Extraction stochastique d'arbres d'analyse pour le modèle DOP

Jean-Cédric Chappelier et Martin Rajman

EPFL – DI-LIA, Écublens, CH-1015 Lausanne, Suisse
{chaps, rajman}@lia.di.epfl.ch

Résumé

Dans le cadre des approches à base de grammaires faiblement sensibles au contexte[1], cette contribution passe en revue le problème de l'extraction de l'arbre d'analyse le plus probable dans le modèle du Data-Oriented Parsing (DOP) (Bod, 1995). Une démonstration formelle de l'utilisabilité des méthodes Monte-Carlo est donnée, puis une technique d'échantillonnage contrôlée est développée permettant de garantir que l'arbre d'analyse sélectionné est bien (avec un certain seuil de confiance fixé *a priori*) l'arbre d'analyse le plus probable au sens du modèle DOP.

1. Introduction

L'analyse syntaxique guidée par les données (Data-Oriented Parsing ou DOP) (Bod, 1995) constitue aujourd'hui une des voix de recherche prometteuses dans le cadre des approches à base de grammaires faiblement sensibles au contexte[1]. Elle correspond à une mise en œuvre spécifique des grammaires probabilistes à substitution d'arbres et présente, à ce titre, plusieurs différences importantes avec le modèle usuel des grammaires stochastiques (SCFG[2]). En particulier, et à la différence des SCFG, il n'y a pas dans le cadre du modèle DOP une correspondance bi-univoque entre un arbre d'analyse et la façon de produire cet arbre par la grammaire (par exemple une liste de sous-arbres)[3].

La conséquence fondamentale de cet état de fait est qu'à la différence des SCFG, trouver l'arbre d'analyse le plus probable devient un problème NP-difficile (Sima'an, 1996). Seule la *dérivation* la plus probable peut être trouvée en un temps polynômial. Une solution possible consiste alors à chercher l'arbre d'analyse le plus probable, non plus par une méthode exacte mais par échantillonnage statistique au sein de la forêt des dérivations.

Cependant, pour qu'une telle technique soit effectivement opérationnelle, il est nécessaire que les probabilités servant à échantillonner la forêt des dérivations soient (1) elles-mêmes

1. « mildly context-sensitive grammars »
2. pour « Stochastic Context-Free Grammars »
3. et ceci malgré la convention consistant à réécrire systématiquement en premier la feuille non-terminale la plus à gauche.

calculables en un temps polynômial et (2) compatibles avec les probabilités des arbres d'analyse au sens du modèle DOP, de façon à ce que la probabilité d'obtenir un arbre d'analyse par échantillonnage soit égale à sa DOP-probabilité.

Le but de cet article est, d'une part, de fournir une démonstration formelle de la faisabilité des deux points ci-dessus (sections 2 et 3 ci-après) et, d'autre part, de proposer une méthode d'échantillonnage contrôlée permettant de garantir (avec un seuil de confiance donné *a priori*) la sélection effective de l'arbre d'analyse le plus probable (section 4). La section 5 complète l'analyse en détaillant certains aspects d'implémentation des méthodes décrites.

2. Problématique

2.1. Définitions et notations

Une *grammaire à substitution d'arbres* est une grammaire formelle dont les règles sont des arbres (appelés ci-après *arbres de référence*) dont les racines et les nœuds intérieurs sont des symboles non-terminaux de la grammaire et les feuilles des symboles quelconques (terminaux ou non-terminaux). Pour un arbre a, on notera $r(a)$ sa racine, $F(a)$ la séquence ordonnée[4] de ses feuilles et $f_i(a)$ la i-ième feuille dans $F(a)$.

Sur l'ensemble des arbres d'analyse[5], on définit une loi interne $\circ$ qui à deux arbres a et b associe l'arbre $c = a \circ b$ résultant de la substitution par b de la feuille non-terminale la plus à gauche de a, si cette feuille est égale à la racine de b, et l'arbre vide sinon. $\circ$ n'étant pas associative, $a_1 \circ ... \circ a_m$ sera interprété par convention comme $(...(a_1 \circ a_2) \circ ...) \circ a_m)$. Une *décomposition* d'un arbre d'analyse a quelconque est la donnée de m sous-arbres de référence a_1, ..., a_m, tels que $a = a_1 \circ ... \circ a_m$.

Dans le cadre du modèle DOP, la probabilisation d'une grammaire à substitution d'arbres s'effectue alors de la façon suivante : à chaque arbre de référence a est associé un coefficient stochastique $p(a)$ (équivalent au coefficient stochastique associé aux règles dans une SCFG) vérifiant la contrainte stochastique $\sum_{b|r(b)=r(a)} p(b) = 1$. Ce coefficient stochastique sera appelé *probabilité élémentaire* de a.

La probabilité P d'une décomposition $a_1 \circ ... \circ a_m$ est alors définie comme le produit des probabilités élémentaires de chacun des a_i, et la probabilité d'un arbre d'analyse quelconque a, appelée DOP-*probabilité* et notée $P_{\text{DOP}}(a)$[6], est définie comme la somme des probabilités de toutes ses décompositions. Notons la différence, pour un arbre élémentaire, entre sa DOP-probabilité et sa probabilité élémentaire: la première est toujours supérieure ou égale à la seconde et ne lui est égale que dans le cas particulier où l'arbre de référence considéré ne possède pas d'autre décomposition que lui-même sur l'ensemble des arbres de référence.

2.2. Extraction de l'arbre le plus probable

Le fait de pouvoir produire toutes les analyses de la séquence à analyser en un temps polynômial ne garantit en aucun cas que l'on puisse déterminer l'analyse la plus probable en un temps polynômial. En effet, non seulement une séquence donnée peut avoir un nombre exponentiel

4. graphiquement de gauche à droite

5. ou « arbres syntaxiques »

6. Ceci est une notation un peu abusive car en toute rigueur il s'agit de la probabilité $P_{\text{DOP}}(a, w)$ puisqu'en effet la somme des DOP-probabilités d'arbre couvrant la même phrase est égale à la probabilité de la phrase, c.-à-d. $P(w) = \sum_{a|F(a)=w} P_{\text{DOP}}(a)$. De ce fait, on notera également $P_{\text{DOP}}(a|w)$ le rapport $P_{\text{DOP}}(a)/P(w)$.

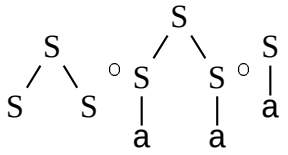
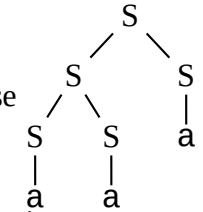
d'arbres d'analyse, mais un arbre d'analyse peut de plus lui-même avoir un nombre exponentiel de décompositions. Il n'est donc pas possible, dans le cas général, d'extraire de façon efficace l'arbre d'analyse le plus probable simplement en parcourant toutes les décompositions possibles. En fait, il a été démontré que ce problème d'extraction est NP-difficile (Sima'an, 1996).

Cependant, le fait qu'il n'existe pas d'algorithme polynômial *exact* permettant de trouver l'arbre d'analyse le plus probable ne signifie pas qu'il n'existe pas d'algorithme polynômial permettant d'*estimer* cet arbre d'analyse avec une probabilité d'erreur aussi petite que possible. C'est précisément cette stratégie qui est appliquée dans (Bod, 1995). Le problème est alors de trouver un algorithme d'échantillonnage probabiliste qui permette l'extraction d'un arbre d'analyse en un temps polynômial[7] et soit compatible avec le modèle probabiliste utilisé, c.-à-d. tel que la probabilité d'obtenir par échantillonnage un arbre d'analyse a soit être égale à $P_{\text{DOP}}(a|w)$, sa DOP-probabilité sachant la séquence w à analyser.

3. Échantillonnage

Pour une séquence de mots à analyser w fixée, la sélection d'une décomposition parmi toutes les décompositions possibles de tous les arbres d'analyse possibles de w s'effectue par une suite de choix successifs (et indépendants) d'*éléments de décomposition*.

Plus précisément, si un élément de décomposition e est un couple $(u(e), \tau(e))$, où $u(e)$ est un arbre de référence et $\tau(e)$ un tuple de taille $p+1$ d'indices strictement croissants, p étant le nombre de feuilles de $u(e)$[8], alors une décomposition a_1, ..., a_m de l'arbre d'analyse a d'une séquence $w = w_1 \ldots w_n$ est obtenue par la sélection successive des éléments de décomposition e_1, ..., e_m tels que pour tout k entre 1 et m, $e_k = (a_k, \tau_k)$ avec τ_k la partition induite par les feuilles de a_k sur la sous-séquence de w couverte dans a par la racine de a_k[9].

Par exemple la décomposition  de l'arbre d'analyse  correspond à la sélection successive des éléments de décomposition e_1, e_2, e_3 suivants :

$e_1 = \left(\begin{array}{c} \text{S} \\ \text{S} \quad \text{S} \end{array}, (1,3,4) \right)$, $e_2 = \left(\begin{array}{c} \text{S} \\ \text{S} \quad \text{S} \\ \text{a} \quad \text{a} \end{array}, (1,2,3) \right)$ et $e_3 = \left(\begin{array}{c} \text{S} \\ \text{a} \end{array}, (3,4) \right)$.

Il est important de souligner ici que c'est précisément la donnée du second élément (la partition) qui permet d'effectuer le choix des éléments d'une même décomposition d'une façon qui ne **dépende que du père** de l'élément choisi et non pas de l'ensemble de la décomposition (*i.e.* un choix local). Il est à noter que le choix d'un sous-arbre seul (comme par exemple indiqué dans (Rajman, 1995)[10]), ne permet pas de déterminer de façon unique la décomposition choisie.

7. Ce qui signifie en particulier que les probabilités d'échantillonnage utilisées doivent également pouvoir être calculées en un temps polynômial.

8. Pour $\tau(e) = (i_1, \ldots, i_{p+1})$, on notera $i_k(e) = i_k$, $i(e) = i_1(e) = i_1$, $j(e) = i_{p+1}(e) = i_{p+1}$.

9. autrement dit : la l-ième feuille de a_k domine dans a la sous-séquence $w_{i_l(e_k)} \ldots w_{i_{l+1}(e_k)-1}$ de w. Par exemple $\left(\begin{array}{c} \text{X} \\ \text{Y} \quad \text{Z} \end{array}, (1,3,4) \right)$ signifie que Y doit dominer w_1 w_2 et Z w_3.

10. Dans (Rajman, 1995), c'est plus précisément une règle de la grammaire équivalente qui est choisie ; ce qui

En effet, une **même** règle peut appartenir à **plusieurs** décompositions différentes. Par exemple, avec la grammaire `S -> S S | a`, la séquence `aaa` possède **deux** arbres d'analyse où la **même** règle s'applique au sommet. Il n'est donc pas suffisant de se contenter de choisir des règles pour caractériser une décomposition.

L'ensemble des éléments de décomposition parmi lesquels on peut choisir pour étendre un non-terminal X dans l'analyse de $w_i...w_j$ ($1 \leq i \leq j \leq n$)) est l'ensemble, appelé ci-après *domaine d'extension* de X en (i, j) défini par :

$$\mathcal{D}(X, i, j) = \{e \mid r(u(e)) = X, i(e) = i, j(e) = j + 1\}$$

Le problème central est alors de déterminer la probabilité avec laquelle on doit choisir les éléments de décomposition au sein des domaines d'extension, de sorte que, pour une séquence w à analyser donnée, la probabilité d'obtenir une séquence d'éléments de décomposition correspondant à un arbre d'analyse de w soit égale à la DOP-probabilité de cet arbre.

Posons

$$P_0(e) = \begin{cases} p(u(e)) \cdot \prod_{1 \leq k \leq \text{nt}(e)} \sum_{e' \in \Delta_k(e)} P_0(e') & \text{si nt}(e) \neq 0 \\ p(u(e)) & \text{si nt}(e) = 0 \end{cases}$$

où nt(e) est le nombre de feuilles non-terminales de $u(e)$ et

$$\Delta_k(e) = \mathcal{D}(f_{\zeta(k)}(u(e)), i_{\zeta(k)}(e), i_{\zeta(k)+1}(e) - 1)$$

avec $\zeta(i)$ l'indice de la i-ème feuille non-terminale[11]. $P_0(e)$ peut être facilement calculé au cours de la phase d'analyse (cf section 5).

On définit alors la probabilité d'échantillonnage d'un élément e au sein de l'ensemble $\mathcal{D}(X, i, j)$ par :

$$P_E(e) = \frac{P_0(e)}{\sum_{e' \in \mathcal{D}(X,i,j)} P_0(e')}$$

et le mécanisme d'échantillonnage des décompositions d'arbres d'analyse d'une séquence $w = w_1...w_n$ donnée est le suivant :

1. On initialise le processus en choisissant aléatoirement suivant P_E un élément de décomposition dans le domaine d'extension $\mathcal{D}(\mathtt{S}, 1, n)$, où `S` est le symbole de plus haut niveau de la grammaire.
2. Soit $\xi = (e_1, ..., e_k)$ la séquence d'éléments de décomposition obtenue après k sélections aléatoires successives.
   - si nt$(\xi) = 0$, l'échantillonnage est terminé[12].
   - sinon, il existe alors nécessairement un i et un j tels que la feuille non-terminale X la plus à gauche de $u(e_1) \circ ... \circ u(e_k)$ soit la j-ème de $u(e_i)$. Le $k+1$-ième élément de décomposition est obtenu en choisissant aléatoirement suivant P_E un élément dans le domaine d'extension $\Delta_j(e_i)$. On appellera e_i le « père » de e_{k+1}. De plus, par construction, $\Delta_j(e_i) = \mathcal{D}(r(u(e_{k+1})), i(e_{k+1}), j(e_{k+1}) - 1)$ que nous noterons par la suite $\delta(e_{k+1})$.

est strictement équivalent au choix d'un sous-arbre de référence seul.

11. En toute rigueur $\zeta(i)$ dépend de e. En cas d'ambiguïté, nous le noterons alors $\zeta_e(i)$.
12. En notant nt$(e_1, ..., e_k)$ le nombre de feuilles non-terminales de $u(e_1) \circ ... \circ u(e_k)$

En termes moins formels, les choix possibles pour un nouvel élément de décomposition à un moment donné du processus d'échantillonage sont tous les éléments de décomposition permettant réécrire le non-terminal courant, tout en respectant la contrainte de couverture de la chaîne imposée par le choix de l'élément « père ».

La probabilité d'obtention d'une séquence d'éléments de décomposition $d = (e_1, ..., e_m)$ est alors le produit des probabilités d'échantillonnage. En effet :

$$P_E(d) = P_E(e_1, ..., e_m) = P_E(e_1) \cdot \prod_{2 \leq i \leq m} P_E(e_i | e_1, ..., e_{i-1}) = P_E(e_1) \cdot \prod_{2 \leq i \leq m} P_E(e_i | \text{père de } e_i)$$

puisque le choix d'un élément ne dépend que de son père. On a donc $P_E(d) = \prod_{e \in d} P_E(e)$.

Montrons alors que cette probabilité d'échantillonnage correspond bien à la probabilité de la décomposition obtenue (sachant w). C'est-à-dire que

$$P_E(e_1, ..., e_m) = P_{\text{DOP}}(u(e_1) \circ ... \circ u(e_n) | w) = \frac{1}{P(w)} \cdot \prod_{i=1}^{i=m} p(u(e_i))$$

Soit $\Theta(p)$ la proposition : Toute décomposition terminale $e_1, ..., e_m$ de **profondeur** p vérifie

$$P_E(e_1, ..., e_m) = \frac{1}{\sum_{e' \in \delta(e_1)} P_0(e')} \prod_{1 \leq i \leq m} p(u(e_i))$$

Montrons par récurrence sur p que $\Theta(p)$ est vraie :

$\mathbf{p = 1}$ Une décomposition de profondeur 1 a nécessairement un seul élément e_1 qui, du fait que la décomposition est terminale, n'a que des feuilles terminales. La proposition $\Theta(1)$ est alors triviale puisque, par définition,

$$P_E(e_1) = \frac{P_0(e_1)}{\sum_{e' \in \delta(e_1)} P_0(e')} = \frac{1}{\sum_{e' \in \delta(e_1)} P_0(e')} \, p(u(e_1))$$

Récurrence Supposons $\Theta(p)$ et montrons que $\Theta(p+1)$ est vraie. Soit donc $e_1, ..., e_m$ une décomposition terminale de profondeur $p+1$.

Alors, pour tout e_i, $u(e_i)$ est contenu dans un unique sous-arbre terminal a_j de racine l'un des fils non-terminaux de e_1 contenant $u(e_i)$. De plus, du fait de la convention consistant à réécrire en premier la feuille non-terminale la plus à gauche, un tel a_j définit alors de manière univoque deux indices n_j et m_j tels que $2 \leq n_j \leq m_j \leq m$ et $a_j = u(e_{n_j}) \circ ... \circ u(e_{m_j})$. Nous pouvons donc écrire :

$$\begin{aligned} P_E(e_1, ..., e_m) &= \prod_{1 \leq i \leq m} P_E(e_i) = P_E(e_1) \cdot \prod_{2 \leq i \leq m} P_E(e_i) \\ &= \frac{P_0(e_1)}{\sum_{e' \in \delta(e_1)} P_0(e')} \cdot \prod_{1 \leq j \leq \text{nt}(e_1)} P_E(e_{n_j}, ..., e_{m_j}) \end{aligned}$$

en regroupant les produits de sorte à faire apparaître les probabilités d'échantillonnage de chacune des décompositions terminales (*i.e.* les $(e_{n_j}, ..., e_{m_j})$) associées aux feuilles non-terminales de $u(e_1)$.

De plus, chacun des $(e_{n_j}, ..., e_{m_j})$ est une décomposition de profondeur au plus p (sinon $e_1, ..., e_m$ serait de profondeur plus grande que $p+1$). Donc par hypothèse de récurrence :

$$P_E(e_{n_j}, ..., e_{m_j}) = \frac{1}{\sum_{e' \in \delta(e_{n_j})} P_0(e')} \prod_{n_j \leq i \leq m_j} p(u(e_i))$$

Ce qui nous donne :

$$\begin{aligned} P_E(e_1, ..., e_m) &= \frac{P_0(e_1)}{\sum_{e' \in \delta(e_1)} P_0(e')} \cdot \prod_{1 \leq j \leq \text{nt}(e_1)} \left[\frac{1}{\sum_{e' \in \delta(e_{n_j})} P_0(e')} \prod_{n_j \leq i \leq m_j} p(u(e_i)) \right] \\ &= \frac{1}{\sum_{e' \in \delta(e_1)} P_0(e')} \cdot \frac{P_0(e_1)}{\prod_{1 \leq j \leq \text{nt}(e_1)} \sum_{e' \in \delta(e_{n_j})} P_0(e')} \cdot \prod_{1 \leq j \leq \text{nt}(e_1)} \prod_{n_j \leq i \leq m_j} p(u(e_i)) \end{aligned}$$

Or par définition de e_{n_j}, $\delta(e_{n_j}) = \Delta_j(e_1)$, donc :

$$\frac{P_0(e_1)}{\prod_{1 \leq j \leq \text{nt}(e_1)} \sum_{e' \in \delta(e_{n_j})} P_0(e')} = \frac{P_0(e_1)}{\prod_{1 \leq j \leq \text{nt}(e_1)} \sum_{e' \in \Delta_j(e_1)} P_0(e')}$$

ce qui, par définition de $P_0(e_1)$, vaut $p(u(e_1))$.

On a donc, en utilisant le même argument que celui qui nous a permis de décomposer le produit à la première étape :

$$\begin{aligned} P_E(e_1, ..., e_m) &= \frac{1}{\sum_{e' \in \delta(e_1)} P_0(e')} \cdot p(u(e_1)) \cdot \prod_{1 \leq j \leq \text{nt}(e_1)} \prod_{n_j \leq i \leq m_j} p(u(e_i)) \\ &= \frac{1}{\sum_{e' \in \delta(e_1)} P_0(e')} \prod_{1 \leq i \leq m} p(u(e_i)) \end{aligned}$$

ce qui termine la démonstration par récurrence.

De plus, par définition $\delta(e_1) = \mathcal{D}(S, 1, n)$ et donc par construction $\sum_{e \in \delta(e_1)} P_0(e) = P(w)$. □

On a donc bien, pour toute décomposition, égalité entre la probabilité d'obtenir cette décomposition par le processus d'échantillonnage précédemment décrit et la probabilité conditionnelle de cette décomposition sachant w. Il en résulte alors que la probabilité d'obtenir un arbre syntaxique par ce processus d'échantillonnage est aussi égale à sa DOP-probabilité sachant w. En effet :

$$P_E(a) = \sum_{\substack{e_1, ..., e_m \mid \\ u(e_1) \circ ... \circ u(e_m) = a}} P_E(e_1, ..., e_m) = \sum_{a_1 \circ ... \circ a_m = a} P_{\text{DOP}}(a_1 \circ ... \circ a_m | w) = P_{\text{DOP}}(a|w)$$

4. Calcul de l'analyse la plus probable par échantillonnage

Nous avons montré dans la section précédente que, si l'on choisit convenablement la loi de probabilité utilisée pour l'échantillonnage au sein des arbres syntaxiques associés à une séquence de mots donnée, la probabilité de sélectionner l'un quelconque des ces arbres est égale à la probabilité conditionnelle de cet arbre au sens du modèle DOP.

L'idée intuitive qui est alors à la base de l'approche par échantillonnage est de chercher à identifier l'arbre le plus probable sur la base de sa fréquence relative d'occurrence au sein d'un échantillon produit de façon aléatoire à l'aide du mécanisme d'échantillonnage détaillé à la section 3. En effet, puisque cette fréquence relative va tendre vers la probabilité de l'arbre, pour une taille d'échantillon suffisamment grande, l'arbre le plus fréquent sera aussi l'arbre le plus probable.

Toute la question est alors de contrôler le mécanisme d'échantillonnage de façon à garantir (avec une probabilité d'erreur fixée *a priori*) la justesse de la procédure de sélection fréquentielle. Ceci correspond à un problème classique d'ordonnancement statistique car, si k est le nombre d'arbres associés à la séquence de mots analysée (k peut être calculé sans sur-coût algorithmique lors de l'analyse syntaxique), rechercher l'arbre le plus probable revient à déterminer la modalité la plus probable d'une variable aléatoire discrète à k modalités (chacune des modalités correspondant à l'un des arbres d'analyse possibles) suivant une loi multinomiale.

La méthode de sélection proposée par R. Bod (1995) constitue un premier exemple d'un tel mécanisme. Elle souffre cependant de l'imprécision introduite par l'estimation de la probabilité de sélection erronée utilisée [13] dont il est extrêmement difficile d'évaluer l'impact sur la qualité des résultats obtenus. De plus, la nature purement *séquentielle* de la méthode (qui fournit en fait un critère d'arrêt pour l'échantillonnage) a pour conséquence qu'elle ne permet pas un calcul *a priori* de la taille d'échantillon nécessaire.

Pour ces différentes raisons, il nous a paru important d'utiliser des méthodes plus sophistiquées telles que celles proposées et testées dans la littérature spécialisée (sélection et ordonnancement de populations statistiques). Dans cette contribution, nous nous limiterons à la présentation de la méthode de sélection séquentielle Bechhofer-Kiefel-Sobel avec troncature (Bechhofer & Goldsman, 1985) (appelée ci-après $BKST$), qui est une des meilleures connues pour le problème de la sélection de la modalité la plus probable d'une variable aléatoire discrète suivant une loi multinomiale.

La procédure de sélection $BKST$ est en fait la combinaison d'une méthode de sélection séquentielle (appelée ci-après BKS) [14] (Bechhofer *et al.*, 1968) et d'une méthode de sélection non-séquentielle (appelée ci-après BEM) (Bechhofer *et al.*, 1959).

4.1. La procédure de sélection non-séquentielle BEM

Pour toute variable aléatoire discrète à k modalités ($k \geq 2$) suivant une loi de probabilité multinomiale, nous noterons ci-après par $p_{[1]}, ..., p_{[k]}$ les k paramètres, classés par ordre décroissant ($p_{[1]} \geq ... \geq p_{[k]}$), de la loi multinomiale [15].

13. En effet, en utilisant les notations $p_{[i]}$ et $f_{[i]}$ introduites dans les sections 4.1 et 4.2 ci-après, dans la méthode de Bod l'erreur $\sum_{i>1}(1-(\sqrt{p_{[1]}}-\sqrt{p_{[i]}})^2)^N$ est simplement estimée par $\sum_{i>1}(1-(\sqrt{f_{[1]}}-\sqrt{f_{[i]}})^2)^N$.

14. correspondant à une spécialisation pour les lois multinomiales d'une méthode plus générale d'ordonnancement des populations de Koopman-Darmois

15. évidemment $p_{[1]} + ... + p_{[k]} = 1$

Si l'on considère alors le problème de sélectionner la modalité la plus probable sur la base des fréquences d'occurrence des modalités dans un échantillon aléatoire de réalisations indépendantes de la variable aléatoire, on peut démontrer (Kesten & Morse, 1959) que, pour toute variable aléatoire discrète à k modalités suivant une loi multinomiale vérifiant $p_{[1]} \geq \theta\, p_{[2]}$ (avec $\theta > 1$ fixé), la probabilité de sélection correcte (*i.e.* les cas où la modalité la plus fréquente –avec tirage aléatoire entre *ex-aequo* si nécessaire– est effectivement la plus probable) est toujours minorée par la probabilité de sélection correcte P_{min} associée à la loi k-nomiale dont tous les paramètres sont égaux sauf un, le plus grand [16]. De plus, pour toute taille d'échantillon N, P_{min} (qui est fonction de k, θ et N) peut être effectivement calculée par addition des probabilités de toutes les situations de sélection correcte, qui peuvent être explicitement énumérées [17]. La fonction $P_{\text{min}}(k, \theta, N)$ peut donc être tabulée (voir par exemple (Bechhofer *et al.*, 1959)).

La méthode de sélection non-séquentielle BEM est alors extrêmement simple :

1. choisir une valeur minimale θ pour le rapport $\frac{P_{[1]}}{P_{[2]}}$ (cette valeur correspond à une estimation *a priori* de la difficulté du problème) et une valeur minimale acceptable P_{min} pour la probabilité de sélection correcte;
2. déterminer, à partir des valeurs tabulées, la plus petite valeur de N telle que $P_{\text{min}} \leq P_{\text{min}}(k, \theta, N)$ [18];
3. déterminer les fréquences d'occurrence des modalités dans un échantillon aléatoire de N réalisations indépendantes de la variable aléatoire.

Si $P_{[1]}$ et $P_{[2]}$ vérifient bien $P_{[1]} \geq \theta\, P_{[2]}$, la modalité la plus fréquente (avec tirage aléatoire en cas d'*ex-aequo*) est alors effectivement, avec une probabilité supérieure P_{min}, la modalité la plus probable.

4.2. La procédure de sélection séquentielle BKS et sa troncature

La procédure de sélection séquentielle BKS repose sur la propriété fondamentale suivante (Bechhofer *et al.*, 1968; Levin, 1984) : pour toute variable aléatoire discrète à k modalités suivant une loi multinomiale vérifiant $p_{[1]} \geq \theta\, p_{[2]}$ (avec $\theta > 1$ fixé), si l'on note $f_{[1]}, ..., f_{[k]}$ les fréquences relatives d'occurrence (classées par ordre décroissant) des k modalités dans un échantillon aléatoire de taille N de réalisations indépendantes de la variable aléatoire, alors la probabilité de sélection correcte est toujours minorée par la valeur $P_{\text{min}} = \frac{1}{1+Z_N}$, où $Z_N = \sum_{i>1} (\frac{1}{\theta})^{(f_{[1]} - f_{[i]})}$.

La méthode de sélection séquentielle BKS est alors extrêmement simple : choisir les valeurs θ et P_{min}; puis échantillonner itérativement la variable aléatoire en tenant à jour les fréquences d'occurrence $f_{[i]}$ des modalités jusqu'à ce que $P_{\text{min}} \leq \frac{1}{1+Z_N}$.

Si $P_{[1]}$ et $P_{[2]}$ vérifient bien $P_{[1]} \geq \theta P_{[2]}$, la modalité la plus fréquente (avec tirage aléatoire en cas d'*ex-aequo*) est alors effectivement, avec une probabilité supérieure P_{min}, la modalité la plus probable.

16. c.-à-d. la loi multinomiale de paramètre $(\theta(\theta + k - 1), \theta + k - 1, ..., \theta + k - 1)$.

17. Pour de grandes valeurs de N ou de k, le nombre de situations à énumérer devient prohibitif et $P_{\text{min}}(k, \theta, N)$ doit elle-même être estimée par une méthode Monte-Carlo.

18. Pour les grandes valeurs de N, la formule approchée suivante peut être utilisée : $N = \left\lceil \frac{\Lambda(k,P)^2 (1+\sqrt{\frac{\theta}{(k-1)(\theta+k-2)}})}{4(\arcsin(\sqrt{\frac{\theta}{\theta+k-1}}) - \arcsin(\sqrt{\frac{1}{\theta+k-1}}))\ 2} \right\rceil$, où $\Lambda(k, P)$ est l'intégrale $(k-1)$-uple de la densité de probabilité normale standard de corrélation uniforme 0.5, tabulée dans (Gupta, 1956) avec les notations de (Bechhofer *et al.*, 1959).

La méthode de sélection avec troncature $BKST$ consiste alors tout simplement à intégrer la méthode BEM dans la méthode BKS en ajoutant au critère d'arrêt la condition consistant à arrêter l'échantillonnage dès que la taille de l'échantillon a atteint la taille minimale N_{min} calculée *a priori* comme indiqué dans la section 4.1.

5. Implémentation

Le but de cette section est d'expliciter quelques aspects d'implémentation importants de notre algorithme d'échantillonage. Nous nous intéresserons en particulier au calcul, durant la phase de construction de la forêt d'analyse, des probabilités P_0 nécessaires pour l'échantillonnage (ce calcul doit pouvoir être fait dans un temps polynômial) ; puis à la nature et au coût de l'extraction aléatoire d'une décomposition.

5.1. Calcul ascendant (bottom-up) des probabilités

L'algorithme d'analyse syntaxique utilisé dans notre approche est une version généralisée de l'algorithme CYK et de l'algorithme Earley bottom-up, voisine des travaux de Graham et al. (1980). Les détails de cet algorithme sont donnés dans (Chappelier & Rajman, 1998). Parmi ses caractéristiques importantes on peut citer (1) qu'il permet de traiter des grammaires SCFG quelconques (*i.e.* non limitées à la forme normale de Chomsky) en effectuant, à l'aide d'une généralisation des « Earley items », une binarisation dynamique de la grammaire [19] ;(2) que le calcul des probabilités [20] de ces « items généralisés » peut être réalisé de façon ascendante sans augmentation de la complexité algorithmique de l'analyse syntaxique [21]. Ce calcul s'effectue en effet de proche en proche au cours de la construction « bottom-up » des interprétations, en multipliant le coefficient de la règle appliquée par les deux probabilités partielles (déjà calculées) des deux constituants utilisés.

5.2. Extraction descendante (top-down) des décompositions

Contrairement à l'approche proposée par R. Bod (1995), nous avons choisi une méthode d'extraction descendante, semblable à celle utilisée pour l'extraction des arbres d'analyse dans le cadre des SCFG classiques. La différence importante est que les choix (des éléments constitutifs des décompositions) sont ici aléatoires au lieu de correspondre à la recherche d'une probabilité optimale. À chaque étape un nouveau constituant de décomposition est choisi de façon aléatoire suivant la probabilité P_E dont les composants P_0 ont été calculés durant la phase d'analyse. Ce processus se déroule itérativement de façon descendante à partir du symbole de plus haut niveau jusqu'à obtention d'une décomposition complète et la démonstration effectuée à la section 3 assure que la probabilité d'obtenir un arbre donné (sachant la séquence à analyser) est alors effectivement égale à sa DOP-probabilité.

Notons par ailleurs qu'à la différence de la méthode proposée par R. Bod (1995), qui, pour chaque extraction, nécessite un tirage aléatoire pour tous les terminal dans toutes les cases de la table, il nous suffit de réaliser autant de tirages que d'éléments présents dans la décomposition.

19. pour une définition détaillée des « items » stockés dans une case de la table, se reporter à (Chappelier & Rajman, 1998). Schématiquement, ces éléments sont les constituants d'éléments de décomposition tels qu'introduits en section 3.

20. qui correspondent au $P_0(e)$ de la section 3.

21. qui reste en $\mathcal{O}(n^3)$, où n est la taille de la séquence à analyser.

6. Conclusion

Dans la présente contribution, nous avons présenté deux résultats importants pour l'extraction d'arbres d'analyse dans le modèle DOP :

- Tout d'abord nous avons démontré de façon formelle le bien-fondé de la mise en œuvre de méthodes de type Monte-Carlo. A notre connaissance, une telle démonstration n'a été, au mieux, que brièvement esquissée dans les publications disponibles sur le sujet.
- Ensuite nous avons proposé une méthode permettant de contrôler la qualité de l'échantillonnage et de garantir (avec un seuil de confiance fixé *a priori*) que l'arbre d'analyse sélectionné est effectivement l'arbre d'analyse le plus probable.

Ces deux résultats permettent de fonder sur une base théorique plus solide les expérimentations nécessaires pour une meilleure évaluation du modèle syntaxique DOP qui apparaît aujourd'hui comme l'un des candidats prometteurs de la classe des grammaires faiblement sensibles au contexte.

Références

BECHHOFER R., ELMAGHRABY S. & MORSE N. (1959). A single-sample multiple-decision procedure for selecting the multinomial event which has the largest probability. *Ann. Math. Statist.*, **30**, 102–119.

BECHHOFER R. & GOLDSMAN D. (1985). Truncation of the Bechhofer-Kiefer-Sobel sequential procedure for selecting the multinomial event which has the largest probability. *Communications in Statistics: simulation and computation*, **14**(2), 283–315.

BECHHOFER R., KIEFER J. & SOBEL M. (1968). *Sequential Identification and Ranking Procedures*. University of Chicago Press, Chicago.

BOD R. (1995). *Enriching Linguistics with Statistics: Performance Models of Natural Language*. Amsterdam (The Netherlands): Academische Pers.

CHAPPELIER J.-C. & RAJMAN M. (1998). A generalized cyk algorithm for parsing stochastic cfg. In *TAPD'98 Workshop*, p. 133–137, Paris (France).

GRAHAM S. L., HARRISON M. A. & RUZZO W. L. (1980). An improved context-free recognizer. *ACM Transactions on Programming Languages and Systems*, **2**(3), 415–462.

GUPTA S. (1956). *On a decision rule for a problem in ranking means*. Number 150 in Institute of Statistics Mimeograph Series. University of North Carolina.

KESTEN H. & MORSE N. (1959). A property of the multinomial distribution. *Ann. Math. Statist.*, **30**, 120–127.

LEVIN B. (1984). On a sequential selection procedure of Bechhofer, Kiefer and Sobel. *Statistics and Probability Letters*, **2**, 91–94.

RAJMAN M. (1995). *Apports d'une approche à base de corpus aux techniques de traitement automatique du langage naturel*. PhD thesis, École Nationale Supérieure des Télécommunications, Paris.

SIMA'AN K. (1996). Computational complexity of probabilistic disambiguation by means of tree grammars. In *Proceedings of COLING'96*, Copenhagen (Denmark). cmp-lg/9606019.