corpus Daft : présentation

- 11 000 requêtes, 3 origines :

- Recueil de requêtes de vrais utilisateurs, dans 4 environnements différents : deux applications Java simples, un site web actif et un site « ouvert au public ».

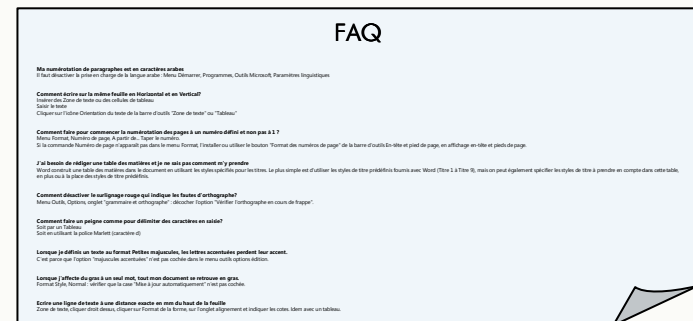
- Structures linguistiques issues de **thésaurus**, réemployées dans le contexte des environnements précédents ;

- Questions issues de **FAQ** de logiciels de composition de documents (LaTeX et Word).



Pourriez-vous... ?
Vous serait-il possible de... ?
Est-ce que vous pourriez... ?

...

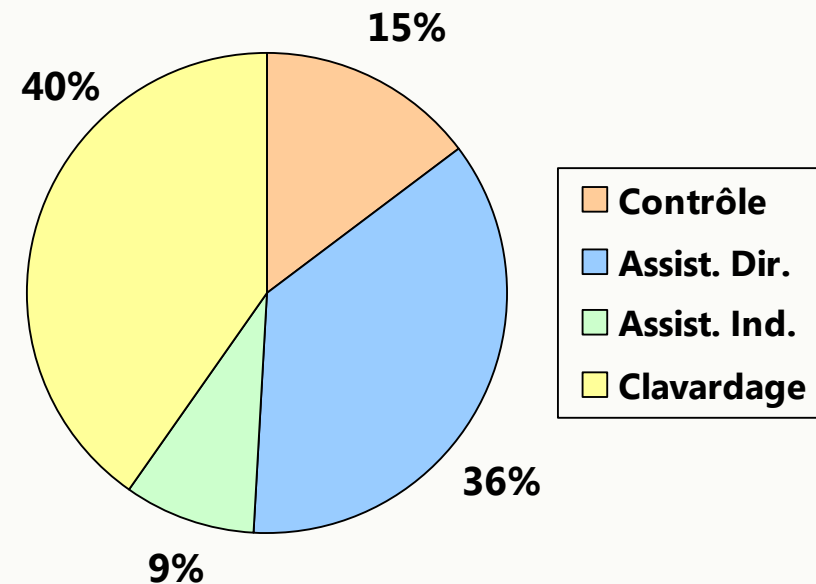


Corpus Daft : catégorisation

- Méthodologie :
Annotation manuelle des requêtes issues de deux sous-ensembles au 1/10^e du corpus.

- Résultats :

- ◆ **Contrôle**
Commandes prédicatives via l'agent
- ◆ **Assistance directe**
Demandes d'aide explicites
- ◆ **Assistance indirecte**
Commentaires sous-entendant une demande d'aide (pragmatique)
- ◆ **Clavardage**
Interactions utilisateur-agent



- Conclusion :
4 activités distinctes
→ 4 sous-corpus

[Bouchet et Sansonnet, 2006]

Langage DAFT : 3 niveaux imbriquables

- On cherche à déterminer un langage permettant de formaliser des requêtes de type : contrôle et assistance (directe et indirecte).
- On se propose de distinguer 3 niveaux imbriquables : $R_i \subset P_i \subset M_i$
 - ◆ 39 **Modalités** (M_i) \approx actes de dialogue
 - Informations structurelles d'une entité : **VALUE, MEANING, ROLE...**
 - Identification d'entités : **EXISTENCE, OTHER...**
 - Relation inter-entités : **DIFFERENCE, ORDER...**
 - Capacités, droits, devoirs : **POSSIBILITY, KNOWLEDGE, OBLIGATION...**
 - Circonstanciennes : **WAY, EFFECT, PLACE...**
 - Sentiments : **LIKE, FEAR, DOUBT...**
 - Assistance : **PROBLEM, HELP...**
 - Marqueurs : **NEG, ASK, CHECK.**
 - ◆ 46 **Prédicats** (P_i) = verbes d'action ou de prédication
 - **Modify, Cancel, Click, Go, Quit...**
 - ◆ ∞ **Références** (R_i) \approx GN – Réf. Extensionnelles Associatives
 - « le petit bouton », « le champ texte du bas »...

Langage DAFT : champs et types

- Modalités et prédicats sont dotés de champs (ch_i) : attribut = valeur

valeur $\in \{M_i, P_i, R_i, \text{marqueur}\}$ ($R_i \subset P_i \subset M_i$)
marqueur = $\{T, \neg T, \text{PAST}, \dots\}$ (lié au type)

- 12 types de champs : [nom_du_type]

Types = {
[act], action N'importe quel prédicat
[obj], objet « un bouton », « une page », ...
[ppt], propriété « couleur », « récent », ...
[per], personne {u, s}
[man], manière « un peu », « à l'envers », ...
[pla], lieu « le site », « la fenêtre », ...
[tim], temps {PAST, PRES, ALWAYS...}
[boo], booléen {T, $\neg T$ }
...}

Langage DAFT : syntaxe générale et exemples

- Exemples de schémas :
 - ◆ Identification : **EXISTENCE**([obj]of, [tim]time, [pla]in)
 - ◆ CDD : **KNOWLEDGE**([per]of, [act]todo, [obj]about, [int]certainty)
 - ◆ Circonstancielle : **WAY**([act]goal, [int]qual) (qual={0:none, 1:best, 2:another})
 - ◆ Prédicat : **Sort**([obj]object, [man]manner)
- Syntaxe générale d'une requête formelle DAFT :
 $M_1(\dots M_n([t_1]c_1 = P_1([t_{11}]c'_1 = R_1, \dots, [t_{1l}]c'_l = R_l), \dots, [t_m]c_m = P_m(\dots)) \dots)$
- Exemples de requêtes :
 - ◆ Trie le tableau $\rho = \text{Sort}(\text{object}=\text{« le tableau »})$
 - ◆ Sais-tu trier le tableau ? $\text{ASK}(\text{KNOWLEDGE}(\text{of}=\text{s}, \text{todo}=\rho))$
 - ◆ Existe-t-il un moyen de trier le tableau ? $\text{ASK}(\text{EXISTENCE}(\text{of}=\text{WAY}(\text{goal}=\rho)))$
 - ◆ Connais-tu un moyen de trier le tableau ?
 $\text{ASK}(\text{KNOWLEDGE}(\text{of}=\text{s}, \text{about}=\text{WAY}(\text{goal}=\rho)))$
 - ◆ Existe-t-il un moyen de savoir comment trier le tableau ?
 $\text{ASK}(\text{EXISTENCE}(\text{of}=\text{WAY}(\text{goal}=\text{KNOWLEDGE}(\text{of}=\text{WAY}(\text{todo}=\rho))))))$

Langage DAFT : des modalités diverses

- Les modalités, regroupées en 8 catégories, se distinguent par :
 - ◆ La **complexité propre** de leurs schémas :
 - **ASK**() → 1 champ unique
 - **KNOWLEDGE**([per]of, [act]todo, [obj]about, [int]certainty) → 4 champs (non simultanés)
 - ◆ Leur **fréquence** : $f(\text{FEAR}) < f(\text{POSSIBILITY}) < f(\text{ASK})$ (loi de Zipf)
 - ◆ Leur **généricité** : des opérateurs communs (**KNOWLEDGE**) côtoient des opérateurs fortement lié au domaine d'étude (**HELP**).
 - ◆ Leur **position** dans la requête : les marqueurs sont plutôt à l'extérieur, les modalités d'informations à l'intérieur. Mais il n'y a pas moyen de redécomposer les modalités (cf. exemples précédents).
 - ◆ La **complexité d'identification** des phénomènes sémantiques représentés : par exemple, besoin d'élicitation pour **PLACE** et **MOMENT**.
 - « je ne trouve pas X » → **PLACE** (« l'endroit où est X »)
 - « stoppe le compteur à 100 » → **MOMENT** (« quand il vaut 100 »)

Langage DAFT : application au corpus

- Une complexité croissante des requêtes formelles :

Contrôle < Assist. dir. < Assist. indir.

- Une couverture inégale (1075 phrases retranscrites manuellement).

Sous-corpus	Ctrl	Assist. dir.	Assist. indir.
Couverture	92,3%	96,7%	70,2%

- Mais une syntaxe robuste qui résiste au changement d'échelle :

Une augmentation de la taille du corpus (+50%) fait augmenter le nombre de prédicats nécessaires de 5% (4 nouveaux).

Glisser le disque de droite à gauche	<code>Move(object="le disque",origin="droite",destination="gauche")</code>
d'après toi, y a t-il des fonctions d'annulation dans cette application?	<code>ASK(KNOWLEDGE(of=s, about=EXISTENCE(time=PRES, of=FUNCTION(doing=Cancel(), in="cette appli..."))))</code>
j'ai peur qu'il n'y ait pas moyen de changer la taille de la police qui est bien trop petite	<code>FEAR(agent=u, fear=NEG(POSSIBILITY(todo=Modify(object="la police...trop petite",property="taille"))))</code>

Conclusion et perspectives

- On a défini un langage formel :
 - ◆ robuste et adapté au contexte d'étude (l'assistance), grâce à l'approche basée sur un corpus ;
 - ◆ décomposable en 3 niveaux :
 - 39 modalités très diverses (en complexité, fréquence...),
 - 46 **prédicats** représentatifs des actions génériques les plus fréquentes,
 - des **références**.
- Perspectives :
 - ◆ Automatiser la transcription (en associant des schémas partiels à des clés sémantiques – ex : « éditable » → **POSSIBILITY**(todo=**Modify**()))
 - ◆ Utiliser le langage comme base de raisonnement pour l'agent rationnel pour produire des réactions correctes et pertinentes.