Chapitre 2 : Automatiser la création d’ontologies

Une limitation fondamentale des systèmes informatiques réside dans leur incapacité à comprendre intrinsèquement les objets qu’ils manipulent. Pour un ordinateur, une variable, qu’elle représente une molécule chimique ou le nom d’un groupe de musique, reste simplement une chaîne de caractères encodée en bits, dépourvue de toute signification contextuelle ou sémantique. Ce problème, souvent décrit comme celui de l’acquisition et de la représentation des connaissances, constitue un défi central dans la conception de systèmes « intelligents ».
Si les embeddings se sont imposés comme une solution prometteuse pour fournir une approximation du sens des objets manipulés, leur portée reste limitée. En effet, bien que ces vecteurs numériques capturent des similarités contextuelles ou conceptuelles à partir de grands corpus de données, ils ne permettent pas de représenter de manière exhaustive et universelle la complexité des relations entre tous les concepts possibles. On ne peut pas, en pratique, créer des embeddings pour tout et pour tous les usages.

Dans ce contexte, il devient crucial d’explorer d’autres approches pour construire des bases de connaissances adaptées à des domaines spécifiques. Parmi celles-ci, les **ontologies** occupent une place de choix. Ces structures permettent non seulement de formaliser les relations entre concepts et objets de manière explicite, mais aussi de capturer des règles qui favorisent le raisonnement et la généralisation. En reliant les données à des modèles conceptuels, les ontologies offrent une passerelle entre la simple manipulation de données brutes et une compréhension plus riche, ouvrant ainsi la voie à des systèmes capables de tirer des inférences et de soutenir des processus décisionnels complexes.

Construire une ontologie a longtemps été l’apanage des linguistes, dont le travail s’apparente à celui des lexicographes. Il s'agit d’organiser et structurer le lexique, de distinguer le sens spécifique du sens commun, et de déterminer les relations sémantiques entre les termes (homonymes, synonymes, hyperonymes, etc.). Ce travail de structuration permet de représenter la manière dont les concepts sont liés entre eux et forment un système cohérent.
Cependant, ce processus manuel et intensif s’est avéré limité face à l’explosion des données numériques et à la nécessité de structurer des connaissances dans des domaines de plus en plus complexes et spécialisés.Des outils techniques sont venus suppléer cette grande tentative de structuration de la réalité et de description de l’expérience, permettant de passer d’un travail artisanal à une approche systématique et reproductible.

Le projet Protégé de Stanford

L’un des outils phares dans ce domaine est Protégé, développé par l’université de Stanford. Ce logiciel open source est conçu pour aider les chercheurs et développeurs à formaliser le formalisme. Protégé intègre des standards bien définis pour formaliser les ontologies, en tenant compte des types de relations autorisées et des contraintes syntaxiques et sémantiques. Une fois construite, l’ontologie peut être exportée sous des formats standards (comme OWL ou RDF) afin d’être utilisée dans des systèmes informatiques, des bases de données ou des moteurs de recherche. Cela permet une production standardisée des ontologie.

Ce type d’outils répond à plusieurs besoins :

*** Interopérabilité** : En utilisant des standards comme OWL (Web Ontology Language), les ontologies produites sont compatibles avec une large gamme de systèmes d’information, facilitant le partage et l’intégration de données.

*** Réduction des erreurs humaines** : Grâce à des interfaces intuitives et à des mécanismes de validation intégrés, les outils comme Protégé minimisent les erreurs lors de la construction des ontologies.

*** Rapidité de production** : Le recours à des logiciels permet de créer et d'exploiter rapidement des ontologies complexes, ce qui était auparavant un processus long et fastidieux.

Les outils comme Protégé suivent toutefois souvent une approche doctrinale de ce qu'une ontologie doit être. Ils imposent des standards rigides et une méthodologie bien définie, ce qui peut être à la fois une force (interopérabilité, clarté) et une faiblesse (manque de flexibilité).

Les contraintes imposées par ces outils peuvent limiter leur capacité à représenter la complexité et les nuances de certains domaines. Par exemple en médecine, une ontologie rigide pourrait ne pas intégrer rapidement des connaissances émergentes, comme de nouvelles interactions médicamenteuses ou des maladies récemment découvertes**.**

Il ne faut pas oublier que ces logiciels d'ontologie gardent l'humain au centre du processus. Ils ne remplacent pas l'expertise ou le jugement humain mais l'épaulent dans sa tâche en automatisant les tâches répétitives et en offrant des suggestions. L'humain décide des concepts à inclure, de la hiérarchie entre eux, et des types de relations pertinents. L'utilisateur vérifie la cohérence et l'exactitude de l'ontologie générée ou mise à jour par le logiciel et les experts peuvent modifier ou compléter l'ontologie pour refléter des spécificités locales ou des exceptions non prévues par l'outil. Ces logiciels gardent « l'humain dans la boucle » et ne sont pensés que comme des outils d'accélération de contenu pas de création, ce qui ne résout qu'en partie le « goulot d'étranglement » généré par le flot des données au moment de construire une ontologie.

Toutefois, à l'ère de l'IA, de nouvelles techniques se développent et ont le potentiel d'accélérer drastiquement la production d'ontologies sans nuire à la qualité des ontologies produites. Après avoir décomposé « l'algorithme de production » d'une ontologie, on présentera certaines méthodes qui ont été proposées pour automatiser ou semi automatiser certaines de ces étapes.

1) Décomposer l'algorithme : quelles sont les étapes de pensée nécessaires pour créer un ontologie ?

a) Comment créer une ontologie à partir de rien ?

Pour rappel, lors de la création d'une ontologie, un utilisateur de Protégé doit suivre à peu de choses près le parcours du combattant suivant. L'utilisateur démarre tout d'abord un projet dans Protégé, souvent basé sur un format standard comme OWL ou RDF. Les étapes suivantes doivent ensuite être réalisées dans l'ordre :

1. **Définition des classes** : Les concepts ou entités principales de l'ontologie sont définis comme des **classes** (ex. : Personne, Organisation).
2. **Ajout des propriétés** : L'utilisateur ajoute des **propriétés** (relations ou attributs), comme des relations entre classes (*works_for*, *is_located_in*) ou des caractéristiques des entités (*âge*, *nom*).
3. **Déclaration des individus** : Les instances des classes (les **individus**) sont ajoutées pour représenter des entités spécifiques (ex. : *Frodon* est un individu de la classe *Personne*).

4. **Hiérarchisation et structuration** : L'utilisateur organise les classes en une **hiérarchie** (ex. : *Hobbit* est une sous-classe de Personne) et précise les relations entre elles.
5. **Ajout de restrictions et axiomes** : Des **règles** ou **contraintes** logiques sont spécifiées pour structurer davantage l'ontologie (ex. : *chaque Uruk Hai doit appartenir à une organisation*).
6. **Validation et test** : L'utilisateur valide l'ontologie en s'assurant qu'elle est cohérente et peut être interrogée sans conflit.
7. **Exportation** : L'ontologie finalisée est exportée dans un format standard (OWL, RDF) pour une intégration ou exploitation dans d'autres systèmes.

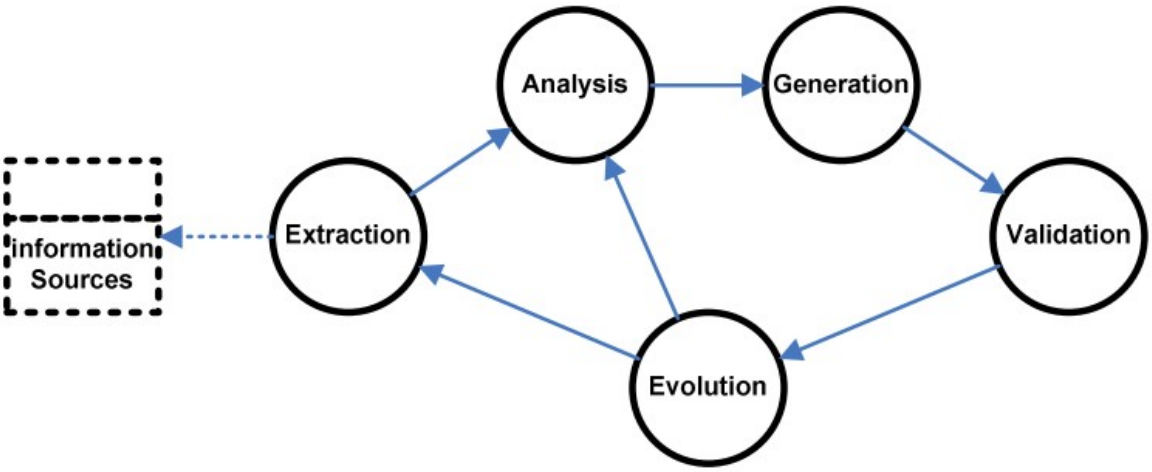
Pour parvenir à l'automatisation complète de la création d'une ontologie, il « suffit » donc d'automatiser chacune de ces étapes.
C'est cette même suite d'étape que Cimiano décrit en détail dans son livre de 2006[1] et que tous les frameworks ont suivi par la suite, rarement de façon exhaustive toutefois.

1. Term extraction methods focus on extracting linguistic realisations of domain-specific concepts.
2. The synonym extraction layer aims at acquiring semantic term variants with sense disambiguation and domain-specific synonym identification.
3. Concept extraction processes include extracting informal definitions, that is, the text description of the concept from terms and synonyms.
4. The concept hierarchisation step tries to find taxonomic relations between concepts.
5. The relation extraction processes focus on discovering non-taxonomic relations.
6. Relation hierarchisation operations intend to order relations potentially linked to each other.
7. Axiom schemata techniques identify generic rules among concepts and relations.
8. General axioms learning processes try to identify more complex relationships and connections to obtain logical implications.

La postérité a retenu ce schéma comme étant « Cimiano's ontology learning layer cake [2]».
Ce schéma est assez calqué sur les étapes que suit un humain pour construire une ontologie, mais on peut critiquer / remettre en question la division de certaines étapes du moment où la tâche est censée être automatisée. La distinction entre les trois premières étapes n'est pas extrêmement claire par exemple. De même les étapes 4 à 6 peuvent potentiellement être réalisées par le même système informatique (avec une notion de seuil / distinction entre les relations repérées). La description donne un cadre nécessaire à une littérature qui n'en avait pas, au défaut peut être de proposer un formalisme écrasant qui limite la créativité ou les écarts hors de ce sentier battu.

Un article des années 2000, collaboration d'EDF et de l'université de Versailles, publié peu avant le livre de Cimiano partait du même constat : la thésaurisation numérique de la connaissance dans un format qui soit utile aux applications informatique est bien trop coûteuse pour être largement généralisée. De ce fait, leur problématique était de se demander s'il existait (à leur époque) des outils permettant de le faire automatiquement, et si non quelles seraient les techniques existantes qui pourraient être assemblées pour ce faire.

1 https://www.amazon.fr/Ontology-Learning-Population-Text-Applications/dp/B01A652EJI
Il coûte 150€ donc je n'ai pas lu et juste fait une capture d'écran du résumé d'un autre article, désolé.
2 Bien qu'il semble que l'expression « ontology layer cake » vienne d'un article de Buitelaar et al. 2005.



Ils ont donc décrit de façon plus schématique et macroscopique le processus de création.

Ce processus (agnostique à la méthode technique utilisée pour réaliser chaque étape) se décompose en cinq étapes :
1) Extraction des composants essentiels de l'ontologie (concepts, attributs, relations et axiomes) à partir du corpus de départ. Vaste panel de techniques possibles : NLP / clustering / NL ! Analyse morpho lexicale / LLM
2) Analyse : matching entre les nouveaux éléments repérés et une ontologie préexistante.
Déterminer la structuration entre les concepts, les attributs communs…
3) génération : conversion dans le formalisme OWL / RDF (à partir de règles, partie déterministe et franchement pas intéressante)
4) Validation : souvent faite à la main mais peut être automatisée. Elle peut aussi être distribuée à la fin de chacune des étapes précédentes.
5) Evolution : permettre de mettre à jour une structure déjà créée à partir de nouveaux documents.

Au final ce pipeline mélange les fonctions de plusieurs rôles ou acteurs et la création de l'ontologie à proprement parler a lieu dans les deux premières étapes. Il nous semblera donc pertinent de nous focaliser sur elles.

b) L'automatisation de la création d'ontologies, un processus assez lent

Si des travaux techniques ont rapidement été proposés pour standardiser l'export, la fusion ou encore la mise à jour automatique d'ontologies déjà existantes, il a fallu plus de temps pour que les informaticiens ne s'attaquent directement au « goulot d'étranglement » qui est la phase d'ingestion des données et ne réfléchissent à automatiser en grande partie le processus de création d'une ontologie. Dans les années 2000, époque des réflexions théoriques importantes sur le domaine l'automatisation n'était qu'à son balbutiement.

Si l'exécution de la recette du gâteau a été longue à mettre en œuvre, très tôt les gens ont fantasmé l'idée d'ontologies rapidement constituées, ce qui s'est accompagné de la création de nouveaux termes :
* « Ontology learning »: terme datant des années 90, lorsque des premières propositions ont été faites pour accélérer la création d'ontologies en usant de diverses techniques de NLP, même si à l'origine aucune technique de *Machine learning* n'était impliquée (système à base de règles, algorithmes statistiques et *data driven*). De façon plus précise l'*ontology learning* désigne l'identification des concepts et de leurs propriétés ou relations;
* En opposition à « *Ontology population* » : remplir le schéma de l'ontologie avec des instances (objets de la réalité appartenant à une classe) et trouver des relations entre les instances.

* « *semantic annotation* » : autre expression pour désigner la phase de structuration de l'ontologie (attributs et relations)

Jusqu'à a encore récemment, l'automatisation de la création d'ontologies restait limitée à l'automatisation de certaines phases du processus, et surtout celle d'extraction des concepts, et de population, qui sont en fait les plus triviales en terme de techniques de NLP utilisées.

En 2023, des doctorants français[3] ont proposé un framework pour automatiser entièrement la création d'ontologies. "OLAF: An Ontology Learning Applied Framework" de Marion Schaeffer et Matthias Sesboüé présente **OLAF**, un cadre modulaire pour l'apprentissage automatique des ontologies à partir de textes non structurés. Conçu pour automatiser entièrement la création d'ontologies spécifiques à un domaine, OLAF se distingue par sa capacité à extraire des concepts et des relations sans intervention humaine, facilitant ainsi l'intégration dans diverses applications.

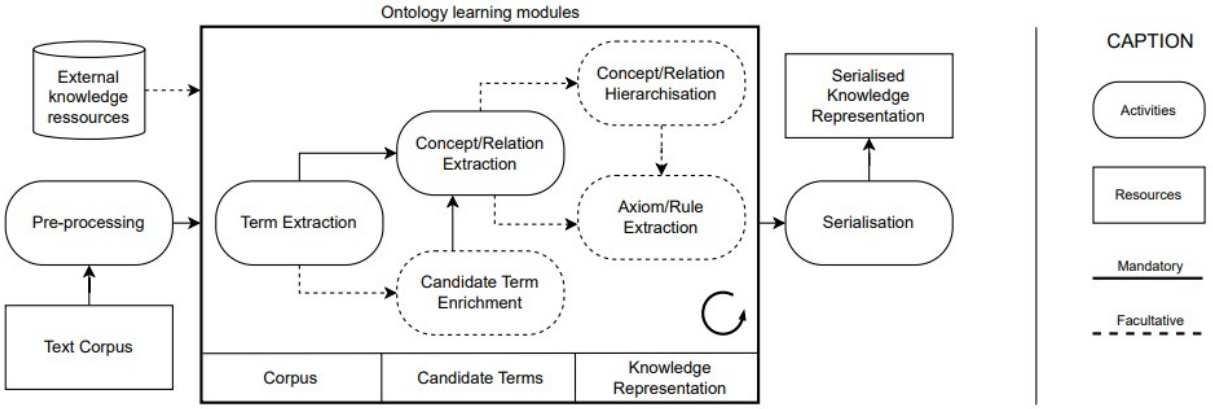


Fig. 2. Ontology Learning Process

OLAF inclut également le prétraitement et l'export des données ainsi que l'import d'ontologies déjà existantes en plus du corps principal de l'outil qui propose de créer des ontologies en procédant d'abord à l'extraction des termes et optionnellement à leur enrichissement (exports extérieurs, recherche de synonymes) pour ensuite déterminer les concepts et leurs relation, avant optionnellement de les hiérarchiser (c'est optionnel car si on a les relations on a déjà une ontologie connectée, la hiérarchisation ne fait que d'ajouter quelques relations d'inclusion / une structuration horizontale).
Chaque composant peut être implémenté par un algorithme ou une approche différente (voire parties suivantes) et l'utilisateur est libre du choix des composants qu'il fait[4]. L'outil satisfait donc la promesse de construction d'une ontologie *end-to-end* tout en laissant l'utilisateur expérimenter et paramétrer.

c) Système, pipeline ou logiciel ? Comment qualifier la bête ?

La construction automatisée d'ontologie n'est pas un système (impliquerait un unique composant). Au mieux un système agentique composé de plusieurs agents et automatisant tout le processus avec des LLMs pourrait être qualifié de système, mais la tâche de création d'une ontologie est trop complexe pour être réduite à cela.

Le **Ontology Layer Cake** de Cimiano présente la création automatique d'ontologie comme un pipeline : un processus linéaire et progressif allant de la collecte de données à la formalisation

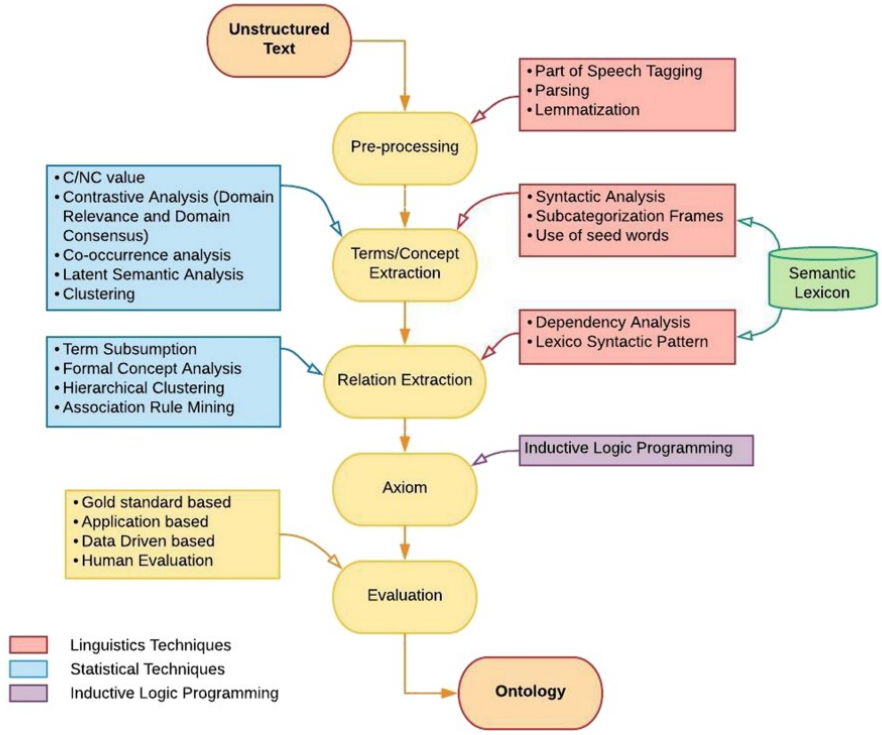
3 https://hal.science/hal-04337228v1
4 https://github.com/wikit-ai/olaf-llm-nlp4kgc2024/blob/main/scripts/pipeline_components_analysis.ipynb
La librairie est encore en développement et seule une dizaine de composants sont disponibles, mais à l'avenir, plus pourraient être intégrés.

finale. Chaque couche du "gâteau" correspond à une étape clé. L'avantage est que le processus clair et ordonné, facile à appliquer à de nouveaux jeux de données, mais il manque de flexibilité, est séquentiel, et est sensible aux erreurs propagées d'une étape à l'autre.

C'est pourquoi il est plus raisonnable ce me semble de considérer qu'un outil de création d'ontologie doit être un framework, comme OLAF, et fournir une structure générique et adaptable pour guider le processus de construction d'ontologies. Il inclut des modules spécialisés pour chaque étape clé, tels que l'extraction des concepts, l'identification des relations, ou encore l'enrichissement automatique. Un tel système est modulaire (les composants peuvent être adaptés ou remplacés en fonction du domaine), extensible (idéal pour expérimenter avec de nouvelles méthodes d'apprentissage ou de structuration, de nouveaux composants peuvent être ajoutés, des composants appelés plusieurs fois) et est conçu pour réduire la dépendance à une intervention humaine. Un framework nécessite toutefois une expertise technique pour configurer et ajuster les modules, et que l'utilisateur comprenne le rôle et le fonctionnement de chacun des composants pour les adapter au mieux à son cas d'usage.

Un logiciel comme Protégé représente une solution clé en main pour construire une ontologie et n'est pas tellement plus qu'un framework sur lequel on branche une interface graphique pour que l'utilisateur puisse l'utiliser de façon plus intuitive et observer en temps réel les résultats.

Souvent le travail universitaire portant sur l'automatisation de la création d'ontologies est soit trop suggestif (un papier de 15 pages décrivant de façon allusive tout ce framework sans vraiment expliquer comment techniquement l'état de l'art est atteint) soit trop pointilliste, focalisé sur un point du framework en n'arrivant à démontrer que rarement (et quantifier plus rarement encore) le gain de la nouvelle méthode sur l'ensemble du framework. Le champ de recherche est donc encore grand ouvert au vu du manque d'étude systématique sur le framework de création automatique d'ontologie.



La construction d'ontologies nécessite l'emploi d'une vaste gamme de techniques différentes pour automatiser chaque étape du processus[5] et nécessite de combiner une expertise en différents domaines (linguistique, domaine de spécialité, statistique…). On tâchera de présenter

5 https://pmc.ncbi.nlm.nih.gov/articles/PMC6173224/pdf/bay101.pdf

2) Méthode d’extraction de termes : keywords extraction

Dans les années 2000 beaucoup de travaux ont été réalisés sur des points pratiques, comme par exemple l’update d’une ontologie ou la fusion de deux ontologies déjà existantes portant sur le même domaine, mais au final peu avait été fait sur la création d’ontologies *from scratch*.
On ne s’intéressera pas aux cas de récupération d’une ontologie déjà existante, ni aux efforts d’une troupe de documentalistes ou d’étudiants pour créer une ontologie, mais bien à des méthodes algorithmiquement stimulantes pour les créer depuis un corpus textuel. On exclura les méthodes fondées sur l’emploi de bases de données externes (wikidata, WordNet) pour se focaliser sur les techniques basées sur la collecte (*data mining*) d’information, et ce à partir de documents non structurés (en bref, du texte, et encore du texte). La réussite de l’entreprise dépend donc de la précision de ces techniques de data mining et des outils de NLP qu’on instrumentalise à cet effet.
L’approche choisie pour construire une ontologie peut être qualifiée de *bottom up* : partir de la donnée pour faire sortir les concepts.[6]

a) Définition plus précise de la tâche

L'extraction des concepts est une étape fondamentale dans le processus de construction d'une ontologie. C'est de cette base que dépend tout le reste du travail, notamment la structuration des relations et l'organisation hiérarchique. Bien que relativement simple à mettre en place techniquement , elle reste une tâche délicate en raison de plusieurs défis : Risque de précision : inclusion de termes non pertinents qui peuvent nuire à la qualité de l'ontologie. Risque de rappel : omission de termes essentiels, ce qui peut appauvrir la structure finale.Cette étape nécessite souvent une validation humaine pour affiner les résultats et éviter ces écueils.
Le processus peut être divisé en deux sous-tâches :

* Extraction des termes : Identification des termes candidats dans un corpus, sans distinction immédiate entre les concepts structurants et les éléments de moindre importance.

* Sélection des concepts : Parmi les termes extraits, choix de ceux qui seront les concepts structurants de l’ontologie. Certains termes seront exclus, tandis que d’autres seront retenus comme instances. C’est surtout de la première étape dont on va parler.

L’extraction de termes diffère fondamentalement de la tâche plus connue d’extraction d’entités nommées (NER). Le NER repose sur une typologie (Personne, Organisation, Lieu, etc.) définie depuis 1991, qui s’applique à des entités bien circonscrites. En revanche, l’extraction de termes doit identifier une variété d’éléments allant des objets spécifiques (ex. : "Anneau de pouvoir") aux concepts abstraits (ex. : "immortalité") et catégories générales (ex. : "Elfe").Les entités nommées détectées par le NER (ex. : "Frodon", "Mordor") deviennent souvent des instances dans une ontologie, alors que les termes extraits pour une ontologie peuvent inclure des concepts ou des catégories structurantes, qui ne sont pas des entités en soi.

L’extraction de termes s’apparente davantage à la tâche d’extraction de mots-clés qui consiste à identifier les éléments saillants d’un texte, mais avec une visée plus large. Cette tâche reste importante en Recherche d’information (indexation des pages web), *data mining*, et bien sûr en NLP. De nombreux outils sont donc disponibles, et c’est une tâche qui a en général été assez vite

---

6 Latifur Khan and Feng Luo. Ontology Construction for Information Selection. In ICTAI '02: Proceedings of the 14th IEEE International Conference on Tools with Artificial Intelligence (ICTAI'02), page 122, Washington, DC, USA, 2002. IEEE Computer Society.

bien maîtrisée. Apprendre la structure (attributs, relations) est beaucoup moins trivial techniquement (cf chapitre 3 entièrement dédié à la question).

b) Méthodes old school : must know algorithmique

Les premières méthodes d'extraction utilisées étaient avant tout des méthodes statistiques. Elles restent encore pertinentes aujourd'hui en raison de leur faible coût computationnel pour des résultats plutôt satisfaisants !

* TF/IDF
TF-IDF est une mesure statistique très connue qui sert à évaluer à quel point un mot est pertinent dans un document au sein d'une collection de documents.
Il est obtenu en multipliant deux métriques : le nombre d'apparition du terme dans le document et l'inverse du nombre de documents contenant ce terme.
À l'origine inventé pour la recherche documentaire, c'est un outil très simple mais très puissant qui peut être converti pour créer des embeddings, faire des résumés de texte mais aussi pour l'extraction d'information et de mots clés.

$$TF(t,d) = \frac{number\ of\ times\ t\ appears\ in\ d}{total\ number\ of\ terms\ in\ d}$$
$$IDF(t) = log\frac{N}{1+df}$$
$$TF-IDF(t,d) = TF(t,d) * IDF(t)$$

NB : cette formule peut être ajustée en prenant en compte par exemple la longueur des documents et d'autres paramètres pour « lisser » la formule.
NB : normalement cette formule simple ne permet de repérer que les *tokens* les plus importants, mais pas les expressions contenant plusieurs mots. Heureusement des options de scikitlearn permettent de repérer des n-grams fréquents avec n paramétrable.

Néanmoins cette méthode présente plusieurs limitations :
* le résultat est entièrement tributaire de la distribution des données.
* le choix de découpage du corpus est important. 1 document = 1 phrase ? Un article ? Et si les documents sont de nature et de taille très différente ?
* le sens des mots n'est pas du tout pris en compte, la mesure est purement statistique.
* les mots rares peuvent être fortement favorisés.
* chaque terme est traité comme étant indépendant des autres or c'est faux (synonymie, associations naturelles…)

Toutefois la méthode reste d'actualité et est encore l'objet de travaux, comme par exemple l'article de Liang-Chin Chen (Sept 2024[7]) qui se propose d'optimiser l'algorithme pour extraire les mots clés d'un corpus sur l'environnement.
Les techniques utilisant les plongements lexicaux ont été développées pour pallier ces limitations.
*** Pointwise Mutual Information**

Cette mesure aide à saisir si deux mots renvoient au même concept.
Par exemple les mots Las et Vegas, Los et Angeles… appartiennent à un concept occupant plusieurs mots qu'il ne faut pas dissocier, alors que les méthodes statistiques comme TF IDF reposent sur la dissociation et le comptage des tokens.

7 https://www.sciencedirect.com/science/article/abs/pii/S0169023X24000466

L’idée clé est de calculer la fréquence de cooccurrence des termes. Si deux termes apparaissent très fréquemment ensemble, cela signifie vraisemblablement qu’ils forment un concept unique. On tente donc simplement de calculer un ratio entre le nombre de cooccurrences de deux termes et leurs apparitions totales.

$$PMI(a,b) = \log(\frac{P(a,b)}{P(\text{a})\text{P(b)}})$$

Si la présence des deux mots côte à côte est liée au hasard, alors le ratio vaut 1 et le log vaut 0.
Mais si un mot a une faible chance d’apparaître seul mais qu’il apparaît toujours accompagné de l’autre alors, il est probable que ces deux mots forment un concept unique. On peut ainsi repérer les associations les plus certaines et utiliser un seuil pour retenir les concepts importants.
À noter que la formule peut facilement être adaptée à des n_grammes de taille arbitraire.
Encore une méthode qui reste d’actualité[8].

*** C-value/NC value :**
Méthode visant spécifiquement à repérer les termes longs de plusieurs tokens. Elle combine des informations statistiques et linguistiques pour identifier les termes longs de plusieurs tokens, notamment dans le cas où ils seraient inclus (*nested term*) dans des termes plus longs. Cela peut être particulièrement utile pour les domaines médicaux et scientifiques où les termes techniques peuvent être très longs. Par exemple une « maladie auto-immune chronique » est une « maladie auto-immune » et on aimerait découvrir aussi ce concept plus simple même dans un corpus qui ne mentionnerait que des « maladies auto-immunes chroniques ».

Formule :

$$C\text{-value}(t) = \log_2(|t|) \cdot \left(f(t) - \frac{1}{P(t)} \cdot \sum_{t' \supset t} f(t') \right)$$

Définitions :

- t : un terme candidat (une séquence de mots).
- $|t|$: longueur du terme t (en nombre de mots).
- $f(t)$: fréquence d'apparition du terme t dans le corpus.
- $t' \supset t$: termes t' contenant t comme sous-terme.
- $P(t)$: nombre de termes t' qui contiennent t.

La NC Value étend la première formule avec plus d’information sur les termes voisins. Si un terme apparaît fréquemment près d’autres termes importants, alors son score en sera renforcé.

8

$$NC\text{-value}(t) = \alpha \cdot C\text{-value}(t) + (1 - \alpha) \cdot \frac{1}{N} \sum_{i=1}^{N} weight(c_i)$$

Définitions :

- α : un poids qui contrôle l'importance relative entre la C-value et le contexte ($0 \leq \alpha \leq 1$).
- N : nombre total de contextes c_i pertinents (voisins du terme t).
- $weight(c_i)$: poids du contexte c_i, mesurant son importance dans la reconnaissance de t comme un terme.

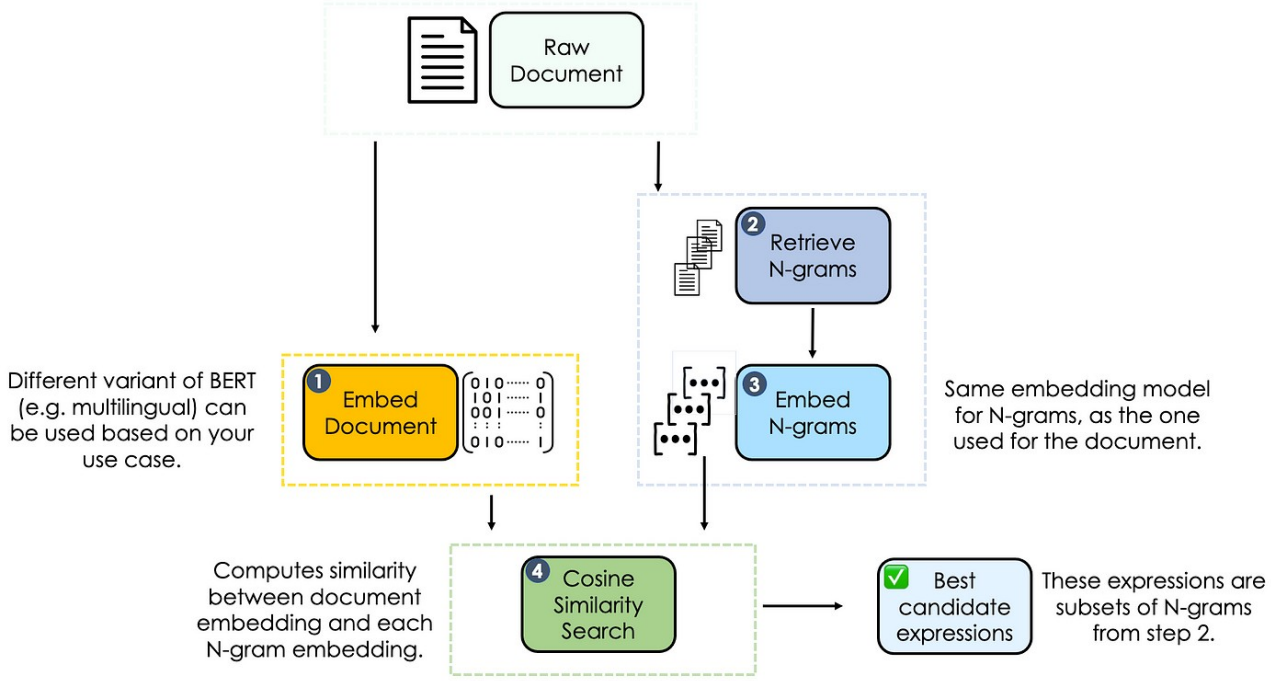
D'autres méthodes et algorithmes existent et le temps et l'espace manquent pour entrer en détail mais on peut citer :
* rake : **(Rapid Automatic Keyword Extraction)** algorithme non supervisé qui se base sur la cooccurrence des mots pour identifier et classer des termes.
* yake : pipeline un peu plus compliqué qui utilise différentes approches statistiques pour extraire les mots clés.
* Textrank : un algorithme à base de graphes qui associe à chaque terme un poids selon son nombre d'apparition et ses relations avec d'autres termes.

c) Les modèles de langue pour l'extraction de termes

Les techniques de machine learning (ML) ont considérablement amélioré les résultats de la tâche de *keyword extraction* en offrant des méthodes plus précises et adaptatives. Les méthodes supervisées nécessitent un ensemble de données annotées où les mots-clés pertinents sont déjà identifiés. Ces méthodes apprennent à distinguer les mots-clés des mots non pertinents en se basant sur diverses caractéristiques. Si on a pu au début créer des features avec par exemple la fréquence des termes**,** leur position dans la phrase, la longueur de l'expression ou ses POS, ce sont surtout les représentations fournies par les modèles de langues pré entraînés (PLM) qui sont utilisées aujourd'hui[9].

De façon non supervisée, on peut aussi utiliser des modèles de langage pré-entraînés pour obtenir des représentations contextuelles des mots et calculer la similarité entre la représentation du document et des mots ou phrases candidates pour identifier les mots-clés. Par exemple **KeyBERT** : Utilise les embeddings de BERT pour extraire des mots-clés en évaluant la similarité entre le document et les termes candidats. Cela capture le contexte sémantique des mots et permet d' identifier des expressions clés même si elles ne sont pas des n-grammes fréquents.

9 https://aclanthology.org/2023.findings-eacl.161.pdf



Ces approches nécessitent toutefois des ressources computationnelles bien plus importantes que les algorithmes présentés précédemment. Ne parlons même pas pour l'instant de l'utilisation des LLM qui reviendrait à utiliser un marteau piqueur pour planter des graines (enfin, les avis diffèrent sur la question…).

3) Le Machine Learning : révolution attendue pour structurer les ontologies ?

Face aux limitations (adaptabilité coûteuse, performances limitées) des approches symboliques, les méthodes d'IA ont été largement explorées pour qu'enfin, il y ait du machine learning dans l'ontology learning, et ce notamment dans la phase de structuration de l'ontologie qui est la plus complexe à mettre en œuvre. Étant donné que l'on traitera le repérage et la catégorie des relations dans le chapitre suivants, on se focalisera ici surtout sur l'identification parmi les termes sélectionnés de ceux qui seront centraux dans l'ontologie puis de leur hiérarchisation / structuration.

a) Les techniques pré modèles de langue pour structurer l'ontologie (« shallow »)

Plusieurs techniques sont employées pour identifier et structurer les concepts pertinents. Avant l'avènement des modèles de langues, des *features* plus simples étaient utilisés pour créer des représentation des termes extraits :

*** L'Analyse Contrastive**

L'analyse contrastive vise à distinguer les termes spécifiques à un domaine en les comparant à un corpus de référence général. Cette méthode permet de filtrer les termes non pertinents extraits lors du processus d'extraction. En comparant un corpus du *Seigneur des Anneaux* à un corpus général, des termes comme "Hobbit", "Gondor" ou "Mordor" apparaissent comme spécifiques,

tandis que des mots courants comme "arbre" ou "rivière" sont filtrés. Cette technique est cruciale lors de la distinction entre concepts et termes une étape initiale où il est essentiel d'identifier les concepts spécifiques au domaine d'intérêt. L'embedding des mots se fait sur les apparition des mots.

*** Analyse Sémantique Latente (LSA)**

La LSA est une méthode qui identifie les relations sémantiques entre les termes en analysant les cooccurrences dans un large corpus. Elle réduit la dimensionnalité des données pour révéler des structures cachées, facilitant ainsi la compréhension des relations entre les concepts. En appliquant la LSA au texte du *Seigneur des Anneaux*, on peut découvrir que les termes "Anneau", "Sauron" et "pouvoir" sont sémantiquement liés, indiquant leur association étroite dans le récit. La LSA est utile lors de l'identification des relations entre les concepts, aidant à déterminer comment les différents termes sont connectés sémantiquement, ce qui est essentiel pour structurer l'ontologie.

* Subsomption des Termes

La subsomption des termes consiste à identifier des relations hiérarchiques entre les concepts, déterminant quels termes sont des sous-classes ou des instances d'autres termes. Cette méthode aide à construire des hiérarchies conceptuelles en établissant des relations de type "est-un". Ainsi "Hobbit" peut être identifié comme une sous-classe de "Race", et "Frodon" comme une instance de "Hobbit". La subsomption est essentielle lors de la structuration hiérarchiqu**e** de l'ontologie, permettant de définir des relations de généralisation/spécialisation entre les concepts. Elle est le plus souvent réalisée par des méthodes statistiques ou symboliques.

b) La révolution des embeddings

L'arrivée des modèles de langue et leur capacité de compréhension contextuelle des mots a été pleine de potentiel pour la création d'ontologie. En effet, on peut espérer en prenant en compte le sens des mots dans les phrases, les systèmes obtiennent de meilleures capacités de généralisation que les systèmes à base de règles ou de représentation creuses. Les représentations vectorielles contextuelles des mots ou des phrases peuvent être utilisées à presque toutes les étapes de la structuration de l'ontologie, que ce soient pour des approches d'apprentissages supervisées ou non supervisées.

* L'identification des synonymes
Les vecteurs contextuels peuvent permettre d'identifier plus facilement (similarité cosinus ou norme l2 très importance) les concepts du domaine de spécialité qui sont synonymes.
On peut donc parvenir à un clustering de termes synonymes (*synsets*).
On peut aussi penser à entraîner des modèles chargés de prédire si deux termes sont synonymes ou non.

* L'extraction de concepts
Une des étapes les plus complexe de la création d'ontologie est le tri et la structuration des termes.
Ce qui implique de distinguer les instances des termes puis de structurer les termes au moyen de relations sémantiques de bases.
On peut soit traiter tous les termes extraits (hors instances) comme des concepts ou utiliser des méthodes de clustering pour identifier les concepts.

* L'extraction et l'étiquetage des relations
Voir chapitre 3 pour de plus amples détails.

Bien que ces approches soient prometteuses, elles sont toutefois plus complexes à mettre en place. Il faut notamment s'assurer que le modèle utilisé soit adapté au domaine de spécialité pour lequel on veut créer une ontologie, au risque que les termes les plus spécifiques ne soient pas connus ou que les concepts proches soient trop rapprochés dans l'espace vectoriel latent pour pouvoir être distingués de façon fiable. Cela implique donc une phase coûteuse d'apprentissage par transfert avant de pouvoir spécialiser un modèle sur une des tâches mentionnées et n'est pas forcément réalisable pour de petits corpus.

* ML et OL : une alliance contre nature ?

Dans l'article "Combining Machine Learning and Ontology: A Systematic Literature Review" de Sarah Ghidalia et al. (2024)[10], les auteurs explorent l'idée que le machine learning (ML) et les raisonnements basés sur les ontologies représentent deux facettes d'une même réalité.

Ils soulignent l'opposition entre les algorithmes de ML, qui apprennent des motifs à partir de données brutes via un raisonnement inductif, et les ontologies structurées, qui imposent des raisonnements sur les données qu'elles organisent, caractérisées par un raisonnement déductif associé aux approches symboliques de l'IA classique (Good Old-Fashioned AI).

Les auteurs illustrent cette dualité en se référant à l'allégorie de la caverne de Platon, suggérant que l'IA, en se basant uniquement sur les données fournies, n'accède qu'à une représentation partielle et schématique de la réalité, similaire aux ombres projetées sur les parois de la caverne.

Cette limitation pourrait restreindre l'impact de la technologie sur le monde réel si elle ne peut y accéder pleinement. Ainsi, les deux aspects — apprentissage automatique et raisonnement ontologique — sont jugés indispensables aux systèmes d'information moderne

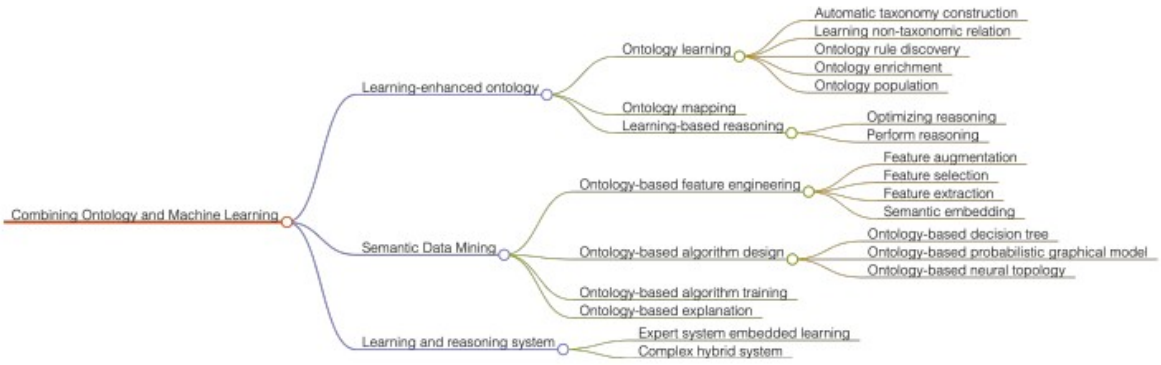


Figure 5: Overview of the combination of ontologies and machine learning techniques

Une étude globale des emplois de l'IA dans la création d'ontologies montre qu'elle est vraiment diverse en terme de rôles et de techniques utilisées :
* Ontology learning
Automatic Taxonomy Construction : parvenir à la structuration hiérarchique.
- concept extraction : LDA, tfidf, using w2vec
- matching phase ; usage de similarity scores
- concept pair extraction : graph algo + relations taxonomiques (is a / has a)
finalement usage d'algos de clustering (Hierarchical Agglomerative Clustering / Neural nets) pour ordonner le tout. A remplacé CRF / chaines de Markov...
Il ne s'agit ici que de relations hiérarchiques

10ArXiv

Finalement ce sont des « anciens » algorithmes qui ont surtout été utilisés pour faire ça.
=> Peut-on penser des algorithmes dynamiques et efficaces (qui permettraient de recalculer sur un corpus et réajuster la population et la taxonomie à moindre coût) ? Quel serait le bénéfice d'employer des modèles SoTA sur ces tâches précises ? L'état de l'art donne vraiment l'impression que le domaine a été abandonné des chercheurs les plus à jour en ML et qu'il y a un gap de plusieurs années à rattraper en terme de techniques algorithmiques. Je n'ai vu le mot « transformers » mentionné nulle part, même sur des articles de 2021.

Emplois des ontologies au ML :
* Semantic Data Mining : adapter les algorithmes de ML sur des domaines particuliers (comme la physique)pour les informer avec des connaissances du monde. « informed machine learning »
on peut se baser sur des ontologies au moment de créer des vecteurs adaptés à la tâche (feature engineering/ augmentation/ selection/ extraction), enrichir le dataset initial
Des ontologies ont été utilisées pour créer des embeddings de graphes de connaissance pour les utiliser pour faire du ML.
Les ontologies peuvent également guider le choix et la paramétrisation de certains algos (arbres de décision, modèles probabilistes voire Nnets). Ex remplacer la grid search en la pré-estimant à partir de connaissances précédentes sur le domaine. Les ontologies peuvent être utilisées pour appuyer l'explicabilité (*explainability*) de certains modèles.

La progressive intégration de l'IA dans le monde des ontologies, auparavant règne du symbolique ouvre de nouvelles possibilités dont au final assez peu ont été exploitées. Possibilité d'une hybridation des méthodes de raisonnement. En tout cas même si ce n'est pas souligné, d'un gain de temps et de précision énorme dans la construction des ontologies.

c) Les LLM : La panacée pour construire automatiquement une ontologie ?

L'apparition des LLMs a permis quand on les intègre dans des chaînes de production d'accélérer de façon remarquable le traitement de nombreuses tâches (par exemple la synthèse de documents en masse, le remplissage automatique de formulaires…). A priori, avec le bon prompt, peu de tâches sont hors de portée des LLMs. D'où un regain d'intérêt très récent pour la création d'ontologies alors que c'est un sujet très complexe que la recherche en NLP avait fini par délaisser (j'espère que les pages précédentes vous avaient convaincu que ce n'était pas si facile que ça).
Mais comme toujours, l'utilisation des LLM présente des risques : hallucinations, non alignement du comportement du LLM sur les préférences humaines… qui mettent en danger la qualité de l'ontologie produite. Puisque la création d'ontologie est un pipeline, elle est soumise à la propagation de l'erreur : toute erreur dans une des étapes initiale (oubli d'une catégorie importante, fausses relations repérées…) se retrouve voire est amplifiée dans l'ontologie finale. Avec les LLMs, la création d'ontologie se transforme potentiellement en boîte noire et il convient de se demander avec sens critique si la technologie est assez mature pour obtenir de bons résultats.
Plus d'une dizaine d'articles publiés ces six derniers mois recensent des essais de création d'ontologie par LLM voire des tentatives de synthèse. On tentera d'en dégager les points les plus importants, en s'intéressant notamment à la manière dont chaque étape de la construction d'une ontologie a été automatisée par le LLM, et comment le résultat produit a été évalué.

* Un exemple

On peut commencer par l'étude de cas d'une solution industrielle française, Lettria.
Les entreprises gardent leurs secrets de fabrication bien gardés comme de juste, et l'on va essayer de dégager de la donnée ouverte des informations sur la façon dont Lettria crée ses ontologies.
Lettria propose une solution « No Code » de gestion des connaissances.[11] L'utilisateur donne en entrée au système ses documents sous différents formats. Ces documents sont analysés et enrichis

11 https://www.lettria.com/features/lettria-knowledge-studio-ontology-and-enrichment

de sorte à repérer concepts et relations. L'utilisateur peut ensuite choisir d'ajouter manuellement (d'après ma compréhension de la page) ces concepts et relations à l'embryon d'ontologies, ou laisser l'IA inférer les classes et les relations (« By organizing data and establishing connections, they enable algorithms to reason and automate tasks, leading to smoother operations and increased efficiency. ». La nature exacte des algorithmes n'est pas précisée. Il est probable que cette page du site renvoie à l'ancienne stack de création d'ontologie antérieure à l'intégration de LLMs.

Une autre page plus récente[12] présente une étude comparative de modèles génératifs sur la création d'une ontologie dans le domaine financier. On peut lire entre les lignes quelles sont les étapes clés de la génération d'ontologies par LLM.
La lecture attentive de la page permet d'apprendre que :
* Le nombre de classes de concepts à extraire fait partie des paramètres du système à sélectionner. Ainsi comme dans un clustering avec KNN, il faut connaître en avance le nombre de classe recherchées (donc souvent des algorithmes bien plus « faciles » que ceux qui déterminent tout seul le seuil auquel s'arrêter.
* Un des aspects de l'étude porte sur le fait de savoir si préciser le domaine de l'ontologie (dire que c'est une ontologie du système financier) améliore les résultats. Il s'agit donc d'un paramètre de prompt engineering.
* L'article évoque brièvement un chaînage de prompts.
* Une notion importante mentionnée est celle de « questions de compétences » sur laquelle on reviendra plus loin.

L'étude se conclue par le fait que Claude opus semple obtenir les meilleurs résultats avec 50 classes, avec un F1 score d'environ 0.45 et que préciser le cas d'usage n'aide par forcément à obtenir de bons résultats. Plusieurs commentaires sur cette étude selon moi :
* Elle ne peut être conclusive parce que le test ne porte que sur UNE ontologie sur UN domaine et que de plus l'ontologie est de taille réduite (30 documents comme source, une 50 aine de classes). Elle a pour standard de référence une ontologie réalisée à la main mais qui n'est peut être pas la meilleure ontologie du domaine. Difficile de dire sans accès aux données.
* Un F1 score de 0.45 est dangereux ! Cela implique qu'au moins la précision ou le rappel sont sous les 0.5 donc soit l'ontologie n'est pas parvenue à capturer plus de la moitié des concepts les plus importants, soit elle contient énormément de bruit, ce qui est aussi génant.
* On s'attendrait que préciser le domaine de l'ontologie aide les LLMs à mieux cerner ses concepts principaux (sens commun) or ce ne semble pas, selon les modèles et le nombre de classes être toujours le cas. Les résultats présentent donc une variabilité importante qui est liée à la variabilité du comportement des LLMs qui n'a pas été stabilisé pour effectuer la tâche de façon reproductible.
* Aucune surprise quand l'ontologie comporte près de cinquante classe (« comprises 37 classes, 54 object properties, and 48 data properties ») que demander de trouver 50 classes au modèles donne de meilleurs résultats que 30 ou 10… il aurait été intéressant d'étudier l'évolution du rappel en allant au-delà par contre, ce qui n'a pas été fait.

En bref ce benchmark intéressant reste toutefois superficiel et ne permet pas de dégager tous les enjeux de la construction automatique d'ontologies. Il ne permet pas non plus de montrer que la technologie de Lettria est mature pour la création d'ontologies *end-to-end* sans intervention humaine.

* Esquisse du potentiel des LLMs appliqué à la génération d'ontologies

12 https://www.lettria.com/benchmarks/benchmark-ontology-toolkit-instructing-llm-for-generating-ontologies

Comme mentionné, les articles sur la question se multiplient, rendant difficile une synthèse exhaustive. En l'absence d'un survey proposant une vue d'ensemble sur l'utilisation des LLMs pour la génération d'ontologies (il est probablement en cours de rédaction quelque part), j'ai basé ma recherche sur la réputation des universités ayant publié des travaux dans ce domaine.
Un premier article de l'Université de Tsinghua[13] explore principalement les **limitations des approches basées sur le deep learning** et les bénéfices potentiels de l'utilisation des LLMs, sans toutefois fournir de description détaillée des travaux existants. Malgré cela, les réflexions proposées sont tout à fait pertinentes.

Il ressort des trois dernières pages de l'étude que la **capacité de classification en one-shot** des LLMs est l'une des fonctionnalités les plus immédiatement exploitables pour créer des ontologies. Avec un prompt adapté, un LLM peut être utilisé pour classifier un terme ou une relation.

En outre, les capacités des LLMs en matière d'**extraction d'information** sont également intéressantes, bien qu'elles souffrent de problèmes de rappel lorsque le texte en entrée est long ou complexe. Leur **capacité de raisonnement**, en revanche, s'avère particulièrement utile pour identifier des **relations taxonomiques**. L'usage de techniques telles que le **Chain of Thought prompting** est prometteur pour structurer les ontologies, pouvant être adapté en **Chain of Layer** pour identifier les différentes couches d'une ontologie.

Les LLMs peuvent être utilisés selon deux approches principales :

- **Approche top-down** : Les LLMs servent à définir les **super-classes** et les **relations taxonomiques**.
- **Approche bottom-up** : Ils sont employés pour analyser les données unitaires et prédire la classe supérieure correspondante.

Bien que l'emploi des LLMs ouvre de nombreuses perspectives, aucun consensus résolutoire n'a encore émergé. Certaines directions clés se dégagent néanmoins :

1. Le développement de **benchmarks fiables** pour comparer les résultats des outils de génération d'ontologies à chaque phase du processus.
2. Une **définition claire et harmonisée des relations non taxonomiques**, qui restent mal définies dans de nombreux travaux.
3. Une réflexion sur la **place de l'humain**, en particulier celle des experts du domaine, dans la génération automatisée d'ontologies.

La plupart des approches ont exploité les LLMs pour la création automatique d'ontologies en les utilisant pour remplacer une des étapes classique de la création d'ontologies[14]. Toutefois certaines tentatives de prise de recul valent la peine d'être mentionnées.

* Une proposition d'apprentissage d'ontologie End to End par LLM.[15]

Des universitaire britanniques dans un article paru en Octobre proposent une solution permettant de générer automatiquement une ontologie de A à Z. La lecture de leur travail est l'occasion de s'interroger comment un LLM a-t-il pu être adapté pour réaliser chaque étape de l'« ontology layer cake ». L'intêret de leur approche est d'être agnostique au domaine sur lequel on construit l'ontologie, et de penser le processus de création d'ontologie comme un processus global et non pas se focaliser uniquement sur une de ses étapes.

13 Zhang, H., & Li, S. (2023). *"Applications of Large Language Models for Knowledge Representation and Reasoning."* Proceedings of the Tsinghua Symposium on AI and Knowledge Engineering.
14 https://arxiv.org/pdf/2307.16648
15 https://arxiv.org/abs/2410.23584

Une idée clé est que l'on peut obtenir de meilleurs résultats en réalisant plusieurs des tâches à la fois car leur résultat est dépendant.
Un LLM a été fine tuné pour apprendre à générer ce que les auteurs appellent un « ontology sub-component » et ces composants sont ensuite fusionnés pour créer l'ontologie totale. L'idée est de parser le texte pour produire des sous graphes (quelques nœuds et relations) puis de recomposer le tout en apprenant au modèle à distinguer les bonnes relations à retenir. L'intérêt est de voir la création d'une ontologie comme une tâche unique. Toutefois l'outil ne se concentre que sut un nombre très limité de relations taxonomiques (is-a, is-subclass-of, is-part-of) limitant l'adaptation directe à des domaines présentant des types de relations spécifiques.

* La détermination sans expert métier des classes et relations structurantes.

Une des étapes cruciales de la création d'ontologies consiste à distinguer les relations et concepts structurants, ce qui est souvent la tâche principale de l'expert métier. Or une utilisation des capacités de raisonnements et de compréhension contextuelle des LLMs pourrait permettre de l'automatiser en passant par une astuce : l'usage de questions de compétences[16] qui doivent permettre l'inférence des classes importantes à partir des documents.
L'dée est de poser des « competency questions » qui sont des questions liées au métier/ domaine de spécialité et auxquelles on aimerait que l'ontologie aide à répondre facilement.
Use Competency Questions*: Competency questions (e.g., "What treatment did this patient receive?") help validate that the ontology can answer relevant domain-specific queries. If your ontology cannot answer the essential questions, it may need further refinement.*[17]

On peut le décliner de deux manières : top down : user sa connaissance du domaine pour produire une structuration de la connaissance. Utiliser un chaînage de prompt à partir de peu de documents pour inférer les classes et relations taxonomiques. mais demande de la planification et risque d'avoir un rappel bas / ne pas correspondre à l'ensemble du corpus.
Bottom up : découvrir les concepts, les regrouper et inférer les classes en parsant les textes.
Data driven, évolutif Mais peut potentiellement manquer de structure, être trop détaillé.

* L'utilisation des capacités de raisonnement des LLMs pour « merge » des catégories et des concepts.

On regroupe les concepts / relations par similarité puis on demande au LLM de distinguer leur organisation. Idem pour les relation. Nécessite de la plannificaiton.

Si l'utilisation des LLMs a permis d'obtenir des résultats SoTA sur les tâches unitaires de la création d'ontologies, peu d'études étudient vraiment son coût et sa scalabilité.
Par exemple si on voulait utiliser des LLMs pour l'extraction de relations sur un corpus de quelques milliers de documents, en envoyant un prompt par phrase :
Si on considère que il y a 100 000 phrases dans le corpus (20 phrases par texte, 5 000 docs), et qu'on envoie un prompt par phrase.
Supposons qu'il y ait environ 100 tokens par prompt et que le modèle en génère 50 en sortie : 100 000 phrases × 100 tokens = 10 000 000 tokens.
Tokens en entrée : 100 tokens × 100 000 phrases = 10 000 000 tokens.
Tokens en sortie : 50 tokens × 100 000 phrases = 5 000 000 tokens.
1 000 000 tokens ÷ 1 000 × $0.0015 = $15.0.
5 000 000 tokens ÷ 1 000 × $0.002 = $10.00.
Coût total : = $25.
Sachant que 5 000 documents est un « petit » corpus. Pour 100 000 documents ce serait 500$ avec gpt 3.5. et avec gpt4, cela serait environ 25 fois plus cher…

16 https://arxiv.org/abs/2412.13688
17 https://www.lettria.com/blogpost/5-tools-to-create-your-ontologies

Conclusion : Automatiser de A à Z : une bonne idée ?

L’automatisation complète de la création d’ontologies, bien qu’attrayante à première vue, n’est pas sans poser des défis. Si le principe des pipelines et frameworks, tels que Protégé, Text2Onto, ou OntoLearn, a permis de rationaliser les processus, il n’existe pas de solution universelle pour produire une ontologie parfaite. Ces outils partagent une structure commune, découpant les tâches en étapes bien définies tout en faisant des choix d’algorithmes différents selon le cas d’usage ou les caractéristiques du corpus.

Le rôle de l’humain et le degré d’automatisation

L’automatisation totale se heurte à un compromis incontournable entre la qualité de l’ontologie produite et le degré de supervision humaine nécessaire. Le rôle de l’humain reste central pour les décisions critiques, telles que le classement des termes problématiques ou la désambiguïsation d’entités complexes. Loin de chercher à exclure l’humain, l’objectif des systèmes automatisés est de concentrer son intervention sur ces étapes essentielles, tout en accélérant les parties répétitives ou systématiques du processus.

Avantages espérés de l’automatisation :

1. Amélioration de la couverture : Réduire la dépendance aux experts pour ne pas se limiter à une partie des termes d’un domaine, mais viser une exhaustivité accrue.
2. Adaptabilité accrue : Faciliter les mises à jour automatiques lors de l’apparition de nouveaux termes ou de l’évolution de concepts existants.
3. Redéfinition du rôle humain : Passer du concepteur manuel à un rôle de juge, permettant une évaluation plus efficace et moins coûteuse cognitivement du travail réalisé par les systèmes automatisés.

Ontologies à grande échelle : une problématique encore ouverte

Une autre limite réside dans la capacité à produire des ontologies à grande échelle. La majorité des études actuelles se concentrent sur des ontologies relativement réduites (par exemple, 100 concepts et 500 relations dans le cas d’OLAF), ce qui limite leur pertinence pour des domaines complexes tels que la médecine, où les besoins atteignent des milliers de concepts et de relations. De plus, les spécificités des domaines, comme celui médical, rendent difficile l’extension directe des méthodes employées à d’autres secteurs.

La synergie entre IA et ontologies

À l’ère de l’IA, l’automatisation partielle des processus ontologiques ouvre des perspectives prometteuses. L’objectif est de tirer parti des capacités de l’IA pour réduire le coût et le temps de création, tout en maintenant la qualité grâce à l’intervention humaine dans les étapes clés. Le trade-off entre la taille des ontologies et la structuration des relations reste une problématique centrale, où l’IA peut jouer un rôle crucial pour rendre le travail plus évolutif et adaptable.

Cependant, les limites de l’automatisation complète et la nécessité de combiner le raisonnement symbolique (ontologies) avec l’apprentissage statistique (IA) soulignent l’importance de cette synergie. Plutôt que de viser une automatisation totale, il s’agit de concevoir des systèmes où humains et machines collaborent efficacement, chacun apportant sa spécificité pour atteindre une couverture et une pertinence optimales.