
Document de travail - Workspace et Workers

Fernandes-Pinto-Fachada Sarah,
Schrottenloher André,
Angibault Antonin,
Hufschmitt Théophile,
Cao Zhixing,
Boisseau Guillaume

12 novembre 2014

Important

MODIFIEZ CE DOCUMENT AUTANT QUE VOUS VOULEZ SELON VOS IDÉES DU MOMENT, C'EST QUELQUE CHOSE DE FULLY COLLABORATIF

1 Présentation

Le Workspace est la structure la plus importante ; plus encore que le réseau de concept. Il nous faut la définir rigoureusement, ainsi que ce que font les workers, afin d'être sûrs à quoi nous arrivons.

C'est aussi ici que nous allons le plus nous démarquer de Copycat notamment, puisque nous voulons faire du résumé automatique : pour ce qui était du réseau de concepts, nous étions encore dans la continuité.

Cahier des charges du workspace

- Contenir des instances des nœuds du réseau de concepts (ie : des concepts signifiants) ;
- Reprendre les informations importantes du texte ;
- Ajouter de l'information ne provenant pas directement du texte, mais jugée importante ? A voir ! (Ex. trivial des nœuds fortement activés dans le RC)
- Ne pas contenir d'information trop évidente : ex. il ne faut pas répéter que Wayne Rooney est un joueur de football, même si c'était un sous-entendu du texte. C'était sous-entendu justement parce que ça ne sert pas de le répéter, a priori ;
- Contenir une information exploitable ;
- Tracer du début à la fin du programme le rôle grammatical des concepts tirés du texte, dans leur contexte. Exemple : Wayne Rooney est un sujet. Il faut savoir tout au long du programme, y compris lorsque le nœud Wayne Rooney sera créé dans le Workspace, que c'est un sujet. Cela permet de ne pas jeter de l'information par la fenêtre.

Cahier des charges des workers Il nous en faut pour :

- Créer des instances de nœuds du RC dans le W ;
- Activer des nœuds du RC ;
- Faire des liens entre les nœuds du W ;
- **Prendre en compte toute l'information grammaticale/syntaxique** et la transférer au W, car elle n'apparaît pas dans le RC. Exemple : un worker spécial qui crée un sujet (Wayne Rooney) dans le W, et va ensuite spécifiquement chercher "ce que fait Wayne Rooney". Un worker spécial qui crée un objet (World Cup) dans le W, et va ensuite chercher "ce qui arrive à" ou "ce qu'on dit sur" World Cup.

2 Idées provenant de Copycat/Bascet

2.1 Copycat

Workspace Le workspace contient des instances du réseau de concepts (Slipnet) organisées en :

- **Description** d'objets
- **Relations** entre objets (lien de succession entre lettres par ex.)
- **Groupes** d'objets
- Correspondances entre objets de chaînes différentes ("est analogue à")
- Règles qui décrit le changement entre la chaîne initiale et la chaîne modifiée
- Règle traduite qui décrit comment modifier la chaîne cible (peut être issue de la règle précédente en remplaçant des concepts, par exemple)

Je retiens : description, relations, groupes d'objets.

codrack Le codrack contient les codelets, agents choisis stochastiquement mais pas n'importe comment.

Ces agents (codelets) représentent des pressions.

- Codelets ascendants : pressions présentes dans toutes les situations (trouver des relations...)
- Codelets descendants : pressions provenant de la situation courante

À chaque pas d'exécution, un codelet est exécuté et supprimé du codrack. Chaque codelet a une certaine urgence, mais elle ne détermine pas sa priorité (le choix se fait en partie au hasard) (elle représente l'intérêt de sa tâche).

- Les codelets activent les noeuds du Slipnet
- Les noeuds actifs ajoutent des codelets dans le Codrack (descendants, par exemple)
- Les codelets lancent d'autres codelets
- Des codelets ascendants sont créés en permanence

Je laisse pour plus tard le problème de la température.

2.2 Bascet

Agents

- Chacun a son **urgence**

- Chacun a un **nœud père** qui l'a lancé (RC)
- Ils peuvent lire et créer les objets du Blackboard (active les nœuds du RC dont l'objet est une instance)
- **Opinionnent** sur la validité d'un objet

Blackboard Contient des objets, instances des noeuds du RC :

- Contenu
- Père (noeud dont il est instance)
- Importance (dépend du père et des liens avec les autres objets)
- Satisfaction (dépend des agents)
- Eminence (objet à traiter)

3 Conclusions

Je pense reprendre dans un premier temps un mélange de ces caractéristiques, et adapter ensuite à notre problème.

Définition 1 (Workspace) *Le Workspace est une structure constituée de :*

- *Nœuds dits "du Workspace", issus de l'instantiation de concepts ;*
- *Nœuds "fantômes" issus de nœuds du RC très activés, mais non présents dans le texte ;*
- *Liens entre ces nœuds.*

Définition 2 (Nœud du Workspace) *Un tel nœud comporte :*

- *Un **nœud père** dans le RC et une étiquette identique à son père ;*
- *Une **importance** qui dépend de celle de son père ;*
- *Un **rôle** plus ou moins défini issu du parsing syntaxique/grammatical ;*
- *Une **satisfaction** calculée par les workers, selon le rôle, et les liens qu'ils ont réussi à créer (par exemple, pour un sujet, on s'attend à ce que celui-ci agisse) ;*

Moins le nœud est satisfait, plus il faut chercher, donc lancer des workers qui traitent de lui OU augmenter l'urgence de ces workers.

Définition 3 (Worker) *Un worker correspond à une tâche. De manière générale, il agit à la fois sur le Workspace et sur le RC. Un worker comporte toujours :*

- *Une urgence ;*
- *Une mission définie ;*

4 Liste actualisée des workers et détail de leur activité

Remarques générales :

- Moins un nœud est satisfait, plus il doit générer de workers constructeurs
- La satisfaction d'un nœud doit augmenter lorsque quelque chose est construit sur ce nœud (par exemple lorsqu'on crée des liens)
- Les workers les plus importants sont ceux qui activent les concepts

- Ensuite viennent ceux qui construisent quelque chose
- Ceux qui font des calculs généraux
- etc

Worker Classe-mère. Augmente le temps lors du lancement du worker, et affiche un message si cette fonction est activée.

4.1 Workers qui n'agissent directement que sur le RC

Activate Urgence : 100 Active un nœud du RC avec un certain montant (l'activation d'un nœud ne peut pas dépasser 100).

- Si le nœud activé dépasse un certain seuil (80), push un WriteConcept correspondant
- Dans tous les cas, push des workers Compute pour tous les nœuds dans linksOut

Compute Urgence : 50

Recalcule l'activation d'un nœud du RC. Push des workers Compute pour tous les nœuds dans linksOut.

4.2 Builders : workers qui agissent sur le workspace, et le RC

ReadAndWrite Urgence : 90

Écrit un nouveau mot dans le workspace, en prenant en argument le mot du texte et son rôle grammatical (notion assez vague) issu du parsing.

- S'il y a un concept correspondant dans le RC, pas de nœud réel dans le workspace, pas de nœud fantôme dans le workspace :
 - Push un worker Activate(concept correspondant)
 - Crée un nœud réel correspondant à ce mot
 - Push une recherche de liens à ce nœud réel, selon son rôle grammatical
- S'il y a un concept, pas de nœud réel, un nœud fantôme :
 - Push un worker Activate(concept correspondant)
 - Crée un nœud réel, supprime le nœud fantôme et met tous les liens qu'il avait sur ce nœud réel
 - Push une recherche de liens à ce nœud réel, selon son rôle grammatical
- S'il y a un concept, un nœud réel, pas de nœud fantôme :
 - Push un worker Activate(concept correspondant)
 - Push une recherche de liens à ce nœud réel, selon son rôle grammatical
- S'il n'y a pas de concept, un nœud réel :
 - Push une recherche de liens à ce nœud réel, selon son rôle grammatical
- S'il n'y a pas de concept, pas de nœud réel :
 - Crée un nœud réel correspondant à ce mot
 - Push une recherche de liens à ce nœud réel, selon son rôle grammatical

WriteConcept Urgence : 90

Crée un nouveau nœud fantôme dans le workspace, correspondant au concept passé en argument.

- Push un worker Activate négatif de ce concept (ie désactive le concept)
- S'il y a un nœud réel correspondant :
 - Push une recherche de liens
- Sinon, s'il y a un nœud fantôme correspondant :
 - Push une recherche de liens
- Sinon :
 - Crée un nœud fantôme correspondant à ce concept
 - Push une recherche de liens

BuildLink Urgence : 100

Construit un lien entre deux nœuds du workspace.

- Si l'un des deux nœuds passés en argument n'existe pas : erreur
- S'il existe un lien entre les deux nœuds pères :
 - Si ce lien est étiqueté par un nœud activé (seuil 50)
 - Créer un lien partant du premier nœud et allant sur le deuxième nœud, étiqueté de la même façon
- Sinon, créer le même lien sans étiquette

ReadPhrase Urgence : 100

Lit une unité complète, avec plusieurs mots éventuellement reliés entre eux (par des relations sujet-action-objet ou objet-caractéristique).

- Pour chaque mot

SearchLinkSubject Urgence : l'importance du nœud considéré (max : 100).

Cherche de nouveaux liens possibles partant d'un sujet.

—