Les applications de la théorie des automates au traitement des langues naturelles

Dominique Perrin

- Domaines d'application (en général et pour les langues naturelles).
- Fonctions rationnelles
- Ensembles localement testables
- Ensembles sans étoile

Une proposition et trois questions.

Applications des automates

- analyse lexicale
- traitement de textes (Unix)
- compression
- conception de circuits
- systèmes de communication (protocoles)
- vérification
- ...

Domaines d'application pour les langues naturelles

- grammaires locales (M. Gross)
- filtrage (INTEX et Unitex)
- étiquetage (H. Johnson, Y. Schabes)
- phonétisation (E. Laporte, M. Mohri)
- compression de texte
- correction orthographique

références:

Maurice Gross, Dominique Perrin, *Electronic Dictionnaries and Automata in Computational Linguistics*, Springer, 1989.
Emmanuel Roche, Yves Schabes, *Finite-state Language Processing*, MIT Press, 1997.

Relations et fonctions rationnelles

Une relation $\rho \subset A^* \times B^*$ est rationnelle si elle peut être calculée par un automate fini.

C'est une fonction rationnelle quand...

Question: peut-on construire un logiciel implémentant de façon générale les relations et les fonctions rationnelles pour les applications à la langue naturelle?

Fonctionalités (outre les accès aux dictionnaires et aux lexique-grammaire):

- composition
- affectation (redirection)
- itération

Ensembles localement testables

Automate local: l'état d'arrivée ne dépend du passé que de façon bornée.

Pour un mot w, $F_k(w)$ est l'ensemble des facteurs de longueur k de w.

Une partie X de A^* est *localement testable* s'il existe un entier k tel que pour deux mots x, y tels que $F_k(x) = F_k(y)$ on a $x \in X$ ssi $y \in X$.

Exemple:

$$(ab)^* \quad \text{(oui)}$$

$$(ab + ba)^* \quad \text{(non)}$$

Raison: $y = (ab)^n b(ab)^n$ a le même contenu que y^2. Pourtant, $y \in X$ mais $y^2 \notin X$.

Question: tous les ensembles considérés en traitement de langue naturelle sont ils localement testables?

Ensembles sans étoile

Une partie X de A^* est sans étoile si elle peut être obtenue à partir des parties de A par les opérations de

- union, intersection et complémentation.
- concaténation.

En particulier, tous les ensembles localement testables sont sans étoile.
Proposition: tous les ensembles considérés en traitement de langue naturelle sont apériodiques.

Automate apériodique: il existe un entier k tel que pour tout état p et tout mot x,

$$p \cdot x^k = p \cdot x^{k+1}.$$

Logique temporelle

Langage logique construit avec:

- variables, quantificateurs et connecteurs.
- Prédicats $a(i)$ (la lettre en position i est un a)
- $i < j$ (la position i est avant j).

Exemple:

$$a(0) \wedge \forall i(a(i) \rightarrow b(i+1))$$

Question: peut-on construire un logiciel fondé sur ce mode d'expression?