

Manipulations d'« Eliza¹ »

Il existe à ce jour de nombreuses versions de ce fameux programme de pseudo-psychothérapie. On va travailler avec celle qui est programmée en Lisp et accompagne le programme de traitement de texte Emacs.

Sous Linux, lancer emacs, soit par le menu, soit dans un shell en tapant `emacs &`.

Sous Emacs, lancer le programme en tapant `esc x doctor` <Entrée>.

Une fois le programme lancé, il faut taper vos phrases suivies de **deux** fois la touche <Entrée>.

1. Exploration du programme

Une fois le programme lancé, interagissez avec notre pseudo-psychothérapeute en tapant des phrases — en anglais.

1. Après quelques échanges, identifiez les stratégies adoptées quand Eliza ne trouve rien à exploiter dans le texte qui précède, listez les différentes réponses possible d'Eliza.
2. Quel est l'effet (en fonction de leur position) des particules *Wh-* ?
3. Identifiez les règles associées aux mots-clés '*always*' et '*when*'. Trouvez des exemples qui font échouer Eliza. Comment améliorer ces règles ?
4. Le programme définit quelques thèmes classiques (plus ou moins freudiens...) dont il repère l'apparition par des mots-clés, et qui déclenchent un traitement particulier. Repérez quelques-uns de ces thèmes.
5. Parlez à Eliza de vos problèmes de santé et notez les comportements observés.
6. Notez le comportement d'Eliza relatif à la longueur/brièveté des phrases.
7. Cas extrême de fonctionnement par mot-clé : dites bonjour et au revoir plusieurs fois à Eliza.
8. Mentionner Stallman² et observer le tour que prend le dialogue.
9. Observer la réaction d'Eliza quand on lui pose une question.
10. Quand est-ce qu'Eliza répond : « *Don't tell me what to do. I am the psychiatrist here !* »

2. Mise en évidence des règles

Une fois votre exploration terminée, tentez d'établir des règles simples qui pourraient donner les effets observés.

Exemple : soit le tour de dialogue :

- *I want to talk to you*
- *Why do you say that you want to talk to me ?*

Voici une règle possible : soit ϕ la phrase du patient. Si ϕ commence par *I*, produire la phrase *Why do you say that* $\phi_{[you/I][I/you]}$, où la notation $\alpha_{[a/b]}$ désigne la phrase α dans laquelle toutes les occurrences de a sont remplacées par b .

Trouvez 3 règles différentes de ce genre de complexité, et vérifiez leur fonctionnement dans Eliza.

3. Vérification : le source en lisp

Suivant les indications de l'enseignants, regarder le code source du programme 'doctor' qui nous venons d'explorer.

Observez l'importance du lexique, la présence massive de phrases toutes faites (*canned text*), ou de phrases à trous sommaires.

¹Invention de Joseph Weizenbaum, MIT, 1964-66.

²Richard Stallman est le concepteur d'Emacs, célèbre pour être à l'origine du projet GNU, visant à créer des logiciels publics et gratuits.
