Peuplement automatique d'ontologie à partir d'un catalogue de produits

Céline Alec, Brigitte Safar, Chantal Reynaud-Delaître, Zied Sellami, Uriel Berdugo

Peuplement automatique d'ontologie à partir d'un catalogue de produits

Céline Alec[1], Brigitte Safar[1], Chantal Reynaud-Delaître[1], Zied Sellami[2], Uriel Berdugo[2]

[1] LRI, CNRS UMR 8623, Université Paris-Sud, France
prénom.nom@lri.fr

[2] Wepingo, 6 Cour Saint Eloi, Paris, France
prénom.nom@wepingo.com

Résumé :
Nous proposons dans cet article une approche de peuplement automatisé d'une ontologie à partir de données issues de catalogues de produits. Le peuplement automatisé est vu ici comme un problème d'annotation de documents. Dans notre contexte, les documents à annoter sont des descriptions relativement pauvres ce qui rend irréalisable un peuplement totalement automatique.

Nous proposons une approche en deux étapes : (1) une étape semi-automatique d'annotation portant sur un petit ensemble de données ; (2) une étape entièrement automatique d'annotations d'autres données basées sur des mécanismes d'apprentissage automatique exploitant les résultats de la première étape. L'originalité de ce travail consiste en une approche incrémentale de raffinement des annotations qui permet de générer des annotations même dans un contexte très restreint. Le travail décrit a été appliqué sur des jeux de données réelles concernant des jouets.

Mots-clés : Peuplement d'ontologie, Annotation sémantique, Application dans le domaine du e-commerce.

1 Introduction

Ce travail a été réalisé dans le cadre d'un partenariat entre le LRI et la startup Wepingo [1], qui développe des systèmes de recommandation de produits à des internautes. Pour faciliter la conception de systèmes adaptables à différentes catégories de produits, l'idée est de s'appuyer sur des ontologies des domaines des produits recommandés et sur les instances de produits associées aux ontologies. Dans le cadre de cette collaboration, Wepingo a mis à notre disposition une ontologie de domaine composée de concepts sans instance ainsi que des catalogues de produits de fournisseurs. Notre objectif est alors de proposer une application permettant de peupler l'ontologie à partir des éléments contenus dans ces catalogues, c'est-à-dire, de mettre en relation de façon automatisée des instances de produits avec des concepts de l'ontologie en s'appuyant sur les données textuelles décrivant ces instances. Cette mise en relation sera représentée par des annotations sur les produits, puis les produits annotés seront introduits en tant qu'instances dans l'ontologie pour être accessibles au système de recommandation.

Dans la pratique, la pauvreté relative des informations sémantiques présentes dans l'ontologie, la grande hétérogénéité des descriptions des produits des catalogues et leur manque de contextualisation rend la tâche d'annotation irréalisable de façon totalement automatique.

Notre approche consiste donc à commencer par concevoir un outil logiciel qui aide un concepteur humain à établir des liens entre des catalogues de produits et l'ontologie du domaine.

1. http://www.wepingo.com/fr-fr/

Nous avons mis en œuvre dans cet outil une démarche originale de génération et d'affinement progressif des annotations qui permet de dégager de l'information même dans un contexte très restreint. Une fois un certain nombre d'instances annotées semi-automatiquement par l'intermédiaire de l'outil, un classifieur est utilisé pour identifier automatiquement les concepts associables à de nouvelles instances.

La méthode de peuplement d'ontologie que nous proposons est a priori indépendante du domaine d'étude et du type de catalogue. Elle est adaptée au peuplement d'une ontologie comportant des classifications de produits et de caractéristiques. Des expérimentations ont été faites sur des données réelles du domaine des jouets.

Après avoir exposé le cadre de ce travail (section 2), nous ferons un rappel des travaux similaires (section 3). Nous présenterons notre approche (section 4) et nous l'évaluerons (section 5). Enfin, nous conclurons et énoncerons quelques perspectives de travail (section 6).

2 Cadre de travail

2.1 L'ontologie du monde des jouets

L'application de notre approche a porté sur le domaine des jouets. Comme support au système de recommandation, Wepingo a mis en place une ontologie des jouets (figure 1), basée sur la norme ESAR définie par des psychopédagogues (Garon *et al.*, 2002).

Cette norme identifie des catégories et des caractéristiques de jouets, en deux classifications indépendantes l'une de l'autre. Les catégories de jouets font référence au type de jouet (jeu de construction, jeu de hasard, ...) et les caractéristiques aux valeurs éducatives transmises par un jeu (concentration, dextérité, ...) ou encore ses conditions d'utilisation (jeu coopératif, associatif, ...). Un exemple de catégorie est présenté tableau 1.

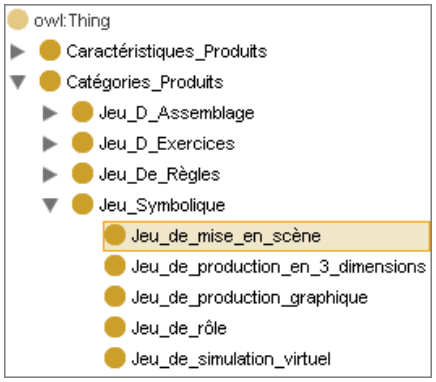


FIGURE 1 – L'ontologie ESAR

Concept $(Label)$	Jeu de mise en scène
Définition	Jeu de faire semblant dans lequel le joueur est le metteur en scène. Il réalise des scénarios élaborés dans le but de reproduire des thèmes particuliers, des scènes précises, des événements, des métiers, etc. Ces formes de jeux exigent de pouvoir mettre en scène les accessoires pertinents au contexte ou à la situation représentée.
Exemples (Ex)	playmobil, marionnette, figurine, ...

TABLEAU 1 – Le concept "Jeu de mise en scène"

L'ontologie ESAR, définie comme O_{ESAR}= (C_{ESAR}, L_{ESAR}, H_{ESAR}, Att_{ESAR}, A_{ESAR}), est limitée. C_{ESAR} est l'ensemble des concepts (33 catégories et 129 caractéristiques). L_{ESAR} est le lexique composé d'un ensemble d'entrées lexicales pour les concepts et muni d'une fonction de référence F telle que $F : 2^L \mapsto 2^C$, qui, à des ensembles d'entrées lexicales, associe des ensembles de concepts. Le lexique est composé de deux sous-ensembles de termes : $Label$ (au moins un label par concept) et Ex (exemples pour certains concepts feuilles cf tableau 1). On notera $L_{ESAR}(c)$ l'ensemble des termes de L_{ESAR} dénotant le concept c. H_{ESAR} est l'ensemble des

relations de subsomption entre les concepts. Att_{ESAR} est l'ensemble des attributs des concepts (uniquement leur définition). L'ensemble des axiomes A_{ESAR} est initialement vide. Aucune relation du domaine ne décrit les liens entre catégories et caractéristiques et ces deux classifications comportent très peu de relations de subsomption.

2.2 Les documents à annoter

Les documents (notés *Corpus*) sont des fiches décrivant un jouet par son label, sa marque, sa description (texte court non contextualisé) et sa catégorie. La catégorie ici n'est pas la même que dans l'ontologie. Elle varie beaucoup suivant le vendeur. Elle peut être très générale ("Jouet", "Jeux"), comme très spécifique ("HABA cubes et perles à assembler", "Briques"), parfois difficilement interprétable ("Bosch", "Couleurs unies"). Un exemple de descriptif de jouet est présenté figure 3a. Les formes et les contenus de ces descriptions sont très éloignés des définitions des concepts de la norme ESAR.

3 État de l'art

Annoter un document avec une ontologie consiste à rechercher dans celui-ci les fragments de texte mentionnant des concepts ou des instances de concepts appartenant à l'ontologie puis à associer ces mentions aux concepts considérés. Divers travaux d'annotation et d'extraction d'informations ont été proposés sur des domaines spécifiques. Beaucoup de ces outils, comme KIM (Popov *et al.* (2004)) ou SOFIE (Suchanek *et al.* (2009)) extraient des groupes nominaux spécifiques correspondant à des entités nommées, i.e. des noms de personnes, de lieux, d'organisations,..., repérables grâce à des grammaires formelles associées à des modèles statistiques et répertoriées dans des bases de connaissances ou des "gazeteers" (Bontcheva *et al.* (2004)).

L'identification d'instances qui ne sont pas des entités nommées est beaucoup plus délicate car aucune base ne répertorie a priori l'ensemble des instances à reconnaître et encore moins les expressions linguistiques qui leur sont associées. Ces ensembles d'instances et la terminologie propre au domaine doivent donc être recueillies pour construire la "gazeteer" adaptée à un domaine particulier. Par exemple, Amardeilh & Damljanovic (2009) prétraitent l'ensemble des termes présents dans les différentes ressources d'une ontologie (classes, instances, propriétés, valeurs de propriétés) pour en extraire un ensemble de lemmes à partir desquels est constituée la "gazeteer" associée à cette ontologie.

D'autres approches exploitent la structure du document à annoter. Par exemple, dans Amardeilh *et al.* (2005), la structure d'un document est représentée sous la forme d'un arbre conceptuel dont chaque nœud est mis en correspondance avec un concept de l'ontologie via des règles définies manuellement. De même, Aussenac-Gilles *et al.* (2013) définissent des règles d'extraction en exploitant la structure hiérarchique exprimée par les marqueurs typo-dispositionnels (police gras, italique, symbole de ponctuation ' :') au sein d'un ensemble de fiches de même format.

Les travaux cités précédemment relèvent directement du domaine de l'extraction d'informations et de l'annotation de documents. D'autres travaux, a priori plus éloignés de ces tâches, sont intéressants pour notre problématique bien qu'ils ne soient pas réalisés dans ce contexte précis. Ainsi, l'objectif de Kessler *et al.* (2012) est de vérifier l'adéquation entre des candidatures à des offres d'emploi (CV et lettres de motivation) et les offres d'emploi considérées,

c'est-à-dire évaluer la proximité entre la description d'un élément général (une offre d'emploi ou un concept d'une ontologie) et celles d'éléments plus spécifiques (des candidatures ou des instances de concept). Après avoir été soumis à différents traitements, tous les documents manipulés sont représentés par des vecteurs qui sont ensuite comparés en utilisant des combinaisons de diverses mesures de similarité (cosinus, Minkowski, ...) afin de classer les candidatures. De plus, pour être sûr de ne pas écarter trop vite une candidature, on évalue aussi sa similarité avec le vecteur représentant l'offre d'emploi enrichie des candidatures jugées pertinentes par un recruteur.

Enfin, dans Béchet *et al.* (2011), l'objectif est de peupler automatiquement une structure hiérarchique de concepts décrivant des services hôteliers, en s'appuyant sur un premier ensemble d'instances identifié par un expert. Les différents services de chaque hôtel, définis par chaque hôtelier avec son propre vocabulaire doivent être comparés aux instances initiales. Un service sera considéré comme une instance du concept correspondant à l'instance dont il est le plus proche suivant un calcul de similarité basé sur les n-grammes.

Dans notre contexte, il faut annoter des descriptions de produits comme des instances de concepts. Pour cela, il faut identifier des instances de concepts qui ne sont pas des entités nommées, au sein de documents non structurés et sans aucune homogénéité. Un certain nombre des techniques présentées précédemment sont donc complètement inadéquates. L'approche consistant à évaluer directement la proximité entre la description d'un concept et celle d'une instance est aussi inapplicable car les descriptions des concepts sont très éloignées des descriptions des produits et leur rapprochement avec des mesures de similarité ne donne aucun résultat. Bien que notre ontologie ne comporte pas initialement d'instances, l'approche proposée par Béchet *et al.* (2011) est celle qui nous est apparue comme la plus prometteuse. Pour l'identification des premières instances, nous nous sommes inspirés des travaux qui s'appuient sur des termes préalablement identifiés dans des "gazetteers" adaptées au domaine ou dans la composante terminologique d'une ontologie (Reymonet *et al.* (2007)).

4 Proposition d'une approche de peuplement d'ontologie

L'approche de peuplement de l'ontologie consiste à générer une base de connaissances $BC(O_{\text{ESAR}}, I_{\text{ESAR}}, W_{\text{ESAR}})$ à partir de l'ontologie O_{ESAR} avec $W : 2^I \mapsto 2^C$, une fonction *membre* qui, à des ensembles d'instances appartenant à I_{ESAR}, associe les ensembles de concepts de C_{ESAR} dont ils sont membres.

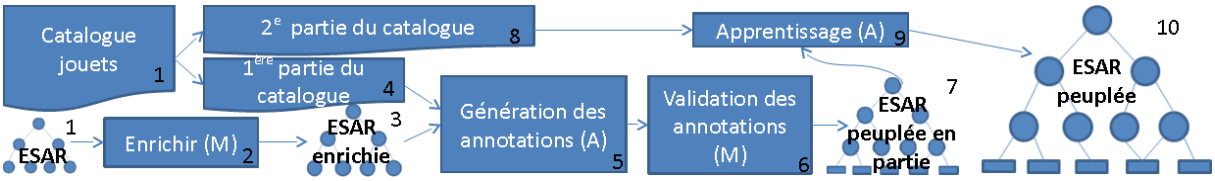


FIGURE 2 – L'approche proposée : (A) automatique, (M) manuel

Un peuplement automatique même partiel n'est possible que si l'ontologie contient les formes linguistiques associées aux concepts dont nous voulons reconnaître des instances. Notre proposition (cf figure 2) consiste donc, dans un premier temps, à enrichir (étape 1 à 3 sur la fi-

gure 2) l'ontologie de connaissances complémentaires (Section 4.1). L'ontologie enrichie est utilisée pour annoter de façon semi-automatique un échantillon de documents (étape 3 à 7 sur la figure 2). Enfin, des techniques d'apprentissage automatique exploitant ces annotations sont appliquées sur l'ensemble du corpus de documents à annoter (étape 7 à 10 sur la figure 2). L'approche d'annotation proposée comporte donc trois phases qui seront successivement décrites : la phase d'annotation d'un échantillon de documents (Section 4.2), la phase de validation des annotations de ces documents (Section 4.3), puis la phase d'annotation du corpus complet de documents (Section 4.4) basée sur l'application de techniques d'apprentissage automatique.

4.1 Enrichissement de l'ontologie ESAR

Des connaissances complémentaires nécessaires au processus d'annotation ont été ajoutées par des experts maîtrisant la norme ESAR. Ces connaissances sont de différentes natures : des termes associés aux concepts de l'ontologie (enrichissement de L_{ESAR}) et des connaissances sur ces concepts (des axiomes de A_{ESAR}).

Concernant L_{ESAR}, nous avons complété les exemples de Ex en utilisant des ressources externes. Des noms de jouets provenant d'un site internet [2] ayant utilisé la classification ESAR ainsi qu'une liste des sports provenant de Wikipedia ont été ajoutés. Nous avons par ailleurs ajouté des signes linguistiques (SL) qui sont des termes ou expressions évocateurs de concepts (par exemple "musical" et "parlant" pour le concept "Jeu sensoriel sonore") ainsi que des signes linguistiques complexes (SLcomp) de la forme "terme ET [NON] terme ET [NON] terme ..." pour aider à différencier les concepts les uns des autres. Par exemple, il existe deux types de jeux de domino : les dominos numérotés que les joueurs doivent associer (jeu d'association), et les dominos à poser debout pour construire un parcours puis à faire tomber (jeu de construction). L'utilisation de signes complexes permet de différencier ces deux jeux : le jeu de construction sera évoqué par la présence conjointe des termes "domino" et "construction" alors que le jeu d'association le sera par la présence du terme "domino" et l'absence du terme "construction". Les exemples et les signes linguistiques étant des connaissances de nature différente, nous les avons différenciés dans la représentation mais le processus d'annotation les exploite de la même façon. Après enrichissement, $L_{ESAR} = \{Label \cup Ex \cup SL \cup SLcomp\}$.

Les axiomes ajoutés dans A_{ESAR} sont exprimés sous la forme de règles propositionnelles. Il s'agit :

- d'expressions d'incompatibilités entre concepts, donnant la priorité à l'un d'eux, et de la forme "SI concept A ET concept B ALORS NON concept A" (30 règles).
- d'expressions d'inclusions de concepts de la forme "concept A IMPLIQUE concept B" (95 règles).

4.2 Annotation initiale d'un échantillon de documents représentatifs du domaine

La génération des annotations est une chaîne de traitements dont le but est de trouver un maximum d'annotations candidates exactes pour un jouet donné (catégories comme caractéristiques). Elle est composée de 3 étapes :

2. `http://www.jeuxrigole.com/liste-des-jeux.html`

1. l'établissement d'un premier ensemble d'annotations candidates qui définit le contexte d'interprétation d'un jouet ;
2. la recherche d'incohérences qui détecte au sein du contexte d'interprétation les annotations incompatibles et effectue un choix parmi elles ;
3. la complétion qui complète la liste des annotations candidates en prenant en compte des relations d'implication entre concepts.

4.2.1 Génération d'un premier ensemble d'annotations

La génération d'annotations des fiches jouets s'appuie, pour chaque concept c, sur l'ensemble $lemme(c)$ des lemmes du lexique L_{ESAR}. De même, on garde pour chaque jouet j appartenant au $Corpus$, l'ensemble $info(j)$ composé des lemmes des informations disponibles sur un jouet, i.e. son nom, sa marque, sa catégorie et sa description :

$\forall c \in C_{\text{ESAR}}, lemme(c) = lemmatisation(L_{\text{ESAR}}(c))$
$\forall j \in Corpus, info(j) = lemmatisation\{Nom(j) \cup Marque(j) \cup Cat(j) \cup Desc(j)\}$

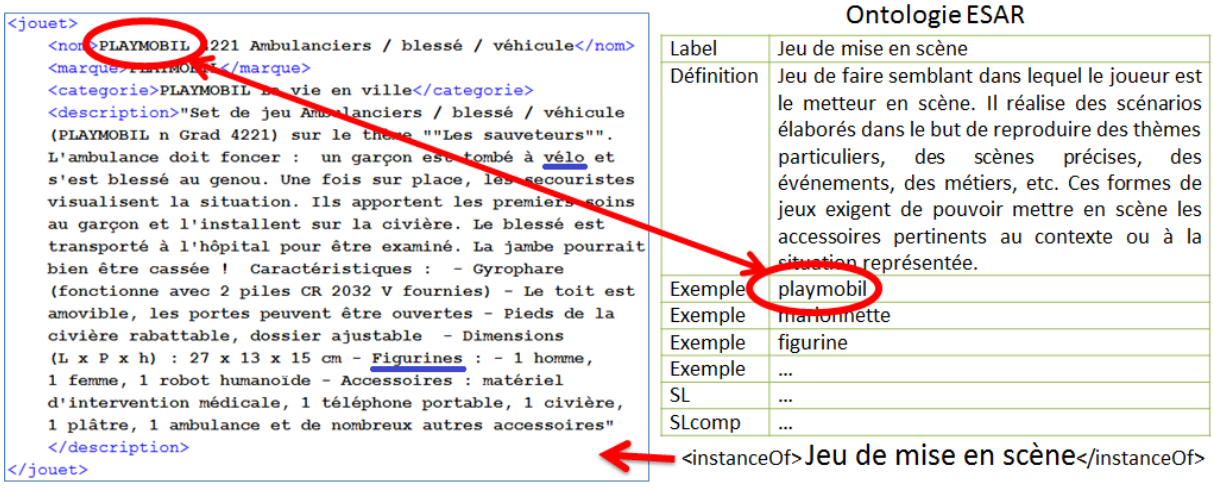
La génération des annotations est une opération de recherche d'inclusion de mots qui consiste à rechercher si un élément de $lemme(c)$ d'un concept c apparaît dans l'ensemble des informations d'un jouet j, et dans ce cas, à annoter le jouet j par le concept c (catégorie ou caractéristique) considéré :

$\forall j \in Corpus, \forall c \in C_{\text{ESAR}}$,
Si $\exists v \in lemme(c)$ tel que $v \in info(j)$ alors j `instanceOf` c.

Pour les signes linguistiques complexes, on appelle "Termes négatifs" les termes précédés de "NON" et "Termes positifs" les autres termes et on considère qu'un jouet j contient un signe linguistique complexe slc d'un concept si

$\forall\, tp \in TermesPositifs(slc), \forall\, tn \in TermesNegatifs(slc)$,
$tp \in info(j)$ et $tn \notin info(j)$

Les premières annotations produites dans cette phase définissent le contexte d'interprétation d'un jouet j, comme suit : $Ctxt(j) = \{c \mid j$ `instanceOf` $c\}$.



(a) Un exemple de descriptif de jouet (b) Le concept "Jeu de mise en scène"

FIGURE 3 – Exemple d'annotation

Par exemple, le descriptif du jouet de la figure 3 contient le terme "playmobil" qui est un "exemple" du concept "Jeu de mise en scène". Ce jouet est annoté avec le concept "Jeu de mise en scène" [3]. De même, le terme "vélo" permet de l'annoter comme "Jeu moteur" et le terme "figurines" permet de rajouter la catégorie "Jeu de mise en scène" et les caractéristiques "Créativité expressive", "Reproduction de rôles" et "Reproduction d'évènements".

Ce contexte est ensuite plus facile à analyser par les étapes suivantes que le contenu non structuré des descriptions textuelles. Pour mettre en œuvre les étapes suivantes, nous avons introduit différents ensembles de règles, chaque ensemble s'appliquant sur les résultats obtenus à la phase précédente.

4.2.2 Phase de recherche d'incohérences

La phase de recherche d'incohérences est un processus de raffinement dont le but est de détecter et d'éliminer des concepts erronés du contexte d'interprétation d'un jouet. Cette phase vise donc à améliorer la **précision** des résultats. Elle consiste à appliquer sur le contexte les règles d'incompatibilité introduites au cours de l'enrichissement. En effet, le contexte peut contenir des concepts multiples dont certains doivent être éliminés en présence d'autres. À l'issue de cette phase, on obtient un ensemble d'annotations A_1 tel que $A_1(j) \subset Ctxt(j)$.

Par exemple, le jouet de la figure 3a est annoté comme "Jeu moteur" car sa description contient le terme "vélo", alors qu'il ne s'agit pas ici d'un vrai vélo mais d'un vélo miniature associé à une figurine ("Jeu de mise en scène"). Dans ce contexte précis, l'annotation "Jeu moteur" n'est pas adaptée et il est plus facile de s'en rendre compte en la confrontant avec l'annotation "Jeu de mise en scène" également présente dans le contexte qu'en cherchant à interpréter finement la description du jouet. L'application de la règle d'incompatibilité r1 "SI Jeu de mise en scène ET Jeu moteur ALORS NON Jeu moteur" permet de supprimer l'annotation inadaptée.

4.2.3 Phase de complétion

La phase de recherche d'incohérences vise à augmenter la précision des annotations et permet à la phase de complétion de s'appuyer sur des données les plus sûres possibles. La complétion cherche à améliorer le **rappel** en exploitant toutes les inclusions entre concepts, qu'elles soient exprimées dans l'ontologie initiale ou enrichie. Elle permet d'identifier des annotations additionnelles non retrouvées lors de la phase de génération d'un premier ensemble d'annotations. À l'issue de cette phase, on obtient un ensemble d'annotations A_2 tel que $A_1(j) \subset A_2(j)$.

Par exemple, connaissant les implications "Endurance IMPLIQUE Jeu_sportif", "Jeu_sportif IMPLIQUE Jeu_moteur", un jouet annoté avec le concept "Endurance" sera par complétion également annoté par les concepts "Jeu sportif" puis "Jeu moteur".

La figure 4 montre une application des phases de recherche d'incohérences et de complétion sur l'exemple du jouet de la figure 3a. La phase de recherche d'incohérences supprime l'annota-

3. Remarquons que le fait de trouver un terme exemple d'un concept dans le nom du jouet ne suffit pas à le classer directement et définitivement comme instance du concept. Par exemple, le jouet "Playmobil pirates interactif", qui serait également annoté comme un "jeu de mise en scène", devra finalement être reconnu comme une instance de "jeu de simulation virtuelle".

tion "Jeu_moteur" en appliquant la règle r1, puis la phase de complétion ajoute les annotations "Jeu_symbolique", "Création_inventive" et "Imitation_différée".

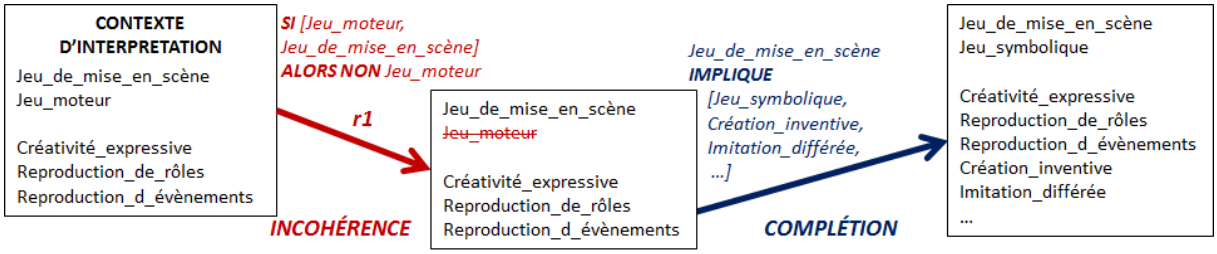


FIGURE 4 – Phases de recherche d'incohérences et de complétion sur le jouet de la fig. 3a

Ces phases s'appliquent indifféremment aux concepts catégories et caractéristiques mais dans la pratique elles ne permettent de trouver que peu d'annotations de caractéristiques car celles-ci font référence à des notions abstraites pour lesquelles les signes linguistiques sont limités. De ce fait, des étapes de raisonnement supplémentaires (cf 4.3) sont nécessaires pour déduire plus d'annotations de caractéristiques (à partir des catégories trouvées).

4.3 Validation des annotations générées

L'objectif de cette phase est de permettre à un utilisateur de confirmer ou de modifier (via une interface graphique) les annotations proposées pour un jouet et d'introduire les annotations de caractéristiques manquantes. Pour aider l'utilisateur à identifier parmi les 129 caractéristiques existantes, celles qui seront pertinentes pour le jouet considéré, le processus utilise deux heuristiques qui s'appuient sur les catégories déjà reconnues, car elles sont plus faciles à identifier.

La première heuristique consiste à identifier les caractéristiques communes aux jouets déjà traités par l'utilisateur qui sont de la même (des mêmes) catégorie(s) que le jouet à insérer. Ainsi, si un jouet est d'une catégorie A et que tous les jouets de catégorie A précédemment classés partagent un ensemble de caractéristiques E, alors l'outil propose également d'annoter ce jouet avec les caractéristiques E, en plus de celles issues du processus de génération des annotations. À l'issue de cette phase, on obtient un ensemble d'annotations A_3 tel que $A_2(j) \subset A_3(j)$. Cet ensemble A_3 est l'**ensemble des annotations proposées**.

La deuxième heuristique utilise des règles dites d'implication "potentielle" qui associent à une catégorie ses caractéristiques potentielles (i.e. qui nous paraissent être impliquées par la catégorie). Pour identifier ces règles, nous nous sommes basés sur les exemples et signes linguistiques partagés par les catégories (Cat) et les caractéristiques (Car), i.e. sur l'heuristique suivante :

$\forall\ cat_i \in Cat, \forall\ car_k \in Car$,
Si $\exists v \in lemme(cat_i)$ tel que $v \in lemme(car_k)$, alors $créer\ la\ règle : cat_i \underset{potentiellement}{\Rightarrow} car_k$.

Par exemple, "Jeu d'adresse" implique potentiellement "Coordination œil-main" car ils partagent l'exemple "toupie". 72 caractéristiques sur 129 ont été associées à au moins une catégorie et l'ensemble des règles obtenues a été ensuite complété manuellement (476 règles). Ces règles ne sont pas des règles certaines mais leur application permet d'obtenir pour un jouet j, un en-

semble supplémentaire d'annotations de caractéristiques dites annotations **suggérées**.

L'utilisateur dispose donc d'un outil muni d'une interface graphique qui, pour chaque jouet, indique des catégories et caractéristiques presque sûres (annotations proposées) et suggère des caractéristiques probables en fonction des catégories retenues (annotations suggérées). L'interface est dynamique : si l'utilisateur ajoute ou supprime des annotations, les concepts impliqués sont automatiquement ajoutés, et les suggestions de caractéristiques évoluent. Quand l'utilisateur valide, les jouets sont ajoutés dans I_{ESAR}.

4.4 Annotation du corpus complet par apprentissage basé sur l'échantillon

Après avoir décrit l'approche utilisée pour annoter un échantillon représentatif de jouets (testée sur 316 jouets), cette section présente le modèle d'apprentissage supervisé qui exploite l'échantillon pour annoter de nouveaux jouets (i.e. n'appartenant pas à l'échantillon) qui seront ajoutés à I_{ESAR}.

Nous avons utilisé le classifieur linéaire LIBLINEAR (Fan *et al.*, 2008), basé sur SVM (Cortes & Vapnik, 1995), et conseillé notamment pour la classification de documents (Hsu *et al.*, 2003). Pour chaque concept c_i, nous avons construit un classifieur SVM qui prédit pour un jouet donné si celui-ci doit être annoté par le concept c_i considéré ou pas. Nous avons donc construit 162 modèles SVM, un pour chaque concept de l'ontologie.

Pour représenter les jouets d'une manière vectorielle, nous avons testé plusieurs représentations de type sac-de-mots (Salton & McGill, 1986) : le monde est décrit avec un dictionnaire de mots et un jouet est représenté par un vecteur de la même taille que le dictionnaire de mots choisi. Chaque élément du vecteur représente un mot. Nous avons testé une représentation sac-de-mots binaire (1 pour la présence du mot dans le descriptif du jouet et 0 pour son absence) et une représentation tf-idf. Le dictionnaire utilisé est composé des lemmes des mots issus des descriptifs des jouets. Pour chaque représentation vectorielle testée, nous avons pris en compte différents sous-ensembles des attributs des jouets. Nous avons aussi appliqué une *stop-list* de mots à ne pas prendre en compte (entre autres les nombres, pronoms, prépositions, déterminants, abréviations et conjonctions) que nous appelons *stop-list* de base. Nous proposons aussi une *stop-list* plus élaborée, paramétrable par l'utilisateur, pour éventuellement ajouter d'autres catégories grammaticales à ne pas prendre en compte. Des compléments d'informations sont donnés dans la partie applicative section 5.2. La représentation vectorielle des jouets et la création des modèles SVM est entièrement automatique. Une fois les paramètres définitifs choisis, tous les jouets du catalogue sont insérés automatiquement dans I_{ESAR}.

5 Évaluation de l'approche

5.1 Évaluation du processus de génération d'annotations

Protocole expérimental. Nous ne considérons ici que les catégories de jouets car les annotations de caractéristiques sont difficiles à établir, que ce soit manuellement ou par l'outil. Pour l'évaluation, nous avons utilisé l'outil d'annotation sur un échantillon de 100 jouets construit de manière aléatoire et annotés manuellement. Les annotations proposées par l'outil ont ensuite été confrontées aux annotations manuelles.

Résultats. Le tableau 2 montre l'amélioration de la précision et du rappel apportée par les différentes étapes d'enrichissement et de raffinement. On remarque que l'amélioration la plus significative vient de l'introduction de nouveaux exemples et des signes linguistiques. Dans la confrontation des résultats, nous avons considéré comme faux un jouet annoté par plusieurs catégories dont l'une au moins était erronée. En revanche, une annotation partielle mais correcte est considérée comme juste. De ce fait, les règles de complétion ne modifient pas le résultat alors qu'en fait, elles introduisent de nombreuses annotations. Les résultats montrent que notre méthode atteint une précision satisfaisante même si le rappel est assez limité.

Étape	Précision	Rappel	F-mesure
Avant enrichissement	0,38	0,20	0,26
Exemples + signes linguistiques ajoutés	0,87	0,55	0,68
Signes linguistiques complexes	0,88	0,59	0,71
Détection incohérences (+ complétion)	0,94	0,64	0,76

TABLEAU 2 – Précision, Rappel et F-mesure du processus d'annotation

5.2 Évaluation du processus d'apprentissage automatique

Protocole expérimental. Pour évaluer la partie apprentissage de l'approche, nous nous sommes concentrés sur le concept "jeu de mise en scène". Un échantillon de jouets (316 jouets), extrait d'un catalogue particulier (Toys'R'Us) et annoté avec l'outil, constitue l'ensemble d'apprentissage. Pour l'ensemble de test, nous avons repris le même catalogue privé des jouets de l'échantillon (595 jouets) et annoté avec l'outil uniquement en terme de jeu de mise en scène ou non. Ainsi, nous construisons un modèle sur un échantillon des jouets d'un catalogue et nous observons le taux d'erreur sur les autres jouets de ce catalogue. Parmi les 36 modèles testés, nous avons opté pour celui qui génère le plus faible taux d'erreur (soit le modèle n° 12b obtenu avec les paramètres en gras italique sur le tableau 3 qui génère 2,52% d'erreur). Nous avons appliqué ces mêmes paramètres pour l'apprentissage de chacun des 162 modèles SVM créés (un par concept de l'ontologie).

Nous avons par la suite cherché à annoter (en tant que "jeu de mise en scène" ou non) avec le modèle SVM trouvé, un ensemble de 100 jouets d'un autre catalogue (Jeux et Jouets en folie) pris de manière à être le plus hétérogène possible. Soulignons que ces jouets sont très différents de ceux du premier (aucun jouet en commun). On peut donc s'attendre à ce que le modèle d'apprentissage, basé uniquement sur un ensemble représentatif du premier catalogue, ne soit pas très performant sur ces données.

Résultats. Le tableau 3 montre un extrait des pourcentages d'erreur pour le classifieur de jeux de mise en scène sur l'ensemble de test du premier catalogue. Le paramètre C du classifieur modélise le coût de violation des contraintes. Autrement dit, plus C est grand, plus on impose que les données soient sûres (non bruitées). Le descriptif correspond aux éléments considérés dans le vecteur parmi les différents attributs d'un jouet (label L, marque M, catégorie C, description D). La représentation correspond à la méthode de représentation vectorielle utilisée (binaire ou tf-idf). Les deux *stop-lists* décrites (la *stop-list* de base et celle qui supprime en plus les adverbes) ont été testées. L'ensemble d'apprentissage étant représentatif du premier catalogue tout entier, il l'est donc aussi de l'ensemble de test. Autrement dit, les jouets de l'ensemble de test sont assez proches d'au moins un jouet de l'ensemble d'apprentissage ce qui explique nos bons résultats.

Taux d'erreur					
N°	C	Descriptif	Représentation	Stop-list de base (a)	***Stop-list de base + sans adverbe (b)***
...	...	...	...	...	...
10	10	LMC	TF-IDF	6,72%	6,72%
11	10	LMCD	Binaire	3,87%	4,87%
12	***10***	***LMCD***	***TF-IDF***	3,03%	***2,52%***
13	100	LM	Binaire	9,41%	9,41%
14	100	LM	TF-IDF	9,75%	9,75%
...	...	...	...	...	...

TABLEAU 3 – Taux d'erreur d'annotation de l'ensemble de test pour les jeux de mise en scène

Le tableau 4 montre les résultats obtenus sur les 100 jouets du second catalogue avec le modèle n° 12b retenu. Parmi les 31 jouets de mise en scène, 15 ont bien été étiquetés comme tel. Aucun jouet n'a été étiqueté comme jeu de mise en scène alors qu'il ne l'était pas. On obtient donc 100% de précision et un rappel de presque 50%. Le rappel est faible car l'échantillon d'apprentissage, basé sur le premier catalogue, n'est pas représentatif des jouets du second catalogue. Cela nous semble donc très satisfaisant et nous pouvons supposer qu'en agrandissant l'échantillon d'apprentissage avec des jouets du deuxième catalogue, nous obtiendrions un meilleur rappel.

Résultats	
Taux d'erreur	16%
Précision	100%
Rappel	48,39%
F-Mesure	65,22%

TABLEAU 4 – Résultats sur 100 jouets de "Jeux et Jouets en folie"

6 Conclusion et perspectives

Dans cet article, nous avons proposé une approche originale pour associer des produits décrits dans des catalogues aux concepts d'une ontologie de domaine. Cette approche a été testé sur l'univers des jouets. Elle répond ainsi à une problématique de peuplement automatisé d'ontologie. Son originalité est d'une part la génération itérative des annotations et d'autre part la complémentarité entre les phases automatiques et semi-automatiques. Ainsi, l'approche est optimisée afin de réduire au minimum le travail de l'expert. Néanmoins le travail de celui-ci est nécessaire car la faible qualité des descriptifs de produits ne permet pas à une approche automatique d'être performante.

Les premiers résultats d'annotation des produits par leurs catégories sont prometteurs. En revanche, les caractéristiques évoquant des notions abstraites rarement utilisées dans les descriptifs, les signes linguistiques évocateurs sont plus rares et les annotations sont plus difficiles à établir.

La partie apprentissage a bien fonctionné sur les jeux de mise en scène même si ces types de jeux sont difficiles à reconnaître. Par exemple, un humain peut lire la description d'un tracteur sans comprendre s'il s'agit d'un tracteur miniature (jeu de mise en scène) ou d'un tracteur à pédales (non jeu de mise en scène). Étant donné cette difficulté pour un humain, nous estimons qu'un tel concept n'était pas simple à traiter d'une façon automatique.

L'approche exige de l'expert un travail de validation des annotations d'un échantillon représentatif de la diversité des produits. Cette tâche est manuelle et peut sembler lourde mais elle est limitée dans le temps car elle n'est à faire qu'une seule fois (modulo quelques ajustements pour prendre en compte les articles nouveaux). Le reste de l'approche est entièrement automatique.

Nous envisageons plusieurs perspectives à ce travail. Tout d'abord, trouver une solution mieux adaptée au traitement des caractéristiques. Ensuite, il faudrait utiliser une ressource externe qui permettrait d'ajouter des signes linguistiques de manière automatique et d'aider l'expert à définir les axiomes. Nous agrandirons l'échantillon afin de tenir compte des jouets de tous les catalogues. On pourrait aussi envisager d'améliorer la partie automatique en testant d'autres méthodes d'apprentissage (Bayes, Perceptron Multi-Couches, ...) et d'autres formes de représentations vectorielles (tenant compte des synonymes par exemple). Plutôt que d'utiliser un classifieur, on pourrait tester une méthode plus proche de (Kessler *et al.*, 2012), consistant à comparer un vecteur représentant plusieurs instances d'un concept donné avec un vecteur représentant un jouet à classer. Enfin, comme cette approche est indépendante du domaine et reproductible avec des connaissances adaptées, il serait intéressant de l'appliquer à d'autres domaines, tels que les cadeaux en général ou les voyages, comme souhaite le faire Wepingo.

Références

AMARDEILH F. & DAMLJANOVIC D. (2009). Du texte à la connaissance : annotation sémantique et peuplement d'ontologie appliqués à des artefacts logiciels. In *IC2009*, p. 157–168.

AMARDEILH F., LAUBLET P. & MINEL J.-L. (2005). Document annotation and ontology population from linguistic extractions. In *K-CAP '05*, p. 161–168, New York, NY, USA : ACM.

AUSSENAC-GILLES N., KAMEL M., COMPAROT C. & BUSCALDI D. (2013). Construction d'ontologies à partir de pages web structurées. In *IC2013*, p. 1–17.

BÉCHET N., AUFAURE M.-A. & LECHEVALLIER Y. (2011). Construction et peuplement de structures hiérarchiques de concepts dans le domaine du e-tourisme. In *IC2011*, p. 475–490.

BONTCHEVA K., TABLAN V., MAYNARD D. & CUNNINGHAM H. (2004). Evolving GATE to Meet New Challenges in Language Engineering. *Natural Language Engineering*, **10**(3/4), 349–373.

CORTES C. & VAPNIK V. (1995). Support-vector networks. In *Machine Learning*, p. 273–297.

FAN R.-E., CHANG K.-W., HSIEH C.-J., WANG X.-R. & LIN C.-J. (2008). LIBLINEAR : A library for large linear classification. *Journal of Machine Learning Research*, **9**, 1871–1874.

GARON D., FILION R. & CHIASSON R. (2002). *Le système ESAR : guide d'analyse, de classification et d'organisation d'une collection de jeux et jouets*. Editions ASTED.

HSU C.-W., CHANG C.-C. & LIN C.-J. (2003). *A Practical Guide to Support Vector Classification*. Rapport interne, Department of Computer Science, National Taiwan University.

KESSLER R., BÉCHET N., ROCHE M., MORENO J. M. T. & EL-BÈZE M. (2012). A hybrid approach to managing job offers and candidates. *Information Processing and Management*, **48**(6), 1124–1135.

POPOV B., KIRYAKOV A., OGNYANOFF D., MANOV D. & KIRILOV A. (2004). Kim – a semantic platform for information extraction and retrieval. *Natural Language Engineering*, **10**(3-4), 375–392.

REYMONET A., THOMAS J. & AUSSENAC-GILLES N. (2007). Modélisation de ressources terminoontologiques en owl. In *IC2007*, p. 169–181.

SALTON G. & MCGILL M. J. (1986). *Introduction to Modern Information Retrieval*. New York, NY, USA : McGraw-Hill, Inc.

SUCHANEK F. M., SOZIO M. & WEIKUM G. (2009). Sofie : a self-organizing framework for information extraction. In *WWW*, p. 631–640.