ARISTOTO COMPREND-IL LE LANGAGE NATUREL ?

François LO JACOMO

Une petite phrase qui pose en réalité trois questions :
qu'est-ce que le langage naturel ? qu'est-ce que comprendre ? et... qu'est-ce qu'Aristoto ?

QU'EST-CE QUE LE LANGAGE NATUREL ?

Lorsque l'homme veut se faire comprendre de la machine, il emploie, encore aujourd'hui, un langage codé. Qu'il s'agisse d'un langage de programmation, d'instructions spécifiques à une application, de choix dans un menu ou de toute autre commande, les possibilités de compréhension de la machine semblent souvent limitées, ses réactions moyennement prévisibles, et le non-expert aura parfois du mal à obtenir de la machine ce qu'il souhaite.

Le langage naturel a pour but de faciliter ce dialogue homme-machine et de permettre à des personnes moins instruites de manipuler l'ordinateur, améliorant ainsi la convivialité des systèmes informatiques. Si l'on met à part les projets ambitieux de traduction automatique ou de traitement de la parole, les applications actuellement opérationnelles sur ce marché consistent généralement à traduire dans un langage compréhensible par la machine (langage d'interrogation de bases de données par exemple) une requête ou une information exprimée "librement" (dans notre langue de tous les jours, avec le moins possible de contraintes au niveau syntaxique et lexical).

QUE FAIT ARISTOTO ?

Aristoto est une application MacIntosh d'apprentissage et de compréhension du langage naturel, la première d'une (longue ?) série d'applications visant petit à petit une meilleure compréhension du langage naturel, à des fins, par exemple, didactiques (E.A.O.). Dans l'immédiat, Aristoto est essentiellement démonstratif de ce que l'on peut

faire, avec des moyens modestes, dans ce domaine; il doit être conçu comme un outil de formation s'adressant à tous ceux qui souhaitent approcher d'un peu plus près cette problématique.

Concrètement, Aristoto se compose de deux modules bien distincts : l'un calcule la conclusion d'un syllogisme en langage naturel, l'autre apprend à reconnaître si un prénom français est masculin ou féminin.

SYLLOGISMES

Le premier module est capable, à partir des deux prémisses : "tous les hommes sont mortels", or "Socrate est un homme", saisies telles quelles (en langage naturel, et non pas sous forme codée), de calculer la conclusion : "donc socrate est mortel". Il en est capable même dans des cas beaucoup plus compliqués : des deux prémisses "il y a des femmes qui sont idiotes", or "les professeurs ne sont pas idiots", il conclura : "donc certain(e)s femmes ne sont pas des professeurs"; ou encore, si on lui dit : "tous les hommes ne sont pas malheureux", or "aucun homme malheureux ne sait profiter de la vie", il répondra : "... cela ne prouve pas que certain(e)s hommes savent profiter de la vie".

Est-ce suffisant pour dire qu'Aristoto comprend le langage naturel ? Oui et non. Oui car il est capable d'extraire de deux phrases en langage naturel suffisamment d'informations pour en construire une troisième qui, dans un grand nombre de cas, est identique à celle qu'aurait construite un humain comprenant le langage naturel. Non, car il ne connaît même pas cinquante mots, ce qui est dérisoire par rapport aux dictionnaires des autres applications de compréhension du langage naturel, qui peuvent contenir des dizaines, voire des centaines de milliers de mots !

L'ARBORESCENCE DE CONTEXTES

C'est grâce à la notion de "contexte" qu'Aristoto peut fonctionner avec un lexique si réduit. Un contexte est une variable constituée de trois composantes (trois lettres), qui évolue au cours de l'analyse de la phrase (chaque mot, qu'il soit ou non dans le lexique, est susceptible de modifier le contexte). Par exemple, l'analyse de la phrase "il y a des grands nez crochus qui ne sont pas bien beaux" peut être symbolisée ainsi : (pbb) + "il" -> (pcb) + "y" -> (qdb) + "a" -> (qxb) + "des" -> (rxn) + "grands" -> (rxm) + "nez" -> (rxl) + "crochus" -> (rxl) + "qui" -> (sik) + "ne" -> (szk) + "sont" -> (tzh) + "pas" -> (toh) + "bien" -> (toj) + "beaux" -> (tok).

Un mot n'est pas analysable dans n'importe quel contexte : "il" n'est reconnu qu'en début de phrase, donc en contexte (pbb), et "y" n'est analysé qu'en contexte (pcb); "crochus" n'est pas analysé comme un pluriel, car en contexte (rxl) le système n'attend pas de pluriel (la gestion des pluriels n'étant peut-être pas optimale). En contexte (qdb), tout mot autre que "a" déclenche une erreur, et le "a" qui apparaît dans ce contexte n'est pas le même que le verbe avoir de "il a un grand nez". Si "qui" n'apparaissait pas, ou si la négation "ne" n'était pas complétée par "pas", une erreur serait déclenchée, car le contexte de fin de phrase ne serait pas valide.

VISUALISER L'ANALYSE

Aristoto étant essentiellement didactique, vous pouvez visualiser cette évolution des contextes, tout comme vous pouvez voir comment Aristoto découpe chaque prémisse en un quantificateur, un premier terme, un groupe verbal, un second terme, et identifie le moyen terme, celui qui apparaît dans les deux prémisses.

Vous pouvez même modifier les procédures de fonétization, troncature et d'orthographe, qui permettent d'identifier deux termes qui ne sont pas totalement identiques. En particulier, la fonétization permet d'identifier le pluriel et le singulier de presque tous les mots français (même "oeil" - "yeux" : des prémisses "aucun oeil n'est aussi beau que ton oeil gauche", or "tes yeux sont bleus", il conclura "donc parmi ceux qui sont bleus, il y en a un(e) qui n'est pas aussi beau que ton oeil gauche"), compensant ainsi, d'une certaine manière, les insuffisances de la gestion des pluriels.

Vous pouvez rajouter des mots au lexique : a priori, "très" est le pluriel de "trè", "faux" le pluriel de "fal", à moins que vous ne rajoutiez ces mots dans le lexique comme mots invariables. Vous pouvez rajouter des verbes en précisant comment ils se conjuguent, et même des quantificateurs ou d'autres mots, en précisant leur contexte de départ et leur contexte d'arrivée.

LES AMBIGUÏTÉS

Aristoto gère certaines ambiguïtés : étant donné le petit nombre de verbes connus au lexique, tout mot qui se termine par "-ent" est analysé soit comme un verbe, soit comme un non-verbe. "Les poules du couvent

couvent souvent leurs oeufs" génère quatre hypothèses : soit le premier "couvent" est un verbe, soit le second "couvent" est un verbe, soit "souvent" est le pluriel du verbe "souver", soit... la quatrième hypothèse est détruite en fin de phrase, car elle ne contiendrait pas de verbe.

Si la seconde prémisse est "tata est une poule", Aristoto conclura "donc tata couve couvent souvent leurs oeufs", choisissant le première hypothèse (en demandant s'il a eu raison d'identifier "poules du" avec "poule"). Mais si la seconde prémisse est "tata est une poule du couvent", il choisira la seconde hypothèse et conclura : "donc tata couve souvent leurs oeufs". Dans tous les cas, "leurs", non analysé, sera conservé tel quel. Lorsque vous visualiserez l'analyse mot à mot d'une prémisse ambiguë, Aristoto ouvrira, en chaque position de la phrase, autant de fenêtres qu'il y a d'hypothèses valides.

LE SEXE DES PRÉNOMS

Ce même souci d'expliciter à l'utilisateur tous les paramètres de l'analyse, à des fins didactiques, se trouve dans le module de prénoms, par lequel l'utilisateur enseigne à la machine à reconnaître si un prénom français est masculin ou féminin. A la première leçon, quel que soit le prénom fourni, l'ordinateur répondra qu'il a 50% de probabilité d'être masculin, car il ne connaît aucun prénom a priori, ni aucune règle permettant de déterminer le sexe d'un prénom. Mais il aura repéré dans le prénom certains indices, et l'information que devra lui donner l'utilisateur ("le prénom est-il masculin ou féminin ?") sera distribuée sur ces différents indices, de sorte que dès la seconde leçon, un prénom différent du premier mais contenant certains indices en commun avec le premier aura une probabilité différente de 50% d'être masculin ou féminin.

En pratique, assez vite, l'ordinateur ne fait presque plus d'erreurs, à condition que les prénoms antérieurement enseignés soient assez représentatifs. Si on lui enseigne le contraire de la réalité (par exemple, que "Sylvie" est masculin, "Robert" féminin, ...), il apprendra ce qu'on lui enseigne. Comme il recalcule à chaque fois la probabilité que le prénom soit masculin ou féminin, il se peut qu'un prénom qu'il a déjà rencontré comme masculin ait une plus forte probabilité d'être féminin ("Amédée", ...), ou inversement, et il signalera l'anomalie.

L'utilisateur peut à tout moment visualiser les indices qui ont été repérés dans n'importe lequel des prénoms déjà enseignés et la valeur

des paramètres qui leurs sont affectés, à partir desquels Aristoto a calculé la probabilité que ce prénom soit masculin; il peut modifier à tout moment la liste des prénoms, ce qui force Aristoto à recalculer instantanément tous les paramètres à partir de la modification.

COMMENTAIRES ET CONVIVIALITÉ

Contrairement aux applications classiques, le module de prénoms d'Aristoto commente ses propres réponses : les commentaires sont déterminés par les erreurs qu'il a commises (en donnant une forte probabilité d'être masculin à un prénom féminin ou inversement), et non par ce que fait l'utilisateur.

D'une manière plus générale, Aristoto propose un autre type de dialogue homme-machine, une nouvelle convivialité, où l'ordinateur est un interlocuteur plus qu'un outil, un peu comme le fameux programme Eliza. Ceci se remarque non seulement dans le choix des commentaires, mais aussi dans le fait que lorsque l'utilisateur commence une prémisse d'un syllogisme par "je", ce "je" deviendras "tu" dans la conclusion, et inversement : "aucune machine n'est aussi stupide que toi", or "tu es une machine très stupide", donnera : "donc je ne suis pas aussi stupide que toi"., ou même : "je suis moi", or "tu n'es pas moi", donnera : "donc je ne suis pas toi".

UN PANORAMA DES PROBLÈMES

Pourquoi réunir tant d'aspects variés dans un si petit programme ? Pour donner une idée de la complexité du langage naturel. Depuis les problèmes de filtrage (Aristoto ignore les ponctuations ou les espaces superflus, les accents, les majuscules...), d'apostrophes (que ce soit pour déceler des erreurs dans "tu ne es pas idiot" ou "les hommes n'pensent pas cela" ou pour conclure, de "certains hommes ne savent pas peindre", or "Arnaud ne sait pas peindre" : "... cela ne prouve pas qu'arnaud est un(e) homme") jusqu'aux problèmes de raisonnement, de gestion des erreurs (Aristoto explicite chaque erreur et sélectionne le mot sur lequel porte l'erreur), de convivialité du dialogue homme-machine, etc... la compréhension du langage naturel est un problème à multiples facettes : sans les traiter exhaustivement, Aristoto s'efforce d'en aborder le maximum, afin de faire réfléchir sur leur pertinence pour des applications plus concrètes, et sur la possibilité, notamment, de faire converger les techniques d'apprentissage du module des prénoms avec les techniques de gestion des contextes du module de syllogismes.

ORIENTATIONS STRATÉGIQUES

Aristoto s'inscrit dans un processus à long terme visant à démocratiser le langage naturel. Aujourd'hui, "langage naturel" signifie application coûteuse et très éloignée des préoccupations normales de l'utilisateur d'informatique. Il est fondamental de faire évoluer cette conception et de faire comprendre à chacun que même à un niveau modeste on peut tirer profit du langage naturel. Aristoto doit amorcer un dialogue entre utilisateurs et concepteurs visant à faire converger l'offre et la demande, car le marché du langage naturel, à l'heure actuelle, est essentiellement défini par l'offre.

Pourquoi vouloir faire de l'ordinateur un interlocuteur de l'homme, et non un outil ? Car j'ai la conviction, depuis plusieurs années déjà, que c'est une condition incontournable de la compréhension du langage naturel. C'est l'étude de la réalité linguistique qui dicte certaines contraintes inéluctables, qu'il faut prendre en compte même en informatique.

LE SENS DU MESSAGE

La première de ces contraintes, mise en évidence, notamment, par Danica Seleskovitch [1] au cours de ses recherches sur la théorie de la traduction, est que le sens d'un message n'est pas dans le message : les unités linguistiques contenues dans le message permettent d'en reconstruire le sens, mais elles ne contiennent pas ce sens. La preuve en est que plus un traducteur tente de se rapprocher des mots du texte à traduire, plus il s'éloigne du sens. En d'autres termes, les unités de sens ne sont pas des unités linguistiques, et c'est cette notion d'unité de sens qu'il convient de cerner et d'implémenter dans les prochains projets de compréhension automatique du langage naturel. Ce ne sera jamais dans des dictionnaires, aussi précis soient-ils, que l'on trouvera le sens d'un texte, et la distinction que font certains linguistes entre "sens" et "signification" ne résout pas totalement ce problème.

Une seconde contrainte, que l'on trouve par exemple dans les ouvrages de base d'André Martinet [2], est qu'une langue évolue parce qu'elle fonctionne. Il est donc nécessaire d'unifier les processus d'évolution et de compréhension du système linguistique, un peu comme

1 Danica Seleskovitch et Marianne Lederer, *Interpréter pour traduire*, Paris (Didier), 1984.
2 André Martinet, *Eléments de Linguistique Générale*, Paris, 1960 (réédité).

le module des prénoms d'Aristoto fait évoluer ses paramètres au fur et à mesure de ses expériences, mais en faisant appel à des techniques connexionnistes plus sophistiquées : c'est là qu'apparaît la notion de fluidité du système linguistique.

Une troisième contrainte, qui porte sans doute en elle la clé du problème, est ce que j'appelle la tridimensionnalité du système linguistique, et que l'on retrouve sous diverses formes chez de nombreux linguistes. Je pense par exemple à la théorie trois points de vue de Claude Hagège [3]. A côté de la dimension syntaxique (qui définit les rapports entre une unité linguistique et les autres unités présentes dans le message) et de la dimension sémantique (qui définit les rapports entre une unité linguistique et celles, absentes du message, qui auraient pu se trouver à sa place), la dimension pragmatique définit les rapports entre une unité linguistique et le contexte cognitif dans lequel elle apparaît. C'est à ce niveau-là que doivent être gérées les unités de sens et leurs rapports avec les unités linguistiques.

Tenir compte de ces trois contraintes pour modéliser un système de compréhension du langage naturel doit faire l'objet d'une prochaine étape de ma recherche : le projet Neurolang. Celui-ci sera appliqué à la compréhension de phrases simples (analyse de réponses) dans le cadre d'un système d'enseignement assisté par ordinateur, mais son lexique sera déjà plus riche que celui d'Aristoto (600 à 800 mots) et ses applications plus immédiates.

Les premières réalisations industrielles qui jalonneront cette démarche à long terme devront mettre en oeuvre des fonctionnalités voisines de celles d'Aristoto ou de Neurolang pour résoudre certains problèmes ponctuels de dialogue homme-machine, en réponse à une demande naissante. Mais parallèlement, l'activité de recherche doit se poursuivre vers la conception d'outils répondant progressivement à la question : que signifie véritablement "comprendre le langage naturel" ?

François LO JACOMO
Ingénieur Linguiste
Paris

3 Claude Hagège, *L'homme de paroles*, Paris (Fayard), 1985.