Construction d'ontologies à partir de textes : démonstration d'une approche basée sur l'analyse de graphes AMR

Aurélien Lamercerie*, David Rouquet*

* Tétras-Libre, 8 Rue Mayencin, 38410 St Martin d'Hères
aurelien.lamercerie@tetras-libre.fr, david.rouquet@tetras-libre.fr,
https://www.tetras-libre.fr

Résumé. Cet article présente le prolongement de travaux autour d'un outil permettant la construction automatique d'ontologies OWL à partir de textes exprimés en langue naturelle. La chaîne de traitement mise en oeuvre part d'énoncés non contraints pour aboutir à une structure logique encodant la connaissance extraite. Elle passe par deux phases majeures : (1) une sérialisation RDF de graphes AMR exploités comme représentation pivot des énoncés, et (2) l'extraction du contenu logique par l'analyse des représentations intermédiaires. L'implémentation est basée sur les standards du Web sémantique.

1 Introduction

Nos travaux portent sur la construction automatique d'ontologies à partir de textes en langue naturelle. Ils s'inscrivent dans la continuité du projet UNseL [1] dont l'objectif était la vérification automatisée d'exigences système, à partir d'inférences sur une ontologie extraite automatiquement. Les premiers résultats ont été présentés lors de la conférence EGC 2022 (Rouquet et al. (2022)). Cet article montre le prolongement de ces travaux avec une nouvelle démonstration traitant d'articles encyclopédiques. Initialement basé sur une sérialisation RDF de structures UNL [2], notre prototype exploite dorénavant et également des graphes AMR [3].

Les ontologies sont des modèles de données décrivant un domaine, sous la forme d'ensembles structurés de concepts et de relations. Elles sont utiles pour de nombreuses applications, comme par exemple les systèmes de questions/réponses ou d'aide à la décision. Leur construction est un enjeu important. Ainsi, l'état de l'art [4] présente plusieurs travaux intéressants s'appuyant sur des techniques variées. Quelques méthodes de pointe semblent déjà suffisantes pour certaines applications pratiques, telles que la classification de documents ou la recherche d'informations. Néanmoins, celles-ci visent à générer des ontologies faiblement contraintes au niveau logique, alors que de nombreuses applications basées sur du raisonnement reposent sur des axiomatisations plus complexes. Il n'existe pas de méthode éprouvée pouvant servir de référence dans ce domaine.

1. *Universal Networking system engineering Language*
2. *Universal Networking Language*, Uchida et al. (1996)
3. *Abstract Meaning Representation*, Banarescu et al. (2013)
4. Voir, par exemple, Khadir et al. (2021).

Il y a donc un intérêt à étudier de nouvelles approches pour la construction automatique d'ontologies à partir de textes. Dans cette optique, nous proposons une chaîne de traitement globale partant d'énoncés exprimés en langue naturelle. Dans un premier temps, les énoncés sont convertis dans une représentation sémantique (AMR ou UNL), puis sérialisés au format RDF. Ces représentations sont ensuite analysées pour en extraire le contenu logique et, finalement, construire une ontologie OWL représentative du document traité. Ce processus ne se limite pas au peuplement d'une ontologie préexistante. Il permet, en premier lieu, de formaliser ce qui est décrit dans un texte, en construisant une hiérarchie des concepts mobilisés, des relations qu'ils entretiennent et des axiomes qui les gouvernent.

La suite détaille notre proposition. La section 2 décrit les structures pivot utilisées, l'accent étant mis sur les graphes AMR. Le processus d'extraction est ensuite présentée dans la section 3, et quelques perspectives sont avancées dans la dernière section. En complément, les outils et ressources peuvent être consultés sur notre dépôt Git [5], accessible en Open Source.

2 Structure pivot AMR-RDF

Les représentations sémantiques définissent des structures semi-formelles qui reflètent le sens d'un énoncé tel qu'il est compris par un locuteur d'une langue. Ces formalismes, dont le développement répond à des objectifs pratiques variés, mettent l'accent sur la représentation des informations sémantiques, telles que le sens des mots, les rôles sémantiques ou la relations entre les entités. L'état de l'art comprend de nombreuses propositions qui divergent sur plusieurs aspects [6]. Notre outil permet d'exploiter deux types de représentations sémantiques : UNL et AMR. Les graphes UNL définissent le sens d'un énoncé, initialement exprimé en langue naturelle (par exemple, le français), sous la forme d'une structure sémantique abstraite d'un énoncé anglais équivalent. Cet article met l'accent sur le second formalisme.

Les représentations AMR définissent des structures sémantiques simples qui permettent de traduire la signification de toute phrase anglaise sous la forme d'un graphe orienté et étiqueté. Un exemple est donné par la figure 1. L'un des nœuds est désigné comme racine du graphe (*s/system*). Chaque nœud est associé à un concept porté par un mot anglais, par une proposition issue de la PropBank [7] ou par un mot-clé spécifique. Les mots-clés permettent d'expliciter certains phénomènes linguistiques, comme la conjonction (*a/and*) ou le nommage d'une entité (*n/name*). Les relations sémantiques sont spécifiées par les arcs, en suivant les conventions de la PropBank (notamment pour relier les arguments aux propositions). Des mots-clés spécifiques sont également utilisés pour certaines relations en complément (*:mod*, *:consist-of*).

Le développement de ce formalisme est inspiré des banques d'arbres syntaxiques [8]. Une ressource d'annotations sémantiques associant des structures AMR à des phrases anglaises a été constituée. Elle atteint aujourd'hui une taille suffisante pour permettre l'entraînement d'analyseurs sémantiques. Les techniques reposent, par exemple, sur le calcul d'alignements entre mots et concepts (Flanigan et al. (2014) ; Liu et al. (2018)), sur un processus transformant des structures de dépendance (Wang et al. (2015)) ou sur un algorithme de prédiction reliant des séquences à des graphes (Zhang et al. (2019)).

5. `https://gitlab.tetras-libre.fr/tetras-mars`
6. Voir, par exemple, Abend et Rappoport (2017).
7. *The Proposition Bank*, Palmer et al. (2005)
8. Par exemple, la Penn TreeBank (Marcus et al. (1993))

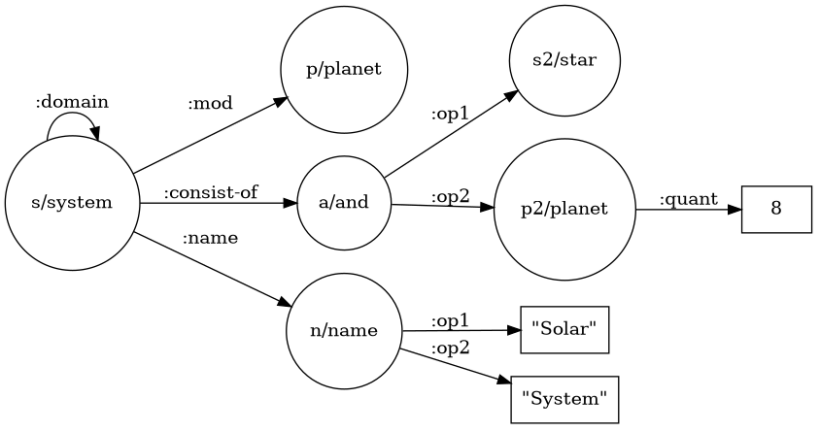

FIG. 1 – *Graphe AMR de la phrase "The Solar System is a planetary system consisting of a star and eight planets."*

Pour la mise au point de notre prototype, nous nous sommes appuyé sur la librairie Python AMRLib [9] pour créer des graphes AMR à partir de phrases anglaises. En complément, les structures obtenues ont été sérialisées en RDF à l'aide de l'outil AMR-LD [10].

3 Analyse de graphes AMR par transduction sémantique

La phase d'extraction permet le passage d'un ensemble de représentations sémantiques (graphes AMR) à une représentation logique formelle (ontologie OWL du document). Notre prototype, *TENET*, implémente un procédé basé sur une méthode de transduction sémantique compositionnelle (Lamercerie (2021)). Son développement s'appuie sur les standards du Web Sémantique du W3C (RDF, OWL, SPARQL). Il requiert, en entrée, un ensemble de structures pivot représentant le document à traiter, et donne en sortie un ensemble de triplets RDF-OWL formant une ontologie, composée de classes, de propriétés, d'instances et de relations logiques entre ces éléments.

La figure 2 montre un extrait d'une ontologie construite automatiquement à partir d'une dizaine de phrases issues de l'article anglais de Wikipedia sur le système solaire [11]. Les phrases ont été traitées telles qu'elles se présentaient, sans simplification *ad-hoc*. L'énoncé "the two largest planets, Jupiter and Saturn, are gas giants, being composed mainly of hydrogen and helium" [12] est un exemple tiré du corpus, qui est consultable sur un dépôt Git dédié [13].

9. `https://github.com/bjascob/amrlib`
10. *AMRs as Linked Data*, Burns et al. (2016)
11. `https://en.wikipedia.org/wiki/Solar_System`
12. "les deux plus grandes planètes, Jupiter et Saturne, sont des géantes gazeuses, principalement composées d'hydrogène et d'hélium"
13. `https://gitlab.tetras-libre.fr/tetras-mars/corpus/solar-system-corpus`

L'ontologie produite exhibe plusieurs concepts mentionnés dans le corpus, structurés sous la forme d'une taxonomie et caractérisés avec des propriétés logiques. Ainsi, nous observons sur cet exemple (figure 2) que la classe des *géantes gazeuses* est définie comme une sous-classe de *planètes*. Elle est associée à une restriction spécifiant que toutes les instances de *géantes gazeuses* ont comme caractéristique un élément de type *gaz* (propriété *hasFeature value gas*). Notons que l'énoncé ne nous donnait pas d'information supplémentaire : on ne sait pas ce qu'est le gaz, ni de quelle nature est le lien entre les *géantes gazeuses* et le *gaz*. Deux instances ont également été extraites et classées comme *géantes gazeuses* : Jupiter et Saturne. De plus, nous observons que les *géantes gazeuses* peuvent être composées d'hélium et d'hydrogène. Dans ce cas on peut être plus spécifique que *hasFeature* et générer une propriété *composedOf*. Ces résultats sont obtenus par la seule analyse automatique du corpus, sans connaissance préalable des classes et propriétés à extraire.

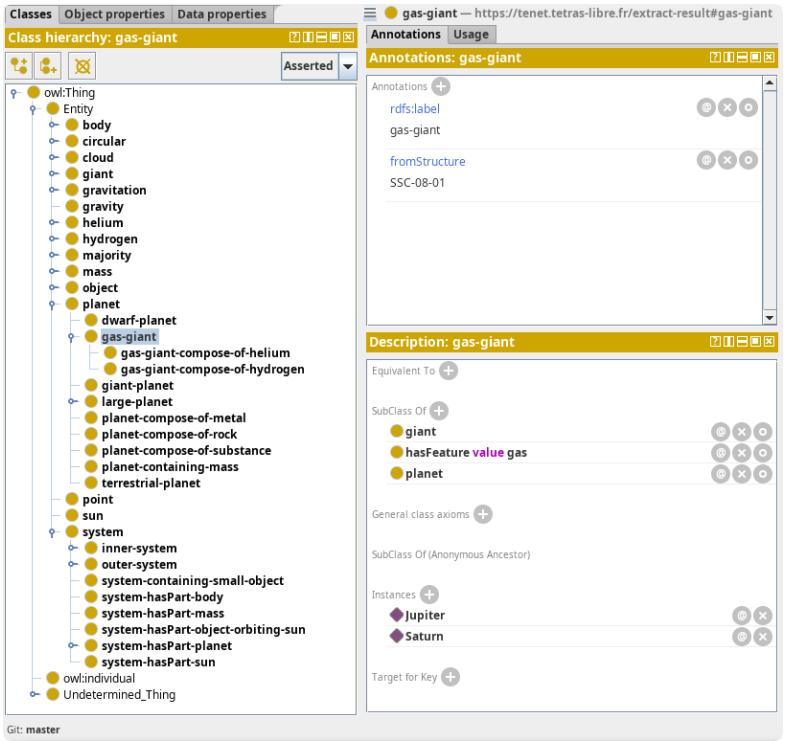

FIG. 2 – *Extrait d'une ontologie construite à partir d'un corpus décrivant le système solaire.*

L'analyse des structures AMR est réalisée avec des règles d'extraction implémentées sous la forme de requêtes SPARQL-construct. Le processus consiste en une suite d'opérations sur une structure AMR-RDF qui l'enrichissent de nouvelles données dérivées de l'interprétation du graphe. La notion de filet sémantique a été introduite pour caractériser ces enrichissements : un *filet sémantique* est un objet construit sur un graphe sémantique de façon inductive, à partir d'une base formée de filets *atomiques* correspondant aux nœuds du graphe. De plus, les filets

sont associés à plusieurs données exploitables durant le traitement. La figure 3 donne l'intuition du traitement en montrant l'émergence d'un filet construit autour de plusieurs noeuds. Ces enrichissements successifs permettent l'*activation* de nouvelles règles et la génération de nouveaux filets, avec un processus de nature dynamique (l'objet traité évolue pendant le traitement) et compositionnel (les filets sont obtenus par composition de plusieurs filets, avec un calcul itératif).

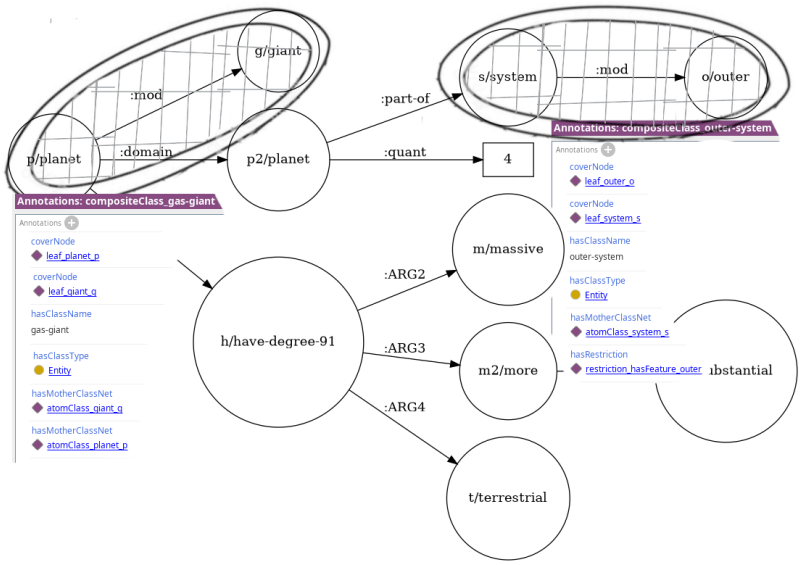

FIG. 3 – *Illustration du processus d'extraction par calcul de filets sémantiques : cet exemple montre deux filets de type "compositeClass". Ces deux filets mettent en évidence l'existence d'entités abstraites avec quelques caractéristiques. La suite du traitement permettra la génération effective des classes correspondantes, mais aussi le calcul de relations entre ces classes et d'autres éléments extraits. Par exemple, l'analyse de la relation " :part-of" entre le filet de droite et un filet construit autour du noeud "p2/planet" permet de déduire l'appartenance d'instances de "planètes" au "système externe".*

D'un point de vue technique, plusieurs schémas RDF ont été définis pour structurer les objets traités (*amr-rdf*, *base-ontology*, *semantic-net*). Les règles à appliquer sont organisées dans un schéma d'extraction. Une *règle d'extraction* est une requête SPARQL-construct, constituée d'un ensemble de contraintes, permettant de sélectionner des ressources (filets et données) vérifiant certaines conditions, et d'un constructeur, permettant de produire de nouvelles ressources (données, filets et éléments de l'ontologie). La figure 4 montre une règle permettant de générer les filets de la figure 3.

Ces règles dépendent fortement de la structure des graphes sémantiques en entrée, c'est à dire du formalisme AMR et des phénomènes linguistiques. Les propriétés sont déduites des rôles fondamentaux définis dans la *PropBank*. Nos développements couvrent plusieurs phénomènes linguistiques, comme les conjonctions (mis en évidence, au niveau du graphe

AMR, par les mots-clés *and*, *or*), la dénomination d'entités (mot-clé *name*), la comparaison (mot-clé *have-degree*) ou certains prédicats (mots-clés *mod*, *domain*). Toutes les requêtes utilisées sont accessibles dans le dépôt Git ((`https://gitlab.tetras-libre.fr/tetras-mars/tenet`). L'efficacité et la terminaison du traitement, dont la complexité est linéaire par rapport au nombre de phrases, sont assurées et ajustées en s'appuyant sur le typage des filets et l'optimisation des requêtes.

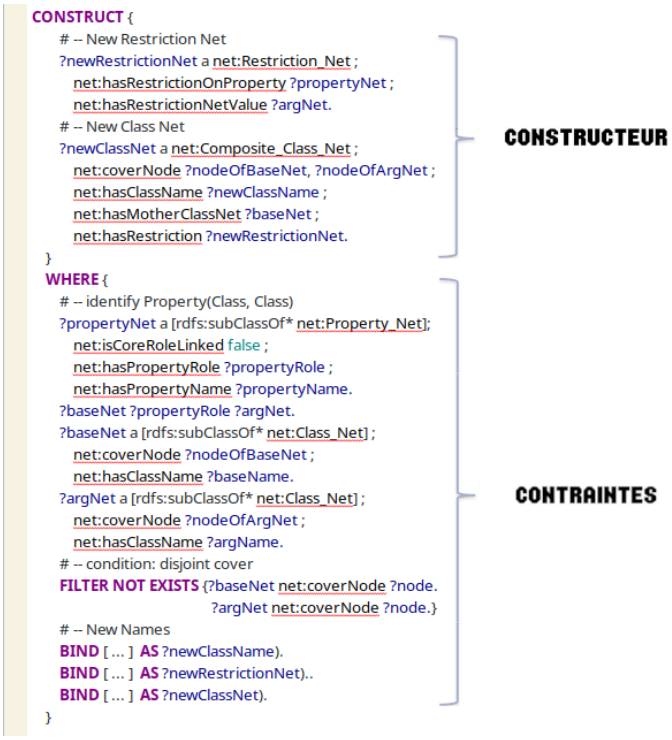

FIG. 4 – *Règle de transduction permettant la création d'un filet de type "compositeClass" à partir de deux filets de types "class" respectant des contraintes précises (relation identifiée entre les deux filets et couverture disjointe).*

Le schéma qui organise les règles est structuré sur plusieurs niveaux. Les principales étapes sont : (1) l'extraction des éléments atomiques (classes, propriétés, instances), (2) la mise en évidence de phénomènes sémantiques, (3) la formation d'éléments composites par un procédé récursif, (4) l'extraction des propriétés et relations pour les éléments atomiques et composites, (5) la classification des ressources extraites et (6) la construction de l'ontologie cible.

4 Perspectives

L'objectif de nos travaux est de maîtriser la construction automatique d'ontologies OWL à partir de textes non contraints. Notre prototype permet de passer d'un ensembles de représentations sémantiques "linguistiques", rattachées à des énoncés, à une structure logique formelle décrivant les connaissances portées par ces structures. Par sa nature, il a vocation à s'intégrer dans une chaine de traitement plus large conçu pour répondre à un problème ou un besoin précis. Nous avons identifié plusieurs cas d'usage, tels que le traitement de documents juridiques, la formalisation de documents de maintenance ou l'ingénierie des exigences.

La construction d'ontologie implique différentes tâches complémentaires, de l'extraction de la terminologie jusqu'à la mise en évidence de propriétés complexes entre des concepts hiérarchisés. Elle se distingue du peuplement d'ontologies, qui vise à extraire des informations linguistiques permettant d'identifier des instances de concepts pour enrichir une ontologie déjà définie. S'il existe beaucoup de travaux sur le peuplement d'ontologies, auquel se rattache des problèmes de reconnaissances d'entités nommés ou de classification, la construction automatique d'ontologies *à partir de zéro* présente également un intérêt certain, tant ces ressources peuvent s'avérer critiques et leurs conceptions laborieuses. Cette démarche nécessite plusieurs opérations réalisées par le prototype actuel, dont les performances précises restent à évaluer.

Cette question de l'évaluation des ontologies produites représente un enjeu important. En pratique, elle est très inégale dans les travaux publiés. L'une des principales difficultés pour évaluer la construction ou l'enrichissement d'ontologies est qu'il n'existe pas de métrique standard pour vérifier automatiquement l'exactitude des représentations logiques obtenues. Il est en effet possible, dans la plupart des cas, de produire de nombreuses ontologies différentes pour un même domaine. En partant des propriétés classiques de cohérence, complétude et concision (Gómez-Pérez (1996)), plusieurs critères d'évaluation peuvent être considérés en prenant en compte les différents aspects d'une ontologie, tels que sa structure, son lexique, la syntaxe de ses définitions, ou bien sa sémantique. Ceux-ci permettent d'évaluer directement et intrinsèquement une ontologie. D'autres approches sont également envisageables selon le domaine, le type d'ontologie ou l'existence d'un corpus de référence (Brank et al. (2007)). Une démarche alternative consisterait à évaluer l'usage de l'ontologie dans un cadre applicatif, par exemple pour une tâche de recherche d'informations.

En gardant l'ossature de notre approche, il est possible d'exploiter et comparer différentes représentations sémantiques des énoncés. Cela nécessite la définition de nouveaux schémas RDF et l'adaptation des règles d'extraction. C'est ce que nous avons fait avec la prise en charge des graphes AMR en complément des graphes UNL. Pour la suite, nous envisageons plusieurs axes de développements pour étendre la couvertures des phénomènes linguistiques et dépasser les limites des représentations sémantiques "linguistiques". Notre ambition est de produire des ontologies suffisamment riche pour une mise en oeuvre sur des applications réelles, avec un système paramétrable en terme de domaine métier et de point de vue sur les textes.

Références

Abend, O. et A. Rappoport (2017). The state of the art in semantic representation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Volume 1, Vancouver, Canada, pp. 77–89. Association for Computational Linguistics.

Banarescu, L., C. Bonial, S. Cai, M. Georgescu, K. Griffitt, U. Hermjakob, K. Knight, P. Koehn, M. Palmer, et N. Schneider (2013). Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, Sofia, Bulgaria, pp. 178–186. Association for Computational Linguistics.

Brank, J., M. Grobelnik, et D. Mladenić (2007). Automatic Evaluation of Ontologies. In *Natural Language Processing and Text Mining*, London, pp. 193–219. Springer.

Burns, G. A., U. Hermjakob, et J. L. Ambite (2016). Abstract meaning representations as linked data. In *The Semantic Web – ISWC 2016*, Cham, pp. 12–20. Springer.

Flanigan, J., S. Thomson, J. Carbonell, C. Dyer, et N. A. Smith (2014). A discriminative graph-based parser for the abstract meaning representation. In *Proceedings of the 52nd ACL Annual Meeting*, Baltimore, pp. 1426–1436. Association for Computational Linguistics.

Gómez-Pérez, A. (1996). Towards a framework to verify knowledge sharing technology. *Expert Systems with Applications 11*(4), 519–529.

Khadir, A. C., H. Aliane, et A. Guessoum (2021). Ontology learning : Grand tour and challenges. *Computer Science Review 39*, 100339.

Lamercerie, A. (2021). *Principe de transduction sémantique pour l'application de théories d'interfaces sur des documents de spécification*. Thèse, Université Rennes 1 ; Rennes 1.

Liu, Y., W. Che, B. Zheng, B. Qin, et T. Liu (2018). An AMR Aligner Tuned by Transition-based Parser. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, pp. 2422–2430. Association for Computational Linguistics.

Marcus, M. P., B. Santorini, et M. A. Marcinkiewicz (1993). Building a large annotated corpus of English : The Penn Treebank. *Computational Linguistics 19*(2), 313–330.

Palmer, M., D. Gildea, et P. Kingsbury (2005). The proposition bank : An annotated corpus of semantic roles. *Computational Linguistics 31*(1), 71–106.

Rouquet, D., A. Lamercerie, V. Bellynck, C. Boitet, V. Berment, et G. d. Malézieux (2022). TENET, un outil pour construire des ontologies OWL à partir de textes en langue naturelle. *Revue des Nouvelles Technologies de l'Information RNTI-E-38*, 429–436.

Uchida, H., M. Zhu, et T. Della Senta (1996). Unl : Universal networking language–an electronic language for communication, understanding, and collaboration. *Tokyo : UNL Center*.

Wang, C., N. Xue, et S. Pradhan (2015). A transition-based algorithm for AMR parsing. In *Proceedings of the 2015 North American Chapter ACL Conference*, Denver, pp. 366–375.

Zhang, S., X. Ma, K. Duh, et B. Van Durme (2019). AMR Parsing as Sequence-to-Graph Transduction. In *Proceedings of the 57th ACL Annual Meeting*, Florence, pp. 80–94.

Summary

This paper presents the extension of a tool to automatically build OWL ontologies from texts expressed in natural language. The implemented processing chain takes unconstrained statements as a starting point and results in a logical structure encoding the extracted knowledge. There are two main phases: (1) RDF serialisation of AMR graphs exploited as pivotal representations, and (2) extraction of logical content through the analysis of the intermediate structures. The implementation is based on Semantic Web standards.