Université de Montréal

Protocoles d'évaluation pour l'extraction d'information libre

par
William Léchelle

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Thèse présentée à la Faculté des études supérieures
en vue de l'obtention du grade de Philosophiæ Doctor (Ph.D.)
en informatique

Juillet, 2019

RÉSUMÉ

On voudrait apprendre à « lire automatiquement ». L'extraction d'information consiste à transformer des paragraphes de texte écrits en langue naturelle en une liste d'éléments d'information autosuffisants, de façon à pouvoir comparer et colliger l'information extraite de plusieurs sources. Les éléments d'information sont ici représentés comme des relations entre entités : (*Athéna* ; *est la fille de* ; *Zeus*).

L'*extraction d'information libre* (EIL) est un paradigme récent, visant à extraire un grand nombre de relations contenues dans le texte analysé, découvertes au fur et à mesure, par opposition à un nombre restreint de relations prédéterminées comme il est plus courant. Cette thèse porte sur l'évaluation des méthodes d'EIL.

Dans les deux premiers chapitres, on évalue *automatiquement* les extractions d'un système d'EIL, en les comparant à des références écrites à la main, mettant respectivement l'accent sur l'informativité de l'extraction, puis sur son exhaustivité. Dans les deux chapitres suivants, on étudie et propose des alternatives à la *fonction de confiance*, qui juge des productions d'un système. En particulier, on y analyse et remet en question les méthodologies suivant lesquelles cette fonction est évaluée : d'abord comme modèle de validation de requêtes, puis en comparaison du cadre bien établi de la complétion de bases de connaissances.

Mots-clefs : extraction d'information libre, évaluation, bases de connaissances

ABSTRACT

Information extraction consists in the processing of natural language documents into a list of self-sufficient informational elements, which allows for cross collection into Knowledge Bases, and automatic processing. The facts that result from this process are in the form of relationships between entities : (*Athena ; is the daughter of ; Zeus*).

Open Information Extraction (OIE) is a recent paradigm the purpose of which is to extract an order of magnitude more relations from the input corpus than classical IE methods, what is achieved by encoding or learning more general patterns, in a less supervised fashion. In this thesis, I study and propose new evaluation protocols for the task of Open Information Extraction, with links to that of Knowledge Base Completion.

In the first two chapters, I propose to *automatically* score the output of an OIE system, against a manually established reference, with particular attention paid to the *informativity* and *exhaustivity* of the extractions. I then turn my focus to the confidence function that qualifies all extracted elements, to evaluate it in a variety of settings, and propose alternative models.

Keywords : open information extraction, evaluation, knowledge base

TABLE DES MATIÈRES

LISTE DES TABLEAUX

LISTE DES FIGURES

CHAPITRE 1

INTRODUCTION

La compréhension de texte est une des tâches fondamentales du traitement automatique des langues. On voudrait apprendre à une machine à « lire automatiquement » ; à transformer des phrases en anglais en entrée, en données qui ont une signification pour la machine en sortie, signification qui corresponde à celle de la phrase initiale.

Plusieurs théories linguistiques s'attardent à définir la sémantique du langage. Dans cette thèse, c'est la tâche d'*extraction d'information* qui précise la représentation de l'information que l'on veut obtenir en sortie. Dans ce cadre, et comme pour le Web Sémantique, des *entités* sont reliées par des *relations*. L'information extraite sera représentée et stockée essentiellement sous forme de triplets (entité A, relation R, entité B)[1], ce qui signifie que la relation R s'applique entre les entités A et B. Des exemples de triplets sont présentés en figure 1.1. Dans de nombreux cas, des n-uplets seraient plus appropriés pour représenter ces connaissances, comme (*Quebec City ; is located ; on the north bank of the Saint Lawrence River ; in the Saint Lawrence River valley*)[2], au lieu des deux triplets séparés (*Quebec City ; is located ; on the north bank of the Saint Lawrence River*) et (*Quebec City* ; *is located* ; *in the Saint Lawrence River valley*).

L'extraction de telles relations permet d'ordonner l'information présente dans le texte non structuré, et de les arranger dans une base de connaissances plus ou moins formelle (avec plus ou moins de contraintes). On peut alors faire des inférences sur cette base, et répondre à des requêtes. De nombreuses tâches de traitement des langues bénéficient de telles bases de connaissances, comme la réponse automatique

[1] Un point de terminologie : on désignera parfois l'entité A de *sujet*, et l'entité B d'*objet*. Ou encore, on dira que la relation R est un *prédicat*, et que les différentes entités sont ses arguments (premier argument, deuxième argument, etc.). On ne fait pas ici de distinction entre ces différentes appellations.

[2] Pour faciliter la lecture, les exemples sont donnés en français lorsque le parallèle avec l'anglais est transparent, et en anglais sinon.

(Quebec ; was founded by ; Samuel de Champlain)
(Quebec City ; is located on ; the north bank of the Saint Lawrence River)
(Quebec City ; is located in ; the Saint Lawrence River valley)
(Quebec City ; is an important hub in ; the province's autoroute system)
(Quebec City ; was struck by ; the 1925 Charlevoix-Kamouraska earthquake)
(Lower Town ; is located at ; shore level)
(The zoo ; specialized in ; winged fauna and garden themes)
(A high stone wall ; surrounds this portion of ; the city)
(The Terrasse Dufferin ; leads toward ; the nearby Plains of Abraham)

Figure 1.1 – **Exemples de relations extraites** à partir de l'article de Wikipédia sur Québec. Les entités mises en relation peuvent être toutes sortes de groupes nominaux.

à des questions (Fader et al., 2014), l'apprentissage d'ontologies (Poon et Domingos, 2010; Velardi et al., 2013; Suchanek et al., 2007; Mitchell et al., 2015), la recherche sémantique (Etzioni, 2011) ou le résumé automatique (Rashidghalam et al., 2016; Timofeyev et Choi, 2017).

Notre objectif est donc de décomposer un document complexe en une série d'éléments d'information élémentaires, qui couvrent au maximum le message du document. Ce processus peut être apparenté à l'abstraction et simplification de phrases : pour chaque élément d'information exprimé dans le texte source, on veut reformuler la phrase de manière à exprimer le plus succinctement possible cet élément d'information (et autant que possible, seulement celui-là). De fait, certains systèmes étudiés dans ces pages (Stanford Open IE, MinIE) suivent exactement cette logique.

Deux familles d'approches coexistent en extraction d'information. Traditionnellement, les relations à extraire sont en petit nombre, déterminées à l'avance, et l'on entraine un modèle d'extraction par relation d'intérêt, grâce à des exemples d'entrainement (ou plus simplement, un modèle est construit à la main). L'inconvénient de cette approche (décrite plus en détail en section 2.1.6) est la nécessité d'annoter plusieurs exemples d'entrainement pour chaque relation, pour entrainer chaque modèle indépendamment, ce qui est coûteux.

Plus récemment, l'Extraction d'Information Libre[3] (EIL) met à profit les grandes quantités de texte non structuré aujourd'hui disponibles pour extraire le plus de relations possible, avec un modèle commun. L'intérêt majeur consiste à s'affranchir de l'ensemble pré-spécifié de relations, la difficulté étant de faire une bonne utilisation d'un ensemble d'entrainement commun à toutes les relations pour obtenir un modèle d'extraction correct. Une autre difficulté consiste à structurer les nombreuses relations plus ou moins synonymes produites par cette méthode. Banko et Etzioni (2008) comparent les deux familles d'approches, pour montrer que l'EIL est un changement de paradigme nécessaire lorsque les relations à extraire sont nombreuses et initialement inconnues ; et que dans le cas contraire, l'EIL arrive à une précision comparable aux systèmes traditionnels, souffrant seulement d'un rappel nettement moindre. Les auteurs montrent aussi que les deux approches peuvent se complémenter, pour réduire le nombre d'exemples d'entrainement à étiqueter pour atteindre une performance donnée.

Cette thèse porte sur l'évaluation des méthodes d'EIL. Lorsque nous avons entamé ces travaux en 2014, aucune méthode ne permettait d'évaluer objectivement (de façon automatique et reproductible) la performance d'un système d'EIL, ou bien de comparer deux systèmes. Ce travail explore plusieurs avenues pour mesurer cette performance.

À mon sens, la tâche d'EIL se décompose en deux parties. Dans un premier temps, du texte source sont extraits des faits candidats. Dans un deuxième temps, les informations pertinentes sont sélectionnées parmi cet ensemble de candidats. Intuitivement, la première étape se veut exhaustive, et est évaluée en rappel. La deuxième étape est une tâche de discrimination, évaluée en précision. Les chapitres 3 et 4 sont centrés sur la première partie, et les chapitres 5 et 6 sur la deuxième partie.

Je présente les travaux reliés au sujet de ma thèse au chapitre 2, d'abord les

[3]On peut traduire l'anglais *Open Information Extraction* par extraction d'information « non restreinte », « non contrainte », « ouverte » par calque, ou dans une certaine mesure « non supervisée ». J'utiliserai le plus souvent « extraction d'information **libre** », qui rend le mieux l'idée d'origine.

systèmes d'extraction d'information existants, puis les travaux consacrés à leur évaluation.

Ensuite, deux chapitres s'attachent à évaluer *automatiquement* les extractions d'un système d'EIL, en les comparant à des références *préalablement* établies manuellement. Ces références mettent l'accent respectivement sur l'informativité de l'extraction (chapitre 3), puis sur son exhaustivité (chapitre 4), entre autres principes directeurs (détaillés en annexe A).

Dans les deux chapitres suivants, j'étudie et propose des alternatives à la **fonction de confiance** qui juge des productions d'un système. En particulier, j'analyse et questionne les méthodologies suivant lesquelles cette fonction est évaluée : d'abord comme modèle de validation de requêtes, au chapitre 5, puis en comparaison du cadre bien établi de la complétion de bases de connaissances, au chapitre 6.

Le chapitre 3 s'ancre dans la compréhension de texte. L'information d'un texte a été correctement extraite, argumente-t-on, si l'on peut répondre aux questions auxquelles le texte permet de répondre, seulement grâce à l'information extraite. Qui plus est, puisque le processus d'extraction structure l'information, la réponse aux questions devrait pouvoir se faire de façon automatique. Un petit corpus de phrases indépendantes est ainsi annoté en questions/réponses (environ 2 ou 3 par phrase, quand la phrase s'y prête). Un système basique de questions-réponses automatique est créé, pour quantifier la capacité des systèmes à répondre aux questions, directement à partir de leurs productions. Les résultats, sans être surprenants, soulignent les forces et faiblesses des systèmes jugés. Ce travail a mis en évidence le grand nombre de phrases n'ayant de sens qu'en contexte : environ deux tiers des phrases du corpus étudié ne répondent à aucune question d'ordre général. D'autres auteurs (Fader et al., 2014) ont exploré la même idée de façon indépendante.

Dans la même veine, le chapitre 4 s'attache à *préciser la définition même* de la tâche d'EIL. Malgré la prolifération de travaux en EIL et de nombreux nouveaux systèmes d'extraction, la tâche en elle-même est restée largement sous-spécifiée, en dehors des objectifs de haut niveau initialement établis. Par exemple, les faits « techniquement corrects » mais non informatifs par manque de contexte ou parce

qu'ils sont excessivement génériques, comme (*le restaurant ; est parfois ouvert ; les mardis*) doivent-ils être extraits ou filtrés ? Les anaphores doivent-elles être résolues ou (*il ; est né ; le 26 février 1931*) est-elle une information valide ? Les faits impliqués plutôt qu'exprimés par la phrase doivent-ils être également extraits, et si oui, jusqu'à quel point[4] ? À quel niveau de spécificité faut-il segmenter les arguments ? Quelle est l'influence sémantique du contexte de l'extraction ? Nous avons construit une référence en faisant des choix par rapport à chacun de ces enjeux, sur la base des principes fondamentaux de l'EIL (informativité, exhaustivité, et minimalité des informations). Nous discutons de ces enjeux et établissons des directives d'annotation, pour clarifier les exigences de la tâche d'extraction. Enfin, nous proposons une procédure d'évaluation automatique permettant de mesurer la performance d'un système de façon détaillée, à l'aune de cette référence.

Outre la production de relations, la deuxième facette de l'extraction d'information consiste à *assigner des scores de confiance* à ces relations. La fonction de confiance est ce qui différencie les extractions hautement probables, « faciles » (par exemple, en concordance directe avec la phrase), des faits obtenus plus difficilement, à l'aide de patrons syntaxiques rares, davantage sujets aux erreurs. Cette confiance permet lors de l'extraction de mettre l'accent sur la précision ou le rappel, selon les besoins.

À la question de savoir si un fait est plausible ou non, les systèmes d'EIL consultent leur corpus textuel à la recherche de phrases exprimant ce fait (et, le cas échéant, y assignent un fort score de confiance). Les systèmes de prédiction de liens, de la même façon, consultent une base de connaissances établie, à la recherche de faits similaires qui apporteraient du crédit à la requête. S'il est connu que « *les hommes ; sont ; mortels* » et que « *les Grecs ; sont ; mortels* », alors il est vraisemblable que « *les philosophes ; sont ; mortels* », même si cette information n'est pas inscrite dans la base de connaissances, ou si aucune phrase ne l'énonce clairement, parce que « *les philosophes* » est proche de « *les hommes* » et de « les

[4] « Zoé a réussi à convaincre Yahn » implique (Zoé ; a convaincu ; Yahn) et « Xavier a écrit le livre dont est tirée la citation » implique (Xavier ; a écrit ; la citation), mais le premier lien est plus direct que l'autre.

Grecs ». C'est l'idée de base de Angeli et Manning (2013), dont l'exemple est tiré. J'examine la même tâche de prédiction de faits candidats, dans un article publié à la conférence LREC en 2018 (chapitre 5). Je propose un modèle alternatif de prédiction (un modèle de langue), et mesure l'impact des paramètres de la procédure d'évaluation : il est plus facile de prédire un fait dont la relation diffère d'un fait connu, que si c'est un argument qui varie.

De nombreux travaux développent des modèles de prédiction de liens fondés sur le plongement des entités et des relations dans des espaces vectoriels (*entity embeddings*). Ces modèles sont fonctionnellement équivalents à la fonction de confiance de l'EIL, mais malheureusement presque exclusivement évalués dans le cadre des bases de connaissances établies, de taille modeste, et bien maitrisées, principalement Freebase[5] et WordNet[6]. Dans un article en préparation (chapitre 6), nous adaptons ces modèles à des bases de connaissances produites par EIL : comportant du bruit, mais plus larges. On observe que de prendre en compte la synonymie des relations aide à compléter les liens manquants dans la base.

Comme il ressortira des différents chapitres, l'évaluation de l'EIL nécessite une certaine méticulosité, car le diable se cache invariablement dans les détails des cas de figure les moins fréquents.

[5]Base créée par Google en 2007 et intégrée en 2015 à Wikidata – `www.wikidata.org`.
[6]Base de données lexicale créée par Princeton - (Miller, 1995), `wordnet.princeton.edu`.

CHAPITRE 2

ÉTAT DE L'ART ET TRAVAUX RELIÉS

Dans cette section, je présente les travaux reliés au sujet de ma thèse. Je commence par une revue des systèmes d'extraction d'information libre, de TextRunner (Yates et al., 2007) à Graphene (Cetto et al., 2018). J'analyse ensuite le nombre restreint de travaux consacrés à l'évaluation de ces systèmes, ce qui est à proprement parler mon sujet en ces pages. Notons qu'une revue de littérature du domaine a été publiée récemment par Niklaus et al. (2018).

Je donne un aperçu du paysage en extraction d'information « traditionnelle » (restreinte à quelques dizaines de relations prédéterminées) en section 2.1.6. Ces travaux dans un environnement plus contrôlé et mieux maitrisé sont des sources d'inspiration et utiles pour mettre mes propres travaux en contexte. Par exemple, le dernier article présenté en ces pages transpose le protocole d'évaluation traditionnellement appliqué sur Freebase aux plus vastes bases de connaissances produites par l'EIL.

Une étape possible de l'extraction d'information, qui n'est pas l'objet de cette étude, consisterait à lier les fragments de texte à des références sémantiquement spécifiées, pour en ancrer le sens (*grounding* en anglais) (par exemple, relier « Obama » à « `http://www.wikidata.org/entity/Q76` »). Dans ce travail, on suppose globalement que les entités extraites sont clairement compréhensibles (mais voir la section 3.3.3 pour une clarification de cette hypothèse).

2.1 État de l'art en EIL

Le paradigme d'extraction d'information libre (en anglais *OIE* pour *Open Information Extraction*) est apparu en 2007 avec le système TextRunner (Yates et al., 2007), par contraste avec l'extraction d'information « classique ». Le constat des extracteurs précédents, par exemple Snowball (Agichtein et Gravano, 2000) et Kno-

wItAll (Etzioni et al., 2005), était qu'ils ne pouvaient extraire que des relations visées, et nécessitaient une quantité d'annotation manuelle prohibitive pour passer à l'échelle des grands corpus, contenant un grand nombre de relations différentes. Ces systèmes ne fonctionnaient que dans le domaine spécifique sur lequel ils étaient entrainés, et, de par leur complexité, que sur des corpus de taille modeste. (Sekine, 2006) et (Shinyama et Sekine, 2006) présentent une autre approche visant à pallier aux mêmes problèmes.

La toute première génération d'extracteurs d'information libres, incarnée par le système TextRunner (Yates et al., 2007), vise à extraire rapidement (sans analyse syntaxique), et en une seule passe sur le corpus, toutes les relations d'intérêt, indépendamment du domaine. Depuis, de nombreux systèmes ont été développés avec les mêmes objectifs : StatSnowBall (Zhu et al., 2009), Wanderlust (Akbik et Bross, 2009), KraKen (Akbik et Löser, 2012) (employé pour construire WeltModell (Akbik et Michael, 2014)), Reverb (Fader et al., 2011), R2A2 (Etzioni et al., 2011), WOE (Wu et Weld, 2010), Ollie (Mausam et al., 2012), CSD-IE (Bast et Haussmann, 2013), ClausIE (Del Corro et Gemulla, 2013), MinIE (Gashteovski et al., 2017), Stanford OIE (Angeli et al., 2015b), Graphene (Cetto et al., 2018), etc. La figure 2.1 montre les extractions produites par six systèmes d'EIL dont le code est disponible, sur quelques phrases. Dans cette section, je présente quelques-unes de ces approches en détail.

2.1.1 Reverb

Dans le système TextRunner (Yates et al., 2007), les arguments sont détectés grâce à un segmenteur, basé sur les parties du discours. La relation est constituée des mots situés entre deux arguments, et deux fonctions de confiance filtrent les faits candidats ainsi obtenus. La première décide sur la base des mots entre arguments, s'il s'agit vraisemblablement d'une relation, et la deuxième, sur la base de l'ensemble des candidats, lesquels sont fiables de par leur répétition. Dans un article subséquent (O-CRF, (Banko et Etzioni, 2008)), un modèle CRF est ajouté, qui décide d'inclure ou non chaque mot dans la relation, ce qui permet d'extraire de plus nombreux

His parents are Ashkenazi Jews who had to flee from Hungary during World War II.

Reverb

(His parents ; are ; Ashkenazi Jews)
(Ashkenazi Jews ; had to flee from ; Hungary)

Ollie

(Ashkenazi Jews ; had during ; World War II)

OpenIE 4

(His parents ; are ; Ashkenazi Jews)

ClausIE

(His ; has ; parents)
(His parents ; are ; Ashkenazi Jews)
(Ashkenazi Jews ; had ; to flee from Hungary during World War II)

MinIE

(His ; has ; parents)
(parents ; are ; Ashkenazi Jews)
(Ashkenazi Jews ; had to flee from Hungary during ; World War II)
(Ashkenazi Jews ; had to flee from ; Hungary)

Stanford OpenIE

∅

Figure 2.1 – Phrase extraite de `https://en.wikipedia.org/wiki/Chilly_Gonzales`. La plupart des systèmes capturent la relation facile d'accès exprimée par le verbe principal « *are* ». Reverb, ClausIE et MinIE extraient des variations sur (*Ashkenazi Jews* ; *had to flee* ; *from Hungary* ; *during World War II*), ce qui est une généralisation de ce que dit la phrase. Quantifiée par « `Some` » comme dans la base de connaissances Aristo (Mishra et al., 2017), l'extraction serait correcte. Aucun système n'est capable de traiter l'anaphore pour produire (*His parents* ; *had to flee*). Ollie et Stanford OpenIE ont de la difficulté avec la structure de cette phrase.

To those fleeing persecution, terror & war, Canadians will welcome you, regardless of your faith.

Reverb

(Canadians ; will welcome ; you)

Ollie

(Canadians ; will welcome regardless of your faith ; you)
(Canadians ; will welcome you regardless of your faith to ; those fleeing persecution)
(you ; will be welcome regardless of your faith to ; those fleeing persecution)

OpenIE 4

(those ; fleeing ; persecution, terror & war)
(Canadians ; will welcome regardless ; you)

ClausIE

(Canadians ; will welcome ; you ; regardless of your faith ; To those fleeing persecution terror &)
(Canadians ; will welcome ; you ; regardless of your faith ; To war)
(your ; has ; faith)

MinIE

(terror & war ; Canadians welcome you To ; fleeing persecution)
(terror & war ; Canadians welcome you regardless of ; faith))
(terror & war ; Canadians welcome ; you)
(Canadians ; terror & war welcome you To ; fleeing persecution)
(Canadians ; terror & war welcome you regardless of ; faith)
(Canadians ; terror & war welcome ; you)
(your ; has ; faith)

Stanford OpenIE

(those ; fleeing ; war)
(Canadians ; will welcome ; you)
(those ; fleeing ; persecution)
(those ; fleeing ; terror)

Figure 2.2 – Célèbre réponse du premier ministre du Canada en réponse à l'interdiction de voyager de Trump, sur Twitter, le 28 janvier 2017. La dislocation et anaphore emphatique de « *those fleeing persecution* » perturbe les règles d'extraction, car peu de systèmes sont équipés pour traiter les anaphores. Reverb extrait correctement les extractions simples. Ollie fait une petite erreur en intégrant à tort le modificateur « *regardless of your faith* » dans la relation, mais capture l'essence de la phrase avec succès. ClausIE fait une erreur d'analyse syntaxique sur le début de la phrase, interprétant « & » comme « *To those fleeing persecution+terror, to war, Canadians will welcome you* ». MinIE est perplexifié par « terror & war » et ne sait pas comment le traiter.

Research is hard, and it takes some getting used to.

Reverb

ø

ClausIE, MinIE, Stanford OpenIE, OpenIE 4

Research ; is ; hard

Ollie

it ; is ; hard

ClausIE

it ; takes ; ø
some ; getting used ; to

OpenIE 4

some ; getting ; used to

Figure 2.3 – Phrase extraite de `https://academia.stackexchange.com/a/2222/2740`. La plupart des systèmes traitent la simple proposition principale correctement, mais échouent largement sur l'idée principale de la phrase. Une extraction correcte serait (*research being hard ; takes ; some getting used to*).

candidats.

L'inconvénient de ce système est que certaines erreurs rendent la relation incompréhensible, même pour des humains. Par exemple, dans la phrase suivante, TextRunner extrait la relation incohérente « *for assumed* ».

> « Extendicare agreed to buy *Arbor Health Care* **for** about US $432 million in cash and **assumed** *debt.* »
>
> `TextRunner` : (*Arbor Health Care* ; * **for assumed** ; *debt*)

Pour pallier ce problème, Fader, Soderland et Etzioni (2011) ont déterminé un certain nombre de contraintes, portant sur les relations verbales binaires, les plus fréquentes, qui sont respectées par la plupart d'entre elles. Concrètement, pour le système Reverb, une relation doit :

- constituer un segment continu de phrase, présent *entre* les deux entités reliées ;
- suivre une expression régulière sur les parties du discours présentée en figure 2.4 ;
- être présente avec suffisamment de variété dans ses arguments dans un grand corpus[1].

La dernière contrainte vise à filtrer les relations trop spécifiques, comme « *is offering only modest greenhouse gas reduction targets at* » dans la phrase :

> *The Obama administration* **is offering only modest greenhouse gas reduction targets at** *the conference.*

L'algorithme de Reverb[2] est le suivant :

[1] Dans (Fader et al., 2011), cette contrainte est implémentée comme « plus de 20 paires différentes d'arguments dans un corpus de 500 millions de phrases tirées du Web ».

[2] Reverb, comme TextRunner utilise OpenNLP pour le découpage en constituants et l'extraction des parties du discours.

V | V P | V W* P
V = verbe particule ? adv ?
W = (nom | adj | adv | pron | det)
P = (prep | particule | infinitif)

Figure 2.4 – Une expression régulière simple portant sur les parties du discours réduit le nombre de relations incohérentes comme « *was central torpedo* » et couvre les relations exprimées par des constructions à verbe support comme « *gave a talk at* ». (Figure traduite de Fader et al. 2011.)

Découvrir les relations : pour chaque verbe, chercher une séquence de mots environnants qui répond aux contraintes ;

Déterminer les arguments : pour chaque relation identifiée, chercher à droite et à gauche un groupe nominal acceptable (pas un pronom relatif, pas un pronom interrogatif, etc.).

Reverb fournit un score de confiance pour chaque extraction, basé sur un petit nombre de caractéristiques de surface de la phrase et de l'extraction. La fonction de score a été entrainée sur un petit corpus de 1000 phrases dont les extractions ont été annotées manuellement comme correctes ou non.

Aujourd'hui, la grande force de ce système est sa simplicité et sa rapidité d'exécution. En ne s'appuyant que sur les parties du discours, il ne nécessite pas la longue analyse syntaxique que nécessitent les extracteurs plus récents, et est donc plus rapide. Concrètement, dans mon expérience, Reverb traite environ 200 phrases par seconde. Le code de Reverb est disponible à `https://github.com/knowitall/reverb`.

La relative simplicité de l'algorithme a inspiré des travaux portant sur d'autres langues que l'anglais à reproduire des systèmes comparables, par exemple (Gotti et Langlais, 2016) et (Zhila et Gelbukh, 2014).

2.1.2 Ollie et successeurs

Développé par la même équipe, Ollie[3] (Mausam et al., 2012) cherche à améliorer deux choses par rapport à Reverb : l'extraction de prédicats non verbaux, et la prise en compte du contexte, par exemple dans le cas d'attribution de propos. Extraire (la Terre ; être ; plate) de « D'après les premières cosmogonies, la Terre était plate » serait insuffisant.

Comme l'énorme majorité des systèmes d'EIL plus récents, Ollie s'appuie sur une analyse syntaxique du texte d'entrée pour procéder à l'extraction. Grâce à l'utilisation du parseur Malt[4], reconnu pour sa vitesse, Ollie peut analyser environ 40 phrases par seconde en moyenne, d'après mon expérience.

Un sous-ensemble d'extractions de haute précision de Reverb est utilisé comme vérité de référence, pour compiler un ensemble d'entrainement constitué de toutes les autres occurrences de ces faits dans un grand corpus. Par exemple, Reverb extrait (*Paul Annacone ; is the coach of ; Federer*) de la phrase « *Paul Annacone is the coach of Federer* ». Ollie peut alors incorporer dans son ensemble d'entrainement des phrases comme « *Now coached by Annacone, Federer is winning more titles than ever* ».

Les structures syntaxiques de ces nouvelles occurrences sont répertoriées, et généralisées en patrons, soit purement syntaxiques, soit avec des contraintes lexicales ou sémantiques. Un modèle d'extraction pour Ollie est simplement une liste de patrons syntaxiques, incluant des contraintes. Un exemple de patron purement syntaxique est `{arg1} ↑nsubj↑ {rel:postag=VBD} ↓dobj↓ {arg2}` (l'argument 2 est l'object direct (`dobj`) d'un verbe dont le sujet (`nsubj`) est l'argument 1). On trouve un exemple de contrainte sémantique dans le patron « `{rel:postag=NN;type=Person} ↑nn↑ {arg1} ↓nn↓ {arg2}` » (qui correspond à l'apposition « *Microsoft co-founder Bill Gates* », où « *co-founder* » est la relation). Enfin, « `{arg1} ↑nsubjpass↑ {slot:postag=VBN;lex` ∈ `announce|name|choose...} ↓dobj↓ {rel:postag=NN} ↓{prep` ∗ `}↓ {arg2}` » est un exemple de patron lexi-

[3] `https://github.com/knowitall/ollie`

[4] (Nivre et al., 2006) – `http://www.maltparser.org/`

calisé, correspondant à la structure « *Federer hired Annacone as a coach* », et où « *hire* » pourrait aussi bien être « *announce* », « *name* », « *choose* », etc.

Pour procéder à l'extraction, Ollie cherche les occurrences dans le texte des patrons du modèle. Quand une phrase correspond à un patron, le sous-arbre correspondant est développé pour inclure toutes les informations pertinentes dans le fait extrait. Des heuristiques déterminent quels liens syntaxiques sont pertinents à inclure ou non.

Par exemple, dans la phrase « *I learned that the 2012 Sasquatch music festival is scheduled for May 25th until May 28th.* », un patron syntaxique correspondrait aux noeuds « *festival - be scheduled for - 25th* », mais seuls, ils formeraient une extraction incomplète. Les noeuds voisins sont incorporés pour former (*the Sasquatch music festival ; be scheduled for ; May 25th*).

Enfin, le contexte de la phrase est analysé, pour produire les extractions finales. Les attributions sont marquées dans l'analyse syntaxique par un lien déterminé (« complément clausal »), qui est détecté et analysé lorsqu'il survient. Certains modificateurs de clauses (signalés par un lien « clause adverbiale ») sont similairement extraits, notamment la structure « <fait> si <condition> ». De cet effort résultent des faits extraits comme `[(la Terre ; est le centre de ; le monde) ; Attribution croyaient ; les premiers astronomes]`, ou encore `[(le pique-nique ; est ; annulé) ; ModificateurClausal si ; il pleut]`.

CSD-IE (Bast et Haussmann, 2013) est un autre système qui vise à capturer le contexte des faits énoncés : des *références* à d'autres faits peuvent figurer parmi les arguments d'un fait, ce qui permet d'exprimer des faits à propos d'autres faits. Par exemple, si #1 est (la Terre ; est le centre de ; le monde), CSD-IE extrait de la phrase « Les premiers astronomes croyaient que la Terre était le centre du monde. » le deuxième fait (Les premiers astronomes ; croyaient ; que #1)[5]. CSD-IE

[5]la position de la préposition, derrière la relation, ou devant l'objet, varie d'un système à l'autre. La plupart des systèmes placent la préposition avec la relation (les astronomes ; croyaient que ; #1), ce qui pose problème dans le cas des quadruplets ou les deux objets ne partagent pas la même préposition : (Dominique ; vivait ; à Québec ; en 1932).

porte l'accent sur la minimalité des faits : chaque élément d'information mérite une extraction autonome. Le principe de fonctionnement de CSD-IE est très similaire à celui de ClausIE, présenté dans la section suivante. La phrase est segmentée en clauses, et de ces clauses sont extraits des faits.

Plusieurs travaux font suite à Ollie. OpenIE 4[6] est une version de travail du code utilisée dans plusieurs publications, mais jamais publiée pour elle-même. La revue de l'EIL par Mausam est ce qui s'approche le plus d'une description. Deux modules nouveaux sont combinés à Ollie, SRLIE et RelNoun. SRLIE, un développement des idées de (Christensen et al., 2011) produit des faits n-aires et lie plusieurs faits entre eux, sur la base d'une analyse en rôles sémantiques. RelNoun (Pal et Mausam, 2016) est un système à base de règles qui se concentre sur les gentilés et les relations nominales, principalement les titres et les professions (par exemple « directeur de »).

Encore plus récemment, OpenIE 5.0[7] est disponible, qui rassemble SRLIE, RelNoun, BONIE (Saha et al., 2017) et CalmIE (Saha et Mausam, 2018). BONIE se concentre sur l'analyse des quantités en lieu d'arguments, améliorant la précision de leur détection et la spécification de leurs unités, et précisant la relation des faits les contenant, le cas échéant. Par exemple (Donald Trump ; a ; environ 70 ans) devient (Donald Trump ; *a pour âge* ; environ 70 ans), grâce à l'analyse du deuxième argument numérique. CalmIE se concentre sur l'analyse des conjonctions, et permet d'extraire les 6 informations indépendantes de la phrase « *Jack and Jill visited India, Japan and South Korea.* »

2.1.3 ClausIE et MinIE

ClausIE (Del Corro et Gemulla 2013), comme Ollie, nécessite une analyse syntaxique du texte d'entrée pour en extraire les informations. Par contre, ClausIE n'utilise aucun apprentissage, et ne nécessite pas de données d'entrainement : passée

[6] Hébergée de façon instable, actuellement à `https://github.com/allenai/openie-standalone`

[7] `https://github.com/dair-iitd/OpenIE-standalone`

l'analyse syntaxique, l'extraction suit simplement un ensemble complexe de règles établies par les auteurs du système. Ces règles s'appuient lourdement sur le concept de *clauses*. Suivant les auteurs, une clause est une partie de phrase qui exprime une information cohérente, elle est constituée d'un sujet, d'un verbe, et optionnellement d'un complément d'objet et de compléments circonstanciels. Les clauses peuvent être de différents types, dépendamment des éléments présents (Sujet - Verbe - Objet (SVO), Sujet - Verbe - Complément (SVC), SVOC, etc.). Un arbre de décision (établi par les auteurs) permet de déterminer le type d'une clause. Une fois une clause typée, ClausIE détermine comment exprimer les différents constituants présents en relations cohérentes. Typiquement, une clause génère plusieurs extractions, pour les différents sous-ensembles de constituants qui forment une information cohérente.

Si on considère par exemple la phrase :

> *Bell, a telecommunication company, which is based in Los Angeles, makes and distributes electronic, computer, and building products.*

ClausIE découvre 3 clauses :

- (S : *Bell*, $\mathrm{V}_{\mathrm{copule}}$: *is*, C : *company*)
- (S : *Bell*, V : *based*, C : *Angeles*)
- (S : *Bell*, V : *makes*, O : *products*)

Après séparation des conjonctions de coordination, ces clauses se déclinent en extractions :

- (*Bell, is, a telecommunication company*),
- (*Bell, is based, in Los Angeles*),
- (*Bell, makes, electronic products*),
- (*Bell, makes, computer products*),
- (*Bell, makes, building products*),

- (*Bell, distributes, electronic products*),
- (*Bell, distributes, computer products*),
- (*Bell, distributes, building products*).

Pour illustrer les combinaisons de constituants pouvant constituer des relations, prenons par exemple la clause analysée :

(S: *Albert Einstein*, V: *died*, C: *[in] Princeton*, C: *[in] 1955*)

ClausIE en extrait 4 clauses dérivées qui deviendront des extractions :

- (S: *Albert Einstein*, V: *died*),
- (S: *Albert Einstein*, V: *died*, C: *[in] Princeton*),
- (S: *Albert Einstein*, V: *died*, C: *[in] 1955*),
- (S: *Albert Einstein*, V: *died*, C: *[in] Princeton*, C: *[in] 1955*).

Concrètement, les extractions produites par ClausIE sont de très bonne qualité linguistique. La multiplication des relations redondantes produites (dû à la décomposition des conjonctions) est un phénomène important, et utile, même s'il peut biaiser les mesures de rendement des extracteurs.

Malheureusement, compte tenu de la relative lenteur de l'analyseur syntaxique embarqué (CoreNLP, produit par Stanford), dans mon expérience, ClausIE ne traite que 2 phrases par seconde environ.

Une extension de ClausIE par le même groupe de recherche, MinIE (Gashteovski et al., 2017), étend le système dans plusieurs directions. Similairement à Ollie, MinIE développe un certain nombre de méta-informations sur les triplets. Suivant Saurí et Pustejovsky (2012), la factualité d'une information est annotée suivant deux axes : une extraction peut être de **polarité** positive (par défaut) ou négative – *Jeanne n'ira pas au bal* – et sa **modalité** peut être la certitude (par défaut) ou la possibilité – *Jean ira peut-être au bal.* Les deux axes de factualité

se complémentent : une information peut être positive et certaine, positive et possible, négative et certaine, ou négative et possible. Comme Ollie, MinIE détecte l'attribution, dans deux situations : lorsqu'un complément adverbial commence par *« according to »*, et lorsqu'un triplet comporte une clause subordonnée complète comme argument et que la relation indique une attribution – par exemple (*John ; believes that ; Earth is flat*). Dans les deux cas, la *source* de l'information est extraite, et une annotation d'attribution, constituée de la source et de la factualité de la source, est ajoutée à l'information extraite de la clause principale. La factualité de la source et celle de l'information principale sont indépendantes : dans le dernier exemple, la source est *John*, de factualité positive incertaine (*believes*). Le fait principal (*Earth is flat*) est positif et certain. L'information finalement extraite serait donc `[(Earth ; is ; flat), (positive, certain), AttributedTo (John, positive, possible)]`.

De plus, comme RelNoun, MinIE détecte plusieurs schémas marquant des relations implicites et extrait les relations correspondantes : par exemple, *« M. Durand »* implique (*Durand ; est ; un homme*), et *« Canada's prime minister Justin Trudeau »* implique (*Justin Trudeau ; is prime minister of ; Canada*). Comme BONIE, MinIE traite les quantités de façon spécifique : un token `<Quantité>` remplace la quantité exprimée, déplacée dans les annotations du triplet (par exemple `<Quantité> = 3`). Ceci est fait dans le but d'unifier les faits ne différant que par la valeur de leur quantité. Rappelons que comme CalmIE, ClausIE divisait déjà les conjonctions en extractions indépendantes.

Comme CSD-IE, MinIE s'attache à minimiser les extractions produites. Plusieurs « modes » (complet, sécuritaire, agressif, etc.) sont autant de compromis entre la précision des informations extraites, et leur minimalité.

2.1.4 Stanford Open IE (SIE)

Similairement à ClausIE, le système d'EIL de Stanford[8] (Angeli et al., 2015b) décompose les phrases d'entrée en clauses pour simplifier l'analyse. Un classifieur

[8]SIE pour « Stanford Open IE » dans la suite

détermine quels constituants de l'arbre syntaxique forment des clauses indépendantes, plus courtes, et impliquées par la phrase, incorporant dans la nouvelle clause des éléments situés plus haut dans l'arbre, si nécessaire. Un petit ensemble de 14 règles segmente ensuite chaque clause indépendante en triplets. La principale différence avec ClausIE est l'utilisation d'un classifieur entrainé par apprentissage automatique comme premier segmenteur, par opposition à un ensemble de règles écrites à la main.

Pour ce système, l'accent est mis sur l'implication logique : un cadre théorique est employé pour identifier les implications / imbrications entre arguments, qui sont alors simplifiés autant que possible, sans changer le sens. Par exemple, dans la phrase « *John gives true praise.* », l'objet « *true praise* » peut être simplifié en « *praise* », et l'extraction (*John ; gives ; praise*) est correcte, mais cette simplification serait impossible si l'objet était « *fake praise* » (car « *fake praise* » n'est pas un sous-ensemble de « *praise* », mais bien autre chose).

En pratique, SIE produit des résultats plutôt étranges, et n'est pas très performant. Le programme renvoie fréquemment un grand nombre d'extractions très redondantes – par exemple (*John ; gives ; praise*) **et** (*John ; gives ; true praise*), difficiles à évaluer. L'approche par modes plus ou moins verbeux de MinIE serait une amélioration. Le système produit techniquement des scores de confiance pour chaque extraction, mais en réalité ceux-ci sont pratiquement toujours égaux à 1, et n'apportent rien. SIE manque fréquemment des informations qui semblent faciles à extraire, comme dans la première phrase de la figure 2.1.

2.1.5 Autres systèmes

WOE (Wu et Weld, 2010) utilise les infoboîtes de Wikipédia comme données d'amorçage (inspirant Ollie), faisant correspondre les paires clef-valeur avec des phrases de l'article, pour ensuite apprendre des patrons d'extraction à partir des phrases. Une version du système (WOE^{pos}) fonctionne au niveau des parties du discours, l'autre ($\text{WOE}^{\text{parse}}$) au niveau des dépendances syntaxiques.

R2A2 (Etzioni et al., 2011) représente l'ajout d'un module d'apprentissage de

détection des arguments, ArgLearner, pour améliorer Reverb.

Élaboré sur la base de Wanderlust (Akbik et Bross, 2009), KrakeN (Akbik et Löser, 2012) est le premier système conçu pour capturer l'entièreté des arguments d'une relation, produisant des faits d'arité variable, sur la base de règles écrites à la main et de l'arbre syntaxique. De même, EXEMPLAR (Mesquita et al., 2013) détecte les marqueurs de relations à l'aide de règles, et trouve les arguments qui y sont connectés dans l'arbre syntaxique.

PropS (Stanovsky et al., 2016) propose de transformer (par des règles) l'arbre de dépendances syntaxiques en une représentation de la phrase plus sémantique, qui reflète plus explicitement la structure des propositions exprimées dans la phrase. Il est alors facile d'extraire les faits des propositions. Similairement, PredPatt (Zhang et al., 2017) extrait des structures prédicat-argument à partir de règles construites sur l'analyse en « dépendances universelles »[9], en faisant apparaître un lien de dépendance particulier, `ARG`, entre les mots de tête des arguments et du prédicat dans le graphe de la phrase. Grâce aux dépendances universelles, PredPatt fonctionne de façon transparente dans les autres langues que l'anglais.

Encore plus récemment, Graphene (Cetto et al., 2018) distingue les clauses, puis les syntagmes essentiels, de ceux qui n'apportent pas d'information cruciale, et identifie les relations rhétoriques qui les lient, en particulier si une clause représente le contexte d'une autre. Une fois les faits fondamentaux et les clauses accessoires identifiés, l'extraction de relations se fait de façon relativement simple.

2.1.6 Supervision distante

La manière traditionnelle de faire de l'extraction d'information consiste à limiter l'extraction à quelques dizaines de relations prédéterminées (par exemple « *born-in* », « *employed-by* », etc.), généralement avec une sémantique bien établie, et un ensemble d'entrainement établi de paires d'entités qui entretiennent cette relation. Par exemple, les relations d'intérêt pour l'épreuve de population de base de connaissance – *Knowledge Base Population* (KBP) – de la compétition TAC

[9] *Universal Dependencies*, de Marneffe et al. (2014)

en 2013 sont listées en figure 2.5. Les relations présentes dans les infoboîtes de Wikipédia[10] sont un autre bon exemple. Des extracteurs spécifiques à chaque relation sont entrainés sur la base de données d'entrainement (des exemples de phrases contenant des exemples de la relation, et des exemples d'entrainement négatifs).

Étant donné que des données d'entrainement annotées manuellement sont coûteuses à obtenir, la plupart des approches proposées ces dernières années suivent l'approche de *supervision distante* proposée par Mintz et al. (2009), à savoir produire automatiquement un plus grand ensemble de données d'entrainement à partir des petites données de référence disponibles. Pour une relation établie entre deux entités, la première hypothèse faite est que toute phrase contenant ces deux entités exprime la présence de cette relation entre ces entités.

Pawar et al. (2017) donnent une vue d'ensemble de l'état de l'art en extraction de relations (entre entités nommées, et pour un ensemble de relations prédéterminé). Leur section 6 se consacre à l'EIL, et la section 7 à la supervision distante.

Typiquement, les données de référence sont présentes dans des bases de connaissances comme Freebase ou DBPedia, et les phrases d'entrainement automatiquement détectées proviennent de corpus journalistiques.

Une fois toutes les phrases d'entrainement collectées (exprimant une relation donnée entre deux entités ou non selon que l'exemple est positif ou négatif), on entraine des extracteurs à reconnaitre la présence de la relation sur la base de caractéristiques de ces phrases. Il y a donc un ensemble d'entrainement et un modèle par relation.

Depuis l'adoption généralisée de ce modèle d'extraction, on a observé que l'hypothèse de départ était approximative, et que les données d'entrainement ainsi générées étaient assez bruitées. Concrètement, une phrase peut mentionner deux entités sans nécessairement les mettre en relation ; ou bien plusieurs relations peuvent relier une même paire d'entités, sans qu'on sache laquelle une phrase exprime.

Les recherches plus récentes ont travaillé à mitiger ce bruit, par exemple (Roth et al., 2014), et de modéliser les différentes relations pouvant coexister entre une

[10] `https://fr.wikipedia.org/wiki/Aide:Infobox`

Person	Organization
per:alternate_names	org:alternate_names
per:date_of_birth	org:political_religious_affiliation
per:age	org:top_members_employees
per:country_of_birth	org:number_of_employees_members
per:stateorprovince_of_birth	org:members
per:city_of_birth	org:member_of
per:origin	org:subsidiaries
per:date_of_death	org:parents
per:country_of_death	org:founded_by
per:stateorprovince_of_death	org:date_founded
per:city_of_death	org:date_dissolved
per:cause_of_death	org:country_of_headquarters
per:countries_of_residence	org:stateorprovince_of_headquarters
per:statesorprovinces_of_residence	org:city_of_headquarters
per:cities_of_residence	org:shareholders
per:schools_attended	org:website
per:title	
per:employee_or_member_of	
per:religion	
per:spouse	
per:children	
per:parents	
per:siblings	
per:other_family	
per:charges	

Figure 2.5 – **Relations d'intérêt pour la tâche de population de base de connaissance de TAC en 2013.** Une partie des relations prennent une valeur unique, tandis que l'autre partie accepte une liste de valeurs. Les valeurs de ces relations sont à remplir pour un certain nombre d'entités fournies dans la compétition.

paire d'entités, notamment MultiR (Hoffmann et al., 2011) et MIML-RE (Surdeanu et al., 2012).

L'atelier de population de base de connaissance KBP à la conférence TAC (*Text Analysis Conference*) a popularisé la tâche d'extraction de relations. Ji et Grishman (2011) et Surdeanu (2013) passent en revue la tâche, les participants et les approches, qui utilisent la supervision distante pour la plupart. Notablement, en 2013, une équipe (Soderland et al., 2013) utilise un système d'EIL avec des adaptations mineures au format de la tâche, et obtient des résultats intéressants (précision élevée, faible rappel).

Significativement, les données qui résultent de cette conférence permettent par la suite l'évaluation de nombreux systèmes destinés à des tâches connexes. Le système d'EIL de Stanford (Angeli et al., 2015b), en particulier, bénéficie de travaux sur l'apprentissage actif menés pour la campagne KBP et des annotations résultantes (Angeli et al., 2014, 2015a).

2.1.7 NELL

Le système NELL (Carlson et al., 2010; Mitchell et al., 2015, 2018) est original. Les auteurs mettent l'accent sur l'apprentissage permanent (*never-ending learning*), accumulant des informations apprises, ainsi que des règles d'extraction progressivement plus riches, avec l'objectif de converger progressivement vers une base de connaissances riche, idéalement sans perdre de cohérence au fil des époques. Une validation manuelle d'un petit nombre de faits à chaque itération de l'algorithme d'extraction est en place pour limiter la dérive du système.

Ce n'est pas véritablement de l'EIL, parce que les relations d'intérêt étaient spécifiées au départ (55 relations binaires et 123 catégories unaires), mais il est inclus dans la structure de pouvoir apprendre de nouveaux prédicats au fur et à mesure des itérations. Aujourd'hui, le système inclut 461 relations binaires (par exemple `VilleSituéeDansPays(.,.)` ou `MusicienAppartientÀGroupeDeMusique(.,.)`, et 293 catégories (`Entreprise(.)` ou `Monarque(.)`)[11].

[11]La base est consultable à `http://rtw.ml.cmu.edu/rtw/kbbrowser/`

Différents modules (CPL, SEAL, RL, CMC, PRA, Neil, LE, OpenEval, etc.) utilisent des approches variées pour produire des faits candidats à partir du gigantesque corpus d'entrée (le Web), et un module d'intégration des connaissances croise ces candidats pour valider les informations obtenant le plus de support indépendant. Idéalement, les erreurs des différents modules ne sont pas corrélées, et chacun diminue le taux d'erreur du système complet.

En pratique, entre 60 et 80% des extractions de NELL sont correctes, selon les méthodes d'évaluation. Par exemple, outre le pays extrait correctement, NELL considère que « `canada` » est une entreprise[12] dans le secteur des assurances – `CompanyEconomicSector(canada, insurance)` – en concurrence avec les entreprises Federal Express (Fedex) et USPS (*US Postal Service*). À propos du pays, un des faits en lesquels NELL a peu confiance est que `Canada` est une `VilleAussiConnueSousLeNomDe Columbia` (et aussi sous le nom de `Queens`, et `Windsor`, et `Washington DC`). Ce dernier prédicat, ainsi que « `CountryAlsoKnownAsCountry` », semblent être le produit de dérives sémantiques, et la plupart des paires d'entités liées par cette relation sont fausses comme (Mongolie ; aussi connue comme ; Chine), ou inutiles comme (Autriche ; aussi connue comme ; République). Tout de même, les nombreux modules proposant des faits candidats sont autant de façons indépendantes d'extraire l'information de façon très faiblement supervisée, et une source d'inspiration pour la recherche en EIL.

2.2 Évaluation de l'EIL

Peu de travaux s'intéressent à l'évaluation de l'EIL (d'où mon projet). Les premiers systèmes sont évalués en précision et rendement, suivant un protocole ad hoc, manuel et subjectif, heureusement toujours le même. Deux ensembles de données ont depuis été produits, qui permettent d'évaluer les différents systèmes d'EIL automatiquement, de façon raisonnablement objective. Je présente tour à tour l'évaluation « classique », et ces deux jeux de données.

[12] aussi désignée par Canada, CAnada, et CANADA

À partir de 2016, les recherches s'orientent vers d'autres critères de performance, comme l'extraction de relations n-aires (notamment KrakeN (Akbik et Löser, 2012) et Exemplar (Mesquita et al., 2013)), la minimalité (CSD-IE (Bast et Haussmann, 2013), MinIE), et l'analyse du contexte (Ollie, NestIE (Bhutani et al., 2016), OpenIE 4, MinIE). Ces éléments sont importants et une évaluation de qualité doit en tenir compte. Toutefois, peu d'efforts sont faits afin de normaliser la mesure de ces caractéristiques, problème auquel les directives de WiRe s'attaquent (voir chapitre 4).

2.2.1 Évaluation habituelle de l'EIL

Les systèmes d'EIL – par exemple Reverb (Fader et al., 2011), R2A2 (Etzioni et al., 2011), Ollie (Mausam et al., 2012), ClausIE (Del Corro et Gemulla, 2013), CSD-IE (Bast et Haussmann, 2013), NestIE (Bhutani et al., 2016), MinIE (Gashteovski et al., 2017) – sont typiquement évalués de la façon suivante : étant donné un corpus textuel, les différents systèmes comparés procèdent à l'extraction d'information sur le corpus. Les faits extraits sont agrégés, et un sous-ensemble aléatoire des résultats obtenus est annoté manuellement comme étant, pour chaque triplet, « correct ou incorrect ». Chaque système se voit ainsi attribuer un score de précision (la proportion des triplets qu'il a produits qui sont corrects), et un rendement (*yield*), le nombre de bons triplets obtenus au total (par produit de sa précision avec le nombre total de triplets extraits). Une courbe de précision/rendement (en fonction du score de confiance des extractions) est produite, et la mesure de l'aire sous la courbe rassemble les deux métriques de précision et rendement en une seule.

Étant donné que la plupart des extractions sont obtenues avec une précision d'environ 70 à 80%, les premières générations successives d'extracteurs ont surtout progressé du côté du rendement :

> « Ollie trouve 4,4 fois plus d'extractions correctes que Reverb, et 4,8 fois plus que $\text{WOE}^{\text{parse}}$, avec une précision d'environ 0,75. »[13]

[13] traduction par mes soins de (Mausam et al., 2012)

« ClausIE produit 1,8-2,4 fois plus d'extractions correctes qu'Ollie. »[14]

Reverb et ClausIE, qui sont évalués suivant cette méthode, mettent à disposition[15] leurs annotations manuelles : la liste des triplets extraits par l'un ou l'autre des systèmes, ainsi que le jugement binaire « correct ou non » de l'annotateur humain.

Ce protocole d'évaluation est raisonnable et naturel, mais présente deux problèmes majeurs. Le premier est l'évaluation du rappel. Sans référence a priori, il est impossible de déterminer combien d'informations sont omises par l'ensemble des systèmes.

Le deuxième est la relative subjectivité de ce qui est annoté comme correct.

Comme indiqué plus haut, la précision est mesurée en prélevant des extractions au hasard parmi les productions du système, et en les annotant manuellement comme valides ou incorrectes. Généralement, on considère qu'une extraction est correcte si elle est « impliquée par la phrase » :

> « Deux annotateurs ont étiqueté les extractions comme correctes si la phrase affirmait ou impliquait que la relation était vraie. »[16]
>
> « Nous avons aussi demandé aux annotateurs d'être flexibles par rapport à la coréférence et à la résolution d'entités ; par exemple, une proposition telle que ("il" , "a" , "bureau"), ou une de ses formes non lemmatisées, est considérée comme correcte. »[17]

Dès lors, un certain nombre d'extractions techniquement correctes (bel et bien impliquées par la phrase), mais vides de sens, sont évaluées positivement, sans véritablement apporter de valeur. La figure 2.6 montre des exemples de faits jugés

[14](Del Corro et Gemulla, 2013)

[15]Voir `http://reverb.cs.washington.edu/` et `http://resources.mpi-inf.mpg.de/d5/clausie/`

[16]*"Two annotators tagged the extractions as correct if the sentence asserted or implied that the relation was true."* (Mausam et al., 2012)

[17]*"We also asked labelers to be liberal with respect to coreference or entity resolution ; e.g., a proposition such as ('he', 'has', 'office'), or any unlemmatized version thereof, is treated as correct."* (Del Corro et Gemulla, 2013)

Phrase : In response, a group of Amherst College students held a patriotism rally in October, reciting the Pledge of Allegiance.
Extraction : (a group of Amherst College students ; held ; a patriotism rally)

Phrase : That's a lot of maybes in a sport where the right thing seldom happens, but [given X, Y should Z if he wants T].
Extraction : (That ; is ; a lot of maybes)

Phrase : The scandal has now forced resignations at Japan's fourth-largest bank and three of Japan's Big Four brokerages.
Extraction : (Japan ; has ; fourth-largest bank)

Phrase : This leads to one of two inescapable conclusions : Either the president reads BioScope or I got lucky.
Extraction : (the president ; reads ; BioScope or I got lucky)

Figure 2.6 – **Exemples d'extractions jugées correctes par des évaluations manuelles passées que je trouve plus ou moins problématiques**, provenant respectivement de Reverb et de ClausIE. Dans la première, même si l'information est vraie en elle-même, le contexte de la phrase précédente (« *In response* ») manque pour lui donner son sens. Dans le deuxième, le premier argument « *That* » est plus que flou, sans référent apparent. Dans la troisième, l'information est vraie, ainsi qu'elle le serait pour la plupart des pays. L'information serait mieux extraite à un niveau d'abstraction plus élevé : « les pays ont des banques ». Le dernier fait ne reflète simplement pas le sens de la phrase, probablement par une erreur humaine.

corrects par les évaluations de Reverb et ClausIE, qui posent problème du point de vue de l'informativité. Mes travaux des chapitres 3 et 4 visent à construire des références tenant compte de l'informativité des extractions. Une difficulté supplémentaire réside dans le manque de discrimination des scores attribués par cette méthode : toutes les versions approximativement acceptables d'une relation idéale sont jugées valides, avec le même score que la relation idéale. Or, la plupart des extractions sont plus ou moins imparfaites (voir en particulier la première phrase figure 2.1). Un jugement plus précis permettrait de quantifier ces imperfections, et de valoriser un système faisant peu de telles erreurs.

2.2.2 Extraction de relations

Mesquita et al. (2013) présentent une comparaison expérimentale de plusieurs systèmes, sur une tâche très semblable à l'EIL : l'extraction de relations (uniquement – *Open Relation Extraction*). Dans leur configuration, seules les entités nommées peuvent être arguments de relations, et la tâche consiste à reconnaitre la relation qui lie les paires d'entités présentes dans le texte. Leur étude se concentre sur le compromis entre vitesse de traitement et profondeur de l'analyse linguistique (allant de pair avec la précision et la qualité des résultats). Ils établissent une référence de 662 relations binaires, présentes à travers 1100 phrases provenant de trois sources (le Web, le *New York Times*, et le Penn Treebank). Ils étiquettent également 222 phrases du *New York Times* avec des relations n-aires, à raison d'une par phrase. Enfin, ils génèrent un corpus d'annotations automatiques, sur 12 000 phrases.

Au-delà des arguments qui sont des entités nommées, leurs annotations consistent en un mot « déclencheur » (indiquant la relation), entouré par une fenêtre de mots « autorisés » (pouvant être considérés comme faisant partie de la relation). De façon à comparer équitablement les systèmes d'EIL avec les autres systèmes adaptés à l'extraction de relations, les auteurs remplacent les entités visées par des entités saillantes (« Europe » et « Asie ») pour éviter des erreurs d'identification des arguments.

Ils soulignent abondamment la difficulté qu'il y a à construire un banc d'évaluation commun à plusieurs méthodes sans être partial à aucune : pour commencer, différents annotateurs définissent les entités nommées différemment. Par exemple, un premier jeu d'annotations que les auteurs réemploient intègrent les appositions dans les entités nommées, ce que les auteurs ne font pas (« *the Quick & Reilly discount brokerage firm* » contre seulement « *Quick & Reilly* »).

Ensuite, la question de la coréférence reste ouverte : dans « Anne est venue, et elle a vu Bertrand. », la plupart des auteurs sont satisfaits de l'extraction (*elle ; a vu ; Bertrand*), alors que Mesquita, Schmidek et Barbosa considèrent que la relation correcte est (*Anne ; a vu ; Bertrand*).

Enfin, les différents systèmes évalués ont été construits avec des objectifs bien distincts. PATTY (Nakashole et al., 2012) et les travaux subséquents (Grycner et Weikum, 2014; Grycner et al., 2015) se concentrent sur l'extraction de patrons de relations, et mettent l'accent sur la signature des patrons (les types des entités qu'ils mettent en relation). Tout groupe nominal peut être un argument pour les systèmes d'EIL tandis que SONEX (Merhav et al., 2012) ne considère que les entités nommées.

Bien que les tâches soient similaires, imposer aux arguments d'être des entités nommées restreint l'extraction d'information aux seuls faits les plus saillants du texte. Sans restreindre les arguments, on peut extraire 6 informations par phrase en moyenne (voir chapitre 4), contre 0.6 dans ce jeu de données. Par ailleurs, certaines relations n'ont pas de déclencheur explicite, par exemple (*Paris ; [is in] ; France*) dans « *Chilly Gonzales lived in Paris, France* ».

2.2.3 Banc d'essai QA-SRL-OIE

Remarquablement, Stanovsky et Dagan (2016) transforment le jeu de données QA-SRL de He, Lewis et Zettlemoyer (2015) en un vaste banc d'essai pour l'EIL. Précisément, pour chaque prédicat annoté dans QA-SRL, ils produisent un fait de

référence qui exprime toutes les réponses aux questions à propos de ce prédicat[18].

Par exemple, pour la phrase « *Investors are appealing to the SEC not to limit their access to information about stock purchases and sales by corporate insiders* », QA-SRL identifie deux prédicats (« *appealing* » et « *limit* ») et cinq questions : « *who are appealing to something ?* », « *who are someone appealing to ?* », « *what are someone appealing ?* », « *what might not limit something ?* » and « *what might not someone limit ?* », avec une réponse par question. Les deux informations à extraire correspondantes sont donc (*Investors ; appealing ; not to limit their access to information about stock purchases and sales by corporate insiders ; to the SEC*) et (*the SEC ; might not limit* ; *their access to information about stock purchases and sales by corporate insiders*).

Le jeu de données résultant porte sur 3200 phrases provenant pour un tiers du *Wall Street Journal* et pour deux tiers de Wikipédia, et est librement distribué par ses auteurs[19]. 7710 prédicats sont couverts, qu'interrogent 18910 questions, résultant en 10359 informations de référence à extraire.

Ce travail fait un grand pas dans la bonne direction, mais souffre de quelques lacunes.

Tout d'abord, une grande force du banc d'essai est sa large couverture. L'évaluation automatique sur un grand nombre de prédicats annotés est objective et limite le biais envers l'un ou l'autre des systèmes d'extraction. L'exhaustivité des annotations, toutefois, laisse à désirer, de par la dépendance des données de QA-SRL.

Seuls les prédicats verbaux sont annotés dans QA-SRL, tandis que l'EIL vise à extraire *toutes* les informations, y compris les prédicats nomimaux et adjectivaux. Par exemple, dans « *Elsa is eight. The younger Anna couldn't do magic.* », on trouve l'information (*Anna ; is younger than ; Elsa*), médiée par l'adjectif « *younger* ». Ou encore, dans « *The meeting between Ashley and Elysha didn't last long* » on trouve

[18]Encore plus précisément, comme les questions peuvent avoir plusieurs réponses, ils produisent un fait de référence pour chaque **élément du produit cartésien** des réponses à toutes les questions.

[19]`http://u.cs.biu.ac.il/~nlp/resources/downloads/`

l'information (*Ashley ; met with ; Elysha*), portée par « *meeting* ».

Concrètement, dans la phrase « *However, Paul Johanson, Monsanto's director of plant sciences, said the company's chemical spray overcomes these problems and is "gentle on the female organ".* », QA-SRL annote les deux verbes « *said* » et « *overcomes* », et la référence de Stanovsky et Dagan (2016) contient les extractions (*Paul Johanson* ; *said* ; *the company's chemical spray overcomes these problems and is "gentle on the female organ.*") et (*the company's chemical spray* ; *overcomes* ; *these problems*), ce qui omet les informations importantes (*the company's chemical spray* ; *is* ; *"gentle on the female organ"*), et (*Paul Johanson* ; *is* ; *Monsanto's director of plant sciences*).

Un autre problème est que certains mots furent ajoutés lors de la transformation de l'annotation en rôles sémantiques des phrases sources vers les questions/réponses (SRL → QA), conservés pendant la transformation en référence pour l'EIL (QA → EIL), et font donc partie de la référence, sans faire partie des phrases d'origine. Dans l'exemple « *Investors are appealing to the SEC not to limit [...]* » cité plus haut, la relation extraite (*the SEC* ; **might not limit** ; *their access to information [...]*) ne provient pas directement de la phrase. Et si dans ce cas-ci l'emploi d'un verbe modal exprime correctement l'information, l'emploi répété de cette tournure affaiblit les faits ainsi exprimés. Par exemple, le triplet inutile (*a manufacturer ; might get ; something*) est généré à partir de la phrase « *... and if a manufacturer is clearly trying to get something out of it ...* », avec le même ajout du modal « *might* ».

Enfin, la procédure d'évaluation manque de robustesse. En utilisant le code mis à disposition par les auteurs[20], il est possible de dépasser tous les systèmes évalués avec un système trivial.

La fonction de score ne pénalise pas les extractions indûment longues, ni le fait de placer des parties de l'objet dans la relation ou vice versa. Ainsi, si $w_0 w_1 ... w_n$ est une phrase d'entrée, le programme qui « extrait » $(w_0; w_1; w_2 ... w_n)$,

[20] `https://github.com/gabrielStanovsky/oie-benchmark` — la fonction de score a été changée depuis sa description dans l'article de Stanovsky et Dagan (2016). La fonction de l'article souffre de problèmes semblables.

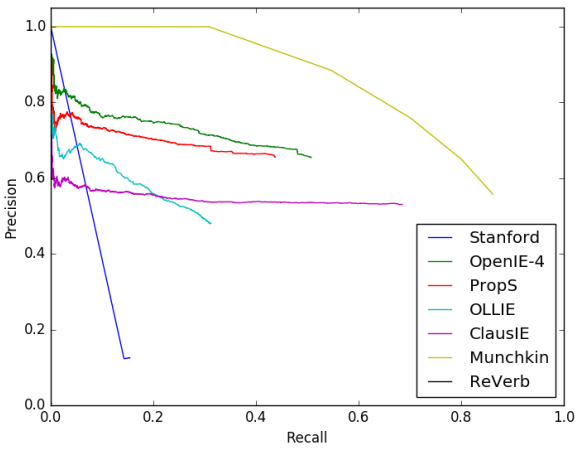


Figure 2.7 – Les métriques de performances doivent prendre en compte la précision de l'extraction des segments. Le programme Munchkin est un script de 25 lignes qui renvoie des variantes de la phrase complète (avec des scores de confiance décroissants). Il n'est pas pénalisé par le script d'évaluation du banc d'essai, et atteint artificiellement des performances extravagantes.

$(w_0; w_1 w_2; w_3 ... w_n)$, etc., obtiendra un excellent score. J'ai implémenté un tel système et l'ai évalué avec le code d'évaluation disponible, surpassant ainsi effectivement tous les systèmes réels (voir figure 2.7). Je développe une fonction de score visant à régler ces problèmes dans le chapitre 4.

2.2.4 Autres travaux sur l'évaluation

Schneider et al. (2017) développent l'outil RelVis[21] et évaluent quatre systèmes (ClausIE, OpenIE 4, SIE et PredPatt[22]) sur les deux jeux de données détaillés plus haut. Ils emploient deux méthodes pour mettre en correspondance les extractions produites avec celles de référence, d'« inclusion » et « inclusion stricte ». Dans les deux cas, l'information produite doit inclure l'information de référence ; dans le cas strict, chaque partie (sujet, relation, objet) doit inclure la partie correspondante, sans mélange. Comme Stanovsky et Dagan (2016), leur fonction d'évaluation ne

[21] Au moment de l'écriture, leur code est annoncé, mais pas disponible, à `https://github.com/schmaR/relvis`.

[22] (White et al., 2016)

semble pas pénaliser les systèmes qui renvoient de trop longues extractions.

Les auteurs analysent aussi qualitativement les erreurs des systèmes, les répertoriant en 5 catégories : erreurs de segmentation, problèmes de rappel, extractions non-informatives, extractions fausses et extractions redondantes. Les erreurs de segmentation sont de loin les plus fréquentes. Globalement, il est difficile de tirer des leçons des résultats obtenus par cette étude : les différents systèmes brillent davantage sur les différents corpus avec une énorme variabilité, pour des raisons circonstancielles. Les nombres d'erreurs précises détectées sont très petits, et les classes d'erreurs assez larges.

Groth et al. (2018) évaluent OpenIE 4 et MinIE sur l'EIL restreinte au domaine scientifique, par « production participative » (*crowdsourcing*). Ils reprennent bon nombre des points de Schneider et al. (2017) et en discutent à l'adresse d'un plus large public. L'évaluation est faite sur deux ensembles d'environ 200 phrases, un corpus encyclopédique et l'autre scientifique. Les participants sont invités à juger si les faits extraits sont ou non impliqués par la phrase, comme dans le premier protocole d'évaluation décrit plus haut. L'article conclut que les systèmes d'EIL font moins bonne figure sur le corpus scientifique que sur le corpus général.

2.3 Conclusion

Peu d'attention a été portée à l'évaluation de l'EIL. La plupart des systèmes sont évalués *in fine* par leurs auteurs, suivant un protocole laissant une large place à la subjectivité des juges humains. Les critères d'évaluation des extractions produites ne sont pas définis à l'avance, et doivent être réinventés par chaque nouveau groupe d'annotateurs. Plus important encore, les extractions qui ne sont *pas* produites sont passées sous silence, et impossibles à quantifier.

Les deux bancs d'évaluation existants souffrent d'importantes limitations, notamment en termes de couverture. Il est difficile de comparer les productions d'un système à une vérité de référence pour de multiples raisons (notamment celles qui rendent imparfaite la métrique BLEU utilisée en traduction), *a fortiori* étant donné

que ce qui est attendu d'un système n'est pas très clairement défini (par exemple la résolution des anaphores, et la définition et la granularité des entités, sont à harmoniser).

Dans la partie suivante, je propose deux bancs d'évaluation supplémentaires, établis dans le but premier d'être capables de mesurer le rappel des systèmes. Il est crucial, pour une évaluation définitive d'un système d'EIL (dont la mesure ne soit pas dépendante par comparaison du progrès d'autres systèmes), que la référence pour une tâche donnée soit *exhaustive*. Cela implique de déterminer à l'avance quels types d'informations on s'attend à extraire, et à l'inverse quels éléments à ignorer, le cas échéant.

CHAPITRE 3

ÉVALUATION DE L'EIL BASÉE SUR LES QUESTIONS-RÉPONSES

William Léchelle et Philippe Langlais. An Informativeness Approach to Open IE Evaluation. Dans Alexander Gelbukh, éditeur, *CICLing*, Lecture Notes in Computer Science. Springer, Springer, 2016.

Best Verifiability, Reproducibility, and Working Description award, 3rd place.

Contexte

L'idée de ce projet, en 2013, partait d'un constat simple : on ne disposait pas de façon directe et automatique d'évaluer les systèmes d'EIL. Comme expliqué dans l'état de l'art, les évaluations précédentes demandaient un lourd effort manuel et difficilement reproductible, pour chaque mesure de performance. On aimerait pouvoir modifier la méthode d'EIL, relancer l'extracteur sur le corpus, et évaluer **automatiquement** la production du système, en appuyant simplement sur un bouton[1].

Par ailleurs, un problème saillant avec les productions des systèmes de l'époque (Reverb et Ollie) était la forte proportion de triplets incompréhensibles, qui ne faisaient pas de sens tels que produits, hors contexte ; par exemple (*the Arab world ; is ; a rich prize*). Outre l'automatisation, le deuxième objectif majeur était d'évaluer les extractions en tenant compte de leur informativité. Le scénario d'application sous-jacent est que chaque triplet correctement extrait devrait pouvoir être intégré dans une base de connaissances générale et consister par lui-même en un élément d'information. Les extractions ne répondant pas à ce critère devraient être exclues,

[1] Idéalement, on aimerait que l'évaluation soit à même de distinguer de petites modifications du système, et de les juger en conséquence, mais cet élément ne sera développé qu'au chapitre suivant.

détectées comme inutiles.

Nous avons donc cherché la façon la plus simple d'évaluer automatiquement les sorties de ces systèmes. Comme soutenu dans l'article, la façon la plus naturelle de juger si l'information extraite du texte est correcte et complète consiste à l'interroger avec des questions. Une méthode d'extraction qui permet de répondre à davantage de questions portant sur le texte est une meilleure méthode. Cette méthodologie d'évaluation requiert une étape de réponse automatique à des questions, que nous avons essayé de garder la plus simple possible. J'ai donc implémenté un système basique de questions-réponses automatique, qui constitue de fait la « fonction d'évaluation » de cette méthode d'évaluation (qui fait le lien entre les productions du système et la référence établie).

Contributions

L'idée d'évaluer les sorties des extracteurs à l'aide d'une tâche de questions-réponses vient de Philippe Langlais. J'ai annoté les phrases et construit la ressource, en m'appuyant sur les extractions et jugements manuels existants publiés par Del Corro et Gemulla (2013). J'ai ensuite implémenté le système élémentaire de questions-réponses (décrit en section 3.4.2), et expérimenté pour en ajuster les paramètres. J'ai mesuré la performance des extracteurs sur la ressource, incluant le retour sur mes jugements a priori de la performance attendue des extracteurs. J'ai écrit l'article final, que mon directeur de recherche a relu et corrigé. L'article a été publié à la conférence CICLing en avril 2016, où je l'ai présenté.

Impact

Je propose la première procédure d'évaluation automatique des systèmes d'EIL : mécaniquement reproductible, donc objective ; et qui permet de quantifier le rappel des extractions de façon absolue. Pour la première fois, on tente de mettre un chiffre sur la quantité d'information que les extracteurs omettent : en étant permissifs, l'ensemble des systèmes capturent environ 40% de l'information exprimée

sur laquelle on peut poser des questions hors contexte (sans compter l'information qui n'a de sens qu'en contexte).

Les résultats de l'évaluation confirment que le système le plus récent à l'époque, ClausIE, est le plus performant, notamment au niveau du rappel. ClausIE avance une réponse à presque 70% des questions, contre 40% pour Reverb et Ollie. Lorsqu'une réponse est proposée, au moins un des systèmes détient la bonne réponse environ une fois sur deux.

Le code et les jeux de données d'évaluation étant publiés en accompagnement de l'article, d'autres chercheurs peuvent directement s'appuyer sur la méthode développée pour évaluer leurs propres systèmes. Cette transparence et les efforts déployés pour rendre l'utilisation du code la plus facile possible ont valu à l'article un prix – *Best Verifiability, Reproducibility, and Working Description award, 3rd place.*

L'idée d'évaluer l'EIL par une tâche de questions-réponses a été explorée de façon indépendante par Fader et al. (2014) et plus tard par Qiu et al. (2018).

3.1 Introduction

Open Information Extraction (OIE) – information extraction without pre-specification of relations or entities to target – seeks to extract relational tuples from large corpora, in a scalable way and without domain-specific training (Mausam et al., 2012) (Del Corro et Gemulla, 2013). Recently, there has been a trend of successful use of OIE output as a text understanding tool, for instance in (Fader et al., 2014), (Soderland et al., 2013) and (Stanovsky et al., 2015).

However, a close look to system output reveals that a large fraction of extracted facts, albeit correctly extracted from the text, are devoid of useful information. The main reason for this is lack of context : many extracted noun phrases, and facts, only have meaning in the context of their sentences. Once the source is lost, the remaining relation is empty, for factual purposes[2]. Figure 3.1 shows examples

[2]For automatic language modeling purposes, on the other hand, extracted facts are a great source of learning material, as demonstrated in (Stanovsky et al., 2015).

of uninformative facts. We discuss in Section 3.2 how state-of-the-art metrics of extraction performance fail to account for meaningless extractions.

Sentence : In response, a group of Amherst College students held a patriotism rally in October, reciting the Pledge of Allegiance.
Fact : (a group of Amherst College students ; held ; a patriotism rally)

Sentence : That's a lot of maybes in a sport where the right thing seldom happens, but [given X, Y should Z if he wants T].
Fact : (That ; is ; a lot of maybes)

Sentence : The scandal has now forced resignations at Japan's fourth-largest bank and three of Japan's Big Four brokerages.
Fact : (Japan ; has ; fourth-largest bank)

Sentence : This leads to one of two inescapable conclusions : Either the president reads BioScope or I got lucky.
Fact : (the president ; reads ; BioScope or I got lucky)

Figure 3.1 – **Extracted facts deemed correct by previous manual evaluations**, respectively from Reverb and ClausIE. In the first extraction, even though it is true in itself, crucial information from another sentence is missing to give the fact its meaning. In the second, the first argument it at best a vague idea. In the third, the fact holds true for most countries and holds little information by itself (it would be more adequate at another level of abstraction, e.g. *countries have banks*). The last extracted fact does not reflect the actual sentence meaning.

We propose an evaluation procedure for open information extractors that more tightly fits downstream user needs. The most direct usage of information is answering questions about it, so we evaluate extractors' output on their capacity to answer questions asked about the text at hand.

This procedure serves two purposes not previously addressed :

1. incorporate informativeness in the judging criterion for correct extractions ;
2. estimate recall for the task.

Section 3.3 details the evaluation methodology we follow, and the guidelines that drive our annotations. Section 3.4 presents the dataset we built and our basic automatic evaluation procedure, and Section 3.5 sets out our results.

Sentence : For the 2006-07 season, Pace played with the Nelson Giants in the New Zealand National Basketball League.
4-ary fact : (Pace ; played ; with the Nelson Giants ; for the 2006-07 season ; in the New Zealand National Basketball League)

Questions :
Who did Pace play with ?
When did Pace play with the Nelson Giants ?
In what league did Pace play with the Nelson Giants ?

Figure 3.2 – **Some facts are intrinsically n-ary, and naturally answer many questions.** An extractor capturing only binary relations would likely miss much context.

3.2 Related work

Little work has directly addressed the issue of evaluating OIE performance. Up to now, performance of extractors mostly relied on two metrics : number of extracted facts and precision of extraction. Area under the precision-yield curve is a common shorthand to summarize them both (Mausam et al., 2012). As the bulk of extractions are usually obtained with a precision in the 70-80% range, consecutive generations of extractors have mostly improved on the yield part :

> "Ollie finds 4.4 times more correct extractions than Reverb and 4.8 times more than WOE$^{\text{parse}}$ at a precision of about 0.75". Mausam et al. (2012)

> "ClausIE produces 1.8–2.4 times more correct extractions than Ollie". Del Corro et Gemulla (2013)

It is commonly lamented that absolute recall cannot be calculated for this task, because of the absence of a reference. We aim to address this issue.

Typically, precision is measured by sampling extractions and manually labeling them as correct or incorrect. As a rule of thumb, an extraction is deemed correct if it is implied by the sentence :

> "Two annotators tagged the extractions as correct if the sentence asserted or implied that the relation was true." Mausam et al. (2012)

> "We also asked labelers to be liberal with respect to coreference or entity resolution; e.g., a proposition such as ('he', 'has', 'office'), or any unlemmatized version thereof, is treated as correct." Del Corro et Gemulla (2013)

By contrast with previously employed criteria, we propose to incorporate the informativeness of extracted facts, as measured by their ability to answer relevant questions, in the judgment of their validity.

Figure 3.1 shows examples of previous facts manually labeled as correct, taken respectively from Reverb[3] and ClausIE[4]. Except for the last, they pass the standard criteria for correctness. We seek to devise an evaluation protocol that would reject such extractions on the grounds that they are not informative.

As highlighted by Akbik et Löser (2012), one major element of OIE performance is the handling of n-ary facts. Most extractors focus on binary relations, but many support n-ary relations to some extent. KrakeN (Akbik et Löser, 2012) and Exemplar (Mesquita et al., 2013) are designed towards n-ary extractions. ClausIE (Del Corro et Gemulla, 2013) supports generation of n-ary propositions when optional adverbials are present and Ollie (Mausam et al., 2012) can capture n-ary extractions by collapsing extractions where the relation phrase only differs by the preposition[5]. Though it is not our main focus, our evaluation protocol addresses this question in that n-ary facts that capture more information will answer more questions than binary facts that would leave out some of the arguments. Figure 3.2 shows an example of 4-ary fact, and the corresponding questions it answers. Information extractors will be evaluated on their ability to answer such questions.

In (Mesquita et al., 2013), the authors present an experimental comparison of several systems, over multiple datasets, on the very similar task of open relation extraction (that differs from OIE by only considering named entities as possible arguments). Their study focuses on the tradeoff between processing speed — depth of

[3] `http://reverb.cs.washington.edu/reverb_emnlp2011_data.tar.gz`

[4] `http://resources.mpi-inf.mpg.de/d5/clausie/`

[5] This was added to the distributed software since publication – see `https://github.com/knowitall/ollie`

linguistic analysis — and accuracy. Much of their discussion stresses the difficulties of building a common evaluation methodology that is fair to various methods. In particular, their Section 3.3 is a good illustration of several evaluation difficulties.

3.3 Proposed Methodology

3.3.1 Evaluation Protocol

As hinted in Figure 3.2, the evaluation protocol we propose for OIE is as follows :

> Given some input text, annotate all factoid questions that can be answered by information contained in that text, and the answers. Run the extractors on the input text. The evaluation metric is the number of questions that can be answered using only the output of each system.

The step of using extracted tuples to answer questions raises issues. Of course manual matching of extractions and questions would be most precise, but unrealistically expensive in human labor. An automatic scoring system also makes for more objective and easily replicable results, albeit less precise. Still, automatic question answering is a notoriously difficult problem that we would rather not tackle. In order to avoid this difficulty, we design our questions to be in the simplest possible form. Figure 3.3 shows a sample of our annotations. The questions are worded in very transparent ways.

We describe the automatic scoring system we use for this paper at greater length in Section 3.4.2, and release it along with our data.

3.3.2 Annotation Process

We wish to annotate all questions that can be answered by information contained in the input text, in a way that is easy to answer automatically.

Our primary goal being to evaluate OIE systems, we found useful to consider their output on the sentences at hand as a base for annotation. As stated before (Figure 3.1), many extractions are not informative by themselves. Examples of

S: Esaka and six other top executives will quit to take responsibility for 67.28 million yen in payoffs to corporate racketeer Ryuichi Koike, 54.

Q: Why will Esaka quit ?
A: to take responsibility for 67.28 million yen in payoffs to corporate racketeer Ryuichi Koike
X: 1

YQ: Will Esaka quit ?
A: Yes
X: 1

Q: What age is Ryuichi Koike ?
A: 54
X: 1

S: And he has eased up on team rules.

S: His teammates indeed loved the show.

S: Mrs. Yogeswaran was shot five times with a pistol near her Jaffna home on May 17, 1998.

Q: What was Mrs. Yogeswaran shot with ?
A: a pistol
X: 1

Q: How many times was Mrs. Yogeswaran shot ?
A: five
X: 1

S: Robert Barnard (born 23 November 1936) is an English crime writer, critic and lecturer.
Q: Who is Robert Barnard ?
A: an English crime writer
(there is no X: annotation on this Q&A pair)

Q: When is Robert Barnard born ?
A: 23 November 1936
X: 0

Figure 3.3 – **Examples of annotated sentences, with questions and answers.** Questions are worded following the original text so that the question answering step is simple to perform. Many sentences are embedded in so specific a context that they do not carry any extractable information, like the second and third sentences. Others usually yield a handful of facts. Lines are prefixed as **S**entences, **Q**uestions, and **A**nswers (**YQ** stands for yes/no questions). In our dataset, most questions (but not all) are tagged with an e**X**pected result of the Q&A system, given the extractions seen by the annotator (these are the **X**: lines – 1 if the answer will be found, 0 if it won't), for intrinsic evaluation purposes.

Sentence: While the Arab world is a rich prize in itself, Europe has been and remains the primary objective.
Extraction: (Europe ; remains ; the primary objective)
Extraction: (the Arab world ; is ; a rich prize)
Important context is missing for these extractions to have meaning.

Sentence: Wu worked as a reporter for United Press International from 1973 until 1978 when she joined WGBH-TV, Boston's public television station, as the Massachusetts State House reporter until 1983.
Extraction: (Wu ; worked as a reporter for ; United Press International)
Sentence: Daughter of the actor Ismael Sanchez Abellan and actress and writer Ana Maria Bueno, Gabriel was born in San Fernando, Cadiz ...
Extraction: (Gabriel ; was born in ; San Fernando)
"Wu" is at the threshold of being sufficiently determined to ask questions about. "Gabriel" being a common first name, it is just on the other side of our threshold.

Sentence: He was born in New York and died at Livonia, Michigan.
Extraction: (He ; was born in ; New York)
Sentence: He consults his family doctor for solution.
Sentence: Had I known then what I know now, I might have argued for a different arrangement.
Coreference resolution is key in many sentences.

Figure 3.4 – Ambiguity due to the loss of context is the main issue for annotation.

correctly extracted facts that cannot answer real-world questions are shown in the first sentence of Figure 3.4.

Given OIE output, all extracted relations that are factually informative are asked about, i.e. a question is added that is expected to be answered by it. Then, we also ask all other questions that can be answered with information contained in the sentence, without being overly specific.

One could argue that the annotator seeing the output of the systems introduces a bias in the evaluation procedure. We do not believe this to be the case. As all OIE output is considered, the annotator is blind to specific extractors and the resulting dataset is fair to all systems. If proper attention is paid to asking questions about facts that are not correctly extracted, then the measure of recall isn't biased towards technology performance either. In favor of using the information, it is easier to ask about useful extracted facts (e.g. in the very terms of the fact), which means all due credit is given to systems' successes. Additionally, annotation so helps reflecting on OIE capabilities and limitations.

3.3.3 Dealing with Ambiguity

The major issue in our search for informativeness is ambiguity due to lack of context. It is a problem on various levels, as exemplified in Figure 3.4.

In some cases, each element of the relation is understandable, but the whole meaning cannot be understood out of context — for instance (*"the Arab world is a rich prize"*) and (*"Europe remains the primary objective"*) in Figure 3.4. When we cannot understand what the text at hand is about, we naturally do not annotate anything. In the last sentence of Figure 3.4, (*I ; might have argued for ; a different arrangement*) similarly relates to a lost specific context.

Often, and this is the key difficulty, the arguments themselves only make sense in context. In *"the scandal forced resignations . . . "* (Figure 3.1), what *"the scandal"* refers to maybe was obvious in the news context of the time, but is lost to us.

Therefore, there is a somewhat arbitrary line to draw regarding the amount of context we can assume a potential user has in mind when asking questions.

Among the many shades of ambiguity, *Albert Einstein* and *New York City* are utmostly self-explanatory[6]. *"I"*, to the contrary, is completely dependent on its particular utterance, and a clear definition of its reference will most often pertain to the metadata of the document at hand.

In between, consider *"Wu"* in the second sentence of Figure 3.4. We consider this name to be on the threshold of acceptable ambiguity for our purpose. Using context, we can trace her to be Janet Wu, an American television reporter from the Boston area[7]. Also consider *"Gabriel"*[8] in the following sentence, which we consider to be on the other side of the threshold.

We envision two possible policies regarding context requirements.

What we did is the following : assume there is no word sense disambiguation. Every argument phrase used in our dataset should refer to a single entity. When an input sentence is about the most common use of its words (like *Europe* for the continent), we consider it well defined and annotate it, using its words in those senses. When sentences are about less common senses of their words (like *Gabriel* for Ruth Gabriel or *New York City* for the video game), we do not annotate them, on the grounds that further processing would be required to make use of such annotations, that is outside the scope of this work. It is normal for OIE to lose the context of extractions, and informativeness judgment calls shall take that into account.

Hence, by the fact that there is only one *"Wu"* in our corpus, it is a valid designation for an entity. *"Gabriel"*, on the other hand, is more often a common first name than refers to Ruth Gabriel, so we would not ask about her.

A looser policy would be to assume that the user that asks questions has in mind the same context as the writer of the original text. Within that view, some user may ask, *"did the scandal force resignations ?"*, having in mind the Japanese

[6]although they could refer to an American actor and a 1984 Atari video game.

[7]The sentence is from `https://en.wikipedia.org/wiki/Janet_Wu_(WCVB)`. Incidentally, `https://en.wikipedia.org/wiki/Janet_Wu_(WHDH)` also is an American television reporter who worked in the Boston area. We consider this to be an ironic coincidence, but stand by our arbitrary line.

[8]Ruth Gabriel is a Spanish actress.

banking public embarrassment of Figure 3.1, or *"what is the primary objective ?"*, thinking of them who sell to the Arab world in Figure 3.4.

3.3.4 Coreference

In a way, coreferential mentions are the extreme case of the ambiguity-without-context problem just mentioned.

Currently available OIE systems don't resolve coreferential mentions, and a significant portion of extracted facts have "it", "he" or "we" as arguments. As such, we consider that these extractions cannot answer questions, as the reference of such mentions is lost. On the Wu sentence, systems extract (**she** ; *joined ; WGBH-TV as the Massachusetts State House reporter*), but we lack the means of answering, *"did Wu join WGBH-TV ?"* with that fact.

In our dataset, we ask such questions that would require coreference to be resolved at extraction time when it happens inside a sentence. As all sentences in our dataset were randomly picked from documents, all the cross-sentential references are lost, and we do not ask questions about them. In Figure 3.4, we can't ask about the main theme of the last three sentences because of that.

It would be natural to extend the evaluation protocol to such facts, and it would enrich the measure of recall to examine how many facts are spread over several sentences, by annotating a whole document. Considering that the issue is not currently addressed by available systems (they treat all sentences independently), it would be a moot point for now.

3.4 Evaluation Data and Program

3.4.1 Question-Answer Dataset

As a proof of concept, we annotated slightly more than 100 Q&A pairs on a corpus previously employed for OIE evaluation. Rather than aiming for these to become a standard dataset, we encourage other researchers to write their own

questions datasets, tailored to the needs of their particular OIE systems, and enrich the pool of available resources for evaluation.

The data we annotate is that distributed[9] by Del Corro et Gemulla (2013). Sentences were randomly picked from 3 sources :

- 500 sentences are the so-called "Reverb dataset", obtained from the web via a Yahoo random-link service ;
- 200 sentences come from Wikipedia ;
- 200 sentences come from the New York Times.

A sample of annotations is shown in Figure 3.3. To give an idea and as discussed in Section 3.3.3, about half of sentences are not suitable to ask meaningful questions about. On the other sentences, we typically find 2–4 questions that can be answered with their information (2.3 on average on the Reverb dataset, 3 on the wikipedia sentences).

The data is available at `http://www-etud.iro.umontreal.ca/~lechellw/`.

3.4.2 Question Matcher

With annotated Q&A pairs and extracted information in hand, the remaining step is to match and evaluate answers to the questions. We develop a very basic Q&A system, based on string matching. In short, it matches the text of extracted facts to questions, and assumes that a good match indicates the presence of an answer. It is not a very good Q&A system, but it is a decent evaluation script. Its most important features are to be fair to all systems, and easy to use, understand and replicate. Figure 3.5 illustrates how the evaluation system works.

As mentioned in Section 3.3.1, and by contrast with the task of open-domain question answering, e.g. studied in (Fader et al., 2014), we deliberately do not address the difficult problem of question understanding. Instead, questions were

[9] `http://resources.mpi-inf.mpg.de/d5/clausie/`

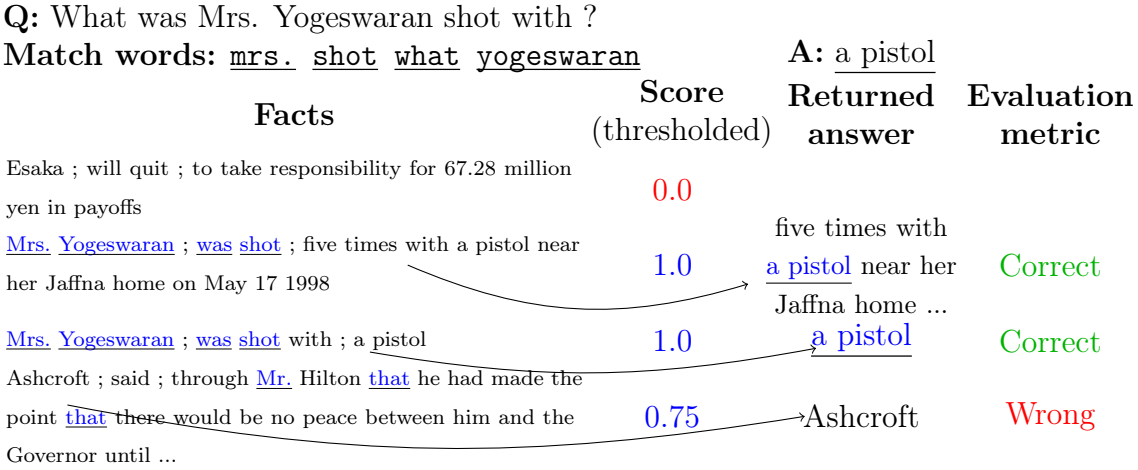


Figure 3.5 – Q&A script based on string matching. A fact that matches the words of a question past a given threshold is assumed to contain an answer to it. The part of that fact (arg1, rel, or arg2) that least matched the question is picked as the answer. A candidate answer is correct if it contains all the words of the reference answer.

written in transparent ways so as to facilitate their automatic answering (see Figure 3.3).

In order to answer a given question, the system attempts to match each extracted fact to it, at the word level. A fact that matches more than 60%[10] of a question's words is assumed to contain an answer to the question. Stop words are excluded, using NLTK (Bird et al., 2009). Edit distance is used to relax the words matching criteria[11], to make up for slight morphological variations between the words of interrogative questions and that of affirmative facts.

When a fact matches a question, we look for the part (first argument, relation, second argument) that least matches the question, and pick it as an answer (typically the second argument, arguments are favored in case of equality, as in the last line of Figure 3.5). We consider an answer to be correct when all the gold answer words are contained in the returned answer.

Were we to build a true question-answering system, we would need to pick or

[10] We examine the impact of this factor in section 3.5.2.

[11] Words differing by 1 or less of their characters are considered to match, as *Mrs.* and *Mr.* in Figure 3.5.

order the various candidate answers gathered for each question. In practice, we seek to devise an evaluation script, and there are only a handful[12] of answers per question, so we consider that a question is correctly answered if any of its candidate answers is correct.

3.5 Results

3.5.1 Performance

The results of the evaluation procedure are shown in Table 3.1. On factoid questions, the automatic answering system finds candidate answers to 35-70% of questions, depending on the information extractor. When candidate answers are found by the answering system, at least one is correct in 20-50% of cases.

Most importantly, recall measures how much of the sentences' relevant information was captured by the extractions, and we can see that by combining the output of all systems, nearly 40% of the information is captured.

Our results fall in line with previous authors' findings. Pattern learning makes Ollie a significant improvement over the simplistic mechanism of Reverb (at the computational price of dependency parsing), making it both more precise and yielding more facts, leading to a large increase in recall. ClausIE extracts more facts than Ollie at a similar level of precision, further boosting recall.

Practically though, while both are open-source software, Ollie runs significantly faster than ClausIE, due to the difference in the embedded parsers they use.

3.5.2 Analysis

In order to assess the quality of the evaluation script in terms of desired behavior, manual assessment of whether a correct answer would be found or not was annotated on a sample of questions (80 out of 106). These are the **X:** lines in Figure 3.3. This would be similar to a human-judged step of answering the questions given the facts, and comparison of the result of the automatic procedure

[12]See Table 3.2 as the exact figure is directly dependent on the matching threshold.

	Answered	Precision	Recall
Reverb	35%	19%	6%
Ollie	39%	43%	17%
ClausIE	68%	42%	29%
All	71%	53%	38%

Table 3.1 – **OIE systems performance results.** Answered is the proportion of factoid questions for which at least one answer was proposed (correct or not) ; Precision is the proportion of answered questions to which one or more answers were correct ; and Recall is the proportion of questions for which at least one candidate answer is correct. As a matter of fact, given the way metrics are computed, answered × precision = recall.

with respect to the manual evaluation (but the annotator tagged its expectation of the automatic procedure, rather than the desired behavior as in the manual judge case). On this sample, the evaluation system performed as predicted in upwards of 95% of cases, which is satisfying.

An important parameter of our approach, mentioned in Section 3.4.2, is the matching threshold past which a fact is assumed to contain an answer to the question it matched. Table 3.2 shows the impact of this parameter on our results, using extractions from all systems.

As expected, the lower the threshold, the looser the answers, and the higher the recall. We retained 0.6 as threshold for performance measures, for it has the highest precision, and above all maximizes recall while keeping the average number of candidate answers reasonable (less than 5 rather than more than 20).

Matching threshold	0.5	**0.6**	0.7	0.8
Questions answered	95%	**71%**	48%	31%
Answers per question	23.	**3.7**	2.5	2.0
Precision	51%	**53%**	46%	38%
Recall	48%	**38%**	22%	12%

Table 3.2 – **Impact of matching threshold on evaluation metrics.**

3.6 Conclusion and Future Work

We presented a new protocol for evaluation of OIE, that consists in annotating questions about the relevant information contained in input text, and automatically answering these questions using systems' output. Our performance metric more closely matches the usefulness of OIE output to end users than the previously employed methodology, by incorporating the informativeness of extracted facts in the annotation process. In addition, our protocol permits to estimate the recall of extraction in absolute terms, which to the best of our knowledge had never been performed. According to our results, about 40% of pieces of knowledge present in sentences are currently extracted by OIE systems.

We annotate a small dataset with Q&A pairs, and present our annotation guidelines, as well as the evaluation script we developed, in the form of a rudimentary Q&A system. We distribute our annotations and evaluation system to the community[13].

As directions for future work, we would like to annotate whole documents rather than isolated sentences, and measure the proportion of cross-sentential information. Our framework also naturally allows for evaluation of other text understanding systems, such as semantic parsers, or full-fledged question answering systems in the place of our own, which would be interesting to perform.

[13] `http://www-etud.iro.umontreal.ca/~lechellw/`

CHAPITRE 4

BANC D'ESSAI DE RÉFÉRENCE POUR L'EIL

William Léchelle, Fabrizio Gotti et Philippe Langlais. WiRe57 : A Fine-Grained Benchmark for Open Information Extraction. *CoRR*, abs/1809.08962, 2018. URL `http://arxiv.org/abs/1809.08962`.

Contexte

Dans la même ligne de raisonnement qu'au chapitre précédent, nous construisons manuellement une référence permettant d'évaluer automatiquement et très précisément la performance d'un système d'EIL. Cependant, les détails diffèrent de plusieurs façons significatives. Au chapitre précédent, le corpus annoté était constitué de phrases indépendantes, et les extractions devaient notamment être minimalement *informatives*, c'est-à-dire pleines de sens par elle-mêmes. Dans ce chapitre, nous annotons des paragraphes suivis, avec la recherche de l'*exhaustivité* des relations extraites comme principe fondamental.

Dès lors, le traitement des anaphores est très différent : au chapitre précédent, toute l'information nécessitant une résolution anaphorique entre plusieurs phrases était ignorée. Ceci s'approchait de la capacité des extracteurs publiés, qui ne résolvent pas ces références. Dans ce travail-ci, la barre est mise plus haut : les anaphores *sont* résolues (explicitement dans la ressource produite), et les informations exprimées par la conjonction de plusieurs phrases *sont* annotées, et font partie du banc d'essai. Même si les systèmes actuels n'arrivent pas à ce niveau d'extraction, un système qui le pourrait serait récompensé. L'un des objectifs de cette approche est de diriger les efforts futurs d'améliorations des EIL, en identifiant les déficiences les plus importantes de façon quantitative. L'absence de traitement de la coréférence est très nettement au premier rang des faiblesses des systèmes d'EIL.

Si dans le chapitre précédent les faits extraits devaient pouvoir s'intégrer à une

base de connaissances tels quels, et y apporter leur contribution, ce critère est légèrement relâché dans ce chapitre-ci. Certes, l'informativité des faits annotés est un critère important dans l'établissement de la référence, mais certains faits exprimés peuvent être redondants, ou peu informatifs lorsque le critère d'exhaustivité prime (dans la mesure où la phrase source les exprime clairement). Par exemple : (*Honshu ; has ; a southeastern side*), ou (*a "metropolitan prefecture" ; differs from ; a prefecture*).

Par ailleurs, comme envisagé en section 3.3.3, un peu de contexte (de l'ordre du domaine d'application, à l'esprit des auteurs et lecteurs typiques du document source) est parfois nécessaire pour interpréter certains faits. Un exemple de cela est (*a maximization step ; computes ; parameters*), qui fait référence à l'étape M de l'algorithme EM, dans un document présentant ledit algorithme.

Plus globalement, ce travail s'attaque à une problématique de fond latente : la définition de la tâche d'EIL est sous-spécifiée. Personne n'a jamais défini précisément le comportement d'un système idéal d'EIL. Par nos « directives d'annotation », nous visons avant tout à **établir**, précisément, quels éléments d'information doivent être extraits, à quel niveau de granularité, la façon de transformer certaines structures syntaxiques, etc., du point de vue d'un système idéal.

Contributions

À force de discussions internes, il est apparu comme une évidence qu'une référence exacte de ce qui est attendu d'un système d'EIL (les extractions qu'un système parfait devrait produire) faciliterait largement toute recherche dans le domaine, ne serait-ce que pour pointer précisément du doigt les déficiences établies, avant d'y remédier.

Fabrizio Gotti et moi-même avons convenu d'un corpus, puis l'avons annoté[1]. Comme détaillé dans l'article, nous avons ensuite réconcilié les différences, et établi des lignes directrices pour l'annotation future. La mesure de l'accord inter-

[1]Philippe Langlais a fourni une annotation partielle.

annotateur a logiquement été un effort commun. J'ai transformé les annotations manuelles en jeu de données au format JSON, et ai formellement défini, puis implémenté la fonction de score. J'ai mené les expériences d'évaluation des systèmes d'EIL disponibles, sur la base d'un outil interne au laboratoire. Fabrizio Gotti a contribué à la rédaction de l'article, ainsi qu'à celle des directives d'annotations.

Impact

Par nos directives d'annotation, nous proposons une spécification précise de ce que constitue la tâche d'EIL. Les principes directeurs, ainsi que les détails d'instanciation dans les phrases du corpus, sont explicités et justifiés (voir l'annexe A). Nous appelons la communauté d'EIL à discuter, reprendre, et améliorer cette première spécification.

Grâce à cette ressource, on peut désormais comparer directement la production d'un système à une référence canonique. Ceci permet d'évaluer très précisément sa performance, de deux façons. D'abord qualitativement, on peut observer les différences mineures entre prédictions et référence, auparavant hors d'atteinte (indéfinies). Plus encore qu'au chapitre précédent, on peut également remarquer les informations que le système n'a pas su extraire (autrement dit, mesurer le rappel). Et quantitativement, on peut maintenant mesurer très précisément l'accord entre un ensemble de prédictions et la référence établie.

De fait, je propose une nouvelle métrique de comparaison entre prédictions et référence, calculée sur la base de la taille des intersections des segments de faits prédits, et de ceux de la référence, pour chaque paire de faits en correspondance (voir section 4.4.1). Puisque cette métrique est précise et sensible à de petits changements dans la prédiction à évaluer, elle permet de procéder de petites modifications itératives d'un même système, pour en mesurer l'impact. Cette métrique se décompose en plusieurs scores de précision, rappel, et correspondances exactes, intéressantes pour examiner en détail la sortie d'un système.

4.1 Introduction

Open Information Extraction systems, starting with TextRunner (Yates et al., 2007), seek to extract all relational tuples expressed in text, without being bound to an anticipated list of predicates. Such systems have been used recently for relation extraction (Soderland et al., 2013), question-answering (Fader et al., 2014), and for building domain-targeted knowledge bases (Mishra et al., 2017), among others.

Subsequent extractors (Reverb (Fader et al., 2011), Ollie (Mausam et al., 2012), ClausIE (Del Corro et Gemulla, 2013), Stanford Open IE (Angeli et al., 2015b), OpenIE4 (Mausam, 2016), MinIE (Gashteovski et al., 2017)) have sought to improve yield and precision, i.e. the number of facts extracted from a given corpus, and the proportion of those facts that is deemed correct.

Nonetheless, the task definition is underspecified, and, to the best of our knowledge, there is no gold standard. Most evaluations require somewhat subjective and inconsistent judgment calls to be made about extracted tuples being acceptable or not. The most recent automatic benchmark of Stanovsky et Dagan (2016) has some shortcomings that we propose to tackle here, regarding the theory underlining the task definition as well as the evaluation procedure.

We manually performed the task of Open Information Extraction on 5 short documents, elaborating tentative guidelines, and resulting in a ground truth reference of 347 tuples. We evaluate against our benchmark the available OIE engines up to MinIE, with a fine-grained token-level evaluation. We distribute our resource and annotation guidelines, along with the evaluation script[2].

Our hope by creating these resources is *not* to create a dataset large enough for the voracity of current machine learning algorithms, but rather to provide a precise, principled benchmark system which we believe can illuminate the current state of OIE technologies as well as some of their shortcomings.

[2] `https://github.com/rali-udem/WiRe57`

4.2 Related Work

For their evaluation, typically, developers of Open IE systems pool the output of various systems on a given corpus. They label a sample of produced tuples as correct or incorrect, with the general guideline that an extraction is correct if it is implied by the sentence. Thus, Mausam et al. (2012) write: *"Two annotators tagged the extractions as correct if the sentence asserted or implied that the relation was true."* Del Corro et Gemulla (2013) propose: *"We also asked labelers to be liberal with respect to coreference or entity resolution; e.g., a proposition such as ('he' ; 'has' ; 'office'), or any unlemmatized version thereof, is treated as correct.*" Saha et al. (2017): *"We sample a random testset of 2,000 sentences [...] Two annotators with NLP experience annotate each extraction for correctness."* Gashteovski et al. (2017): *"A triple is labeled as correct if it is entailed by its corresponding clause."* Then, precision and yield are used as performance metrics. Without a reference, recall is naturally impossible to measure.

We define a reference *a priori*. This allows for automatic scoring of systems' outputs, which greatly diminishes subjectivity from the process of labeling facts "for correctness". Above all, it is meant to help researchers agree on what the task precisely entails. As a consequence, it allows to measure a true recall (albeit on a small corpus).

The complexity of our guidelines is indicative of all that is swept under the carpet when "annotating for correctness". As a matter of fact, when closely examining other references for OIE, many extracted tuples eventually labeled as "good" have more or less important issues. Some really dubious cases are hard to gauge and their labeling is ultimately subjective. To showcase the devilishly difficult judgment calls that this implies, compare the following two extractions. *"'The opportunity is significant and I hope we can take the opportunity to move forward,' he said referring to his coming trip to Britain.*" yields (*his ; has ; coming trip*), and *"[...], the companies included CNN, but not its parent, AOL Time Warner*" yields (*its* ; *has* ; *parent*). Are the extractions implied by the sentence ? In Del Corro et Gemulla

(2013), the annotator approved the latter, and rejected the former. The extraction (*he ; said* ; *The opportunity is significant referring to his coming trip to Britain*) was also deemed correct, despite the composed second argument.

Some other tasks for which OIE output is used, such as Open QA (Fader et al., 2014), TAC-KBP (Soderland et al., 2013), or textual similarity and reading comprehension as in (Stanovsky et al., 2015) — could in principle be used to compare extractors' performance, but only give a very coarse-grained signal, mostly unaffected by the tuning of systems.

A promising method is that explored by Mishra et al. (2017) for the Aristo KB[3]. Aristo is a science-focused KB extracted from a high-quality 7M-sentence corpus. The authors preprocessed a smaller, similarly science-related, independent corpus of 1.2M sentences, into a "Reference KB" of 4147 facts, validated by Turkers. Assuming that these 4147 facts are representative of the science domain as a whole, they measured comprehensiveness (recall) over this domain by measuring coverage on the Reference KB.

4.2.1 ORE Benchmark

Mesquita et al. (2013) compare more or less deep 'parsers', including the OIE systems Ollie and Reverb, on the germane task of Open Relation Extraction (ORE), between named entities. They build a benchmark of 662 binary relations over 1100 sentences from 3 sources (the Web, the New York Times and the Penn Treebank). They label an additional 222 NYT sentences with n-ary relations (one per sentence), and 12,000 with automatic annotations.

Besides the named entity arguments, their annotations consist of one mandatory trigger word (indicating the relation), surrounded by a window of allowed tokens. To compare OIE with ORE systems, they have to replace the target entities by salient arguments (*Asia* and *Europe*) which are easy to recognize. They discuss some of the challenges that arise from divergent annotation styles and evaluation methods.

[3] `http://data.allenai.org/tuple-kb/`

While the tasks are similar, restraining arguments to be named entities limits IE to capturing only the most salient relations expressed in the text. Allowing for any NP to be an argument, we extract 6 facts per sentence on average in the benchmark presented here, compared to 0.6 in the ORE dataset. We also annotate some relations that do not have a trigger in the sentence, such as (*Paris ; [is in] ; France*) from *Chilly Gonzales lived in Paris, France.*

4.2.2 QA-SRL OIE Benchmark

Stanovsky et Dagan (2016) build a large benchmark for OIE, by automatically processing the QA-SRL dataset (He et al., 2015). Precisely, for each predicate annotated in QA-SRL, they generate one tuple expressing each element of the Cartesian product of answers (excluding pronouns) to the questions about this predicate.

For instance, QA-SRL lists five questions asked about the sentence *"Investors are appealing to the SEC not to limit their access to information about stock purchases and sales by corporate insiders"* : *"who are appealing to something ?"*, *"who are someone appealing to ?"*, *"what are someone appealing ?"*, *"what might not limit something ?"* and *"what might not someone limit ?"*, with one answer per question. This generates the reference tuples (*Investors ; appealing ; not to limit their access to information about stock purchases and sales by corporate insiders ; to the SEC*) and (*the SEC ; might not limit* ; *their access to information about stock purchases and sales by corporate insiders*).

Their dataset is comprised of 10,359 tuples over 3200 sentences (from the Wall Street Journal and Wikipedia), and is available for download.[4]

While this work makes a big step in the right direction, there are a few important issues with this benchmark.

First, a major strength of the dataset is its intended and partly achieved completeness, but we do not find it to be a suitably comprehensive reference against which to measure systems' recall. This might be because the QA-SRL dataset

[4] `http://u.cs.biu.ac.il/~nlp/resources/downloads/`

doesn't lend itself well to exhaustiveness in the realm of Open IE, partly because it is restricted to explicit predicates. For instance, the sentence *"However, Paul Johanson, Monsanto's director of plant sciences, said the company's chemical spray overcomes these problems and is 'gentle on the female organ'."* contains two predicates, generating the extractions (*Paul Johanson ; said* ; *the company's chemical spray overcomes these problems and is "gentle on the female organ."*) and (*the company's chemical spray ; overcomes ; these problems*). Yet, that omits the (in our view useful) extractions (*the company's chemical spray ; is ; "gentle on the female organ"*), and (*Paul Johanson ; is ; Monsanto's director of plant sciences*).

Another issue is that some words not found in the original sentence were added by the SRL-to-QA process and retained in the QA-to-OIE transformation, and become part of the reference. In the example above, it is unclear how the second predicate *"might not limit"* is extracted from the sentence. Further, although in this particular case adding the modal is a good way of expressing the information, its repeated use by QA-SRL to produce questions waters down the expressed facts in the end. For instance, the uninformative triple (*a manufacturer ; might get ; something*) is generated from the sentence *"... and if a manufacturer is clearly trying to get something out of it ..."*, with the same added *"might"*.

Last, the scoring procedure isn't robust. Using the code made available by the authors[5], we were able to get top results with a dummy extractor.

This is because the scorer doesn't penalize extractions for being too long, nor for misplacing parts of the relation in the object slot or vice versa. Therefore, if $w_0 w_1 ... w_n$ is an input sentence, a trivial system that "extracts" $(w_0; w_1; w_2 ... w_n)$, $(w_0; w_1 w_2; w_3 ... w_n)$, etc., will be given an unfairly great score. We implemented that program (dubbed Munchkin) which predictably performed well above other genuine extraction systems, as pictured in Figure 4.1.

[5] `https://github.com/gabrielStanovsky/oie-benchmark` — the scoring function was updated since its description in Stanovsky et Dagan (2016). We believe the function from the article suffers from similar issues.

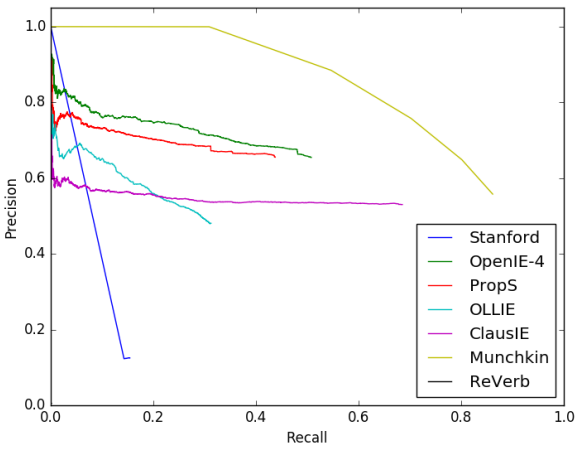


Figure 4.1 – Performance metrics must take span precision into account. The 25-line long Munchkin script returns variations of the full sentence (with decreasing confidence) and is not penalized by the evaluation script of the latest benchmark (Stanovsky et Dagan, 2016). Its superior performance is artificially inflated.

4.2.3 RelVis Benchmarking Toolkit

Schneider et al. (2017) evaluate four systems (ClausIE, OpenIE 4, Stanford Open IE and PredPatt) against the two datasets mentioned above.[6] They use two methods to match predicted and reference tuples : "containment" and "relaxed containment". These methods mean that the predicted tuple must include the reference tuple, and that inclusion must happen for each argument, in the non-relaxed case. In the relaxed case the boundaries between parts of a tuple are ignored. Similarly to that of Stanovsky et Dagan (2016), this scoring procedure doesn't penalize systems for returning overlong spans.

4.2.4 Scoring

To compare facts with a reference, most authors require matching tuples to have the same number of arguments and to share the grammatical head words of each part (e.g. in Angeli et Manning (2013) and in the article of Stanovsky et

[6] Their code is announced but not available as of this writing — `https://github.com/schmaR/relvis`

Dagan (2016)). In the updated GitHub repository of Stanovsky et Dagan (2016), lexical match is used instead : more than half of the words of a predicted tuple must match the reference for it to be considered correct.

In contrast with these works and Schneider et al. (2017), our scorer penalizes verbosity by measuring precision at the token level. We penalize the omission of parts of a reference tuple by gradually diminishing recall (at the token level), instead of a sharp all-or-nothing criterion.

Mesquita et al. (2013) annotate relations as one mandatory target plus some optional complementary words, and treat arguments (named entities) in an ad hoc fashion for OIE systems.

4.3 WiRe57

Open IE bears some similarity to the task of Semantic Role Labeling, as explored in Christensen et al. (2011), Mesquita et al. (2013), and as demonstrated by SRLIE, a component of the OpenIE 4 system.

In effect extracted tuples are akin to simplified PropBank[7] (Kingsbury et Palmer, 2002) or FrameNet[8] (Ruppenhofer et al., 2005) frames, and our annotations were inspired by those projects. Still, with a focus on extracting new relations at scale, optional arguments such as Propbank's modifiers (ArgM) are *discouraged* in OIE. Another major difference is the vocabulary of predicates being open to any relational phrase, rather than belonging to a closed curated list such as VerbNet (Kipper et al., 2000). Within reason, OIE seeks to extract rich and precise relations phrases.

4.3.1 Annotation Process

A small corpus of 57 sentences taken from the beginning of 5 documents in English was used as the source text from which to extract tuples. Three documents

[7] `https://propbank.github.io`
[8] `http://framenet.icsi.berkeley.edu/`

Sentence CH 7 – "*His parents are Ashkenazi Jews who had to flee from Hungary during World War II.*"

Annotations
– (His/(Chilly Gonzales's) parents ; are ; Ashkenazi Jews)
– (His/(Chilly Gonzales's) parents ; are ; Jews)
– (His/(Chilly Gonzales's) parents ; had to flee from ; Hungary ; during World War II)
– (His/(Chilly Gonzales's) parents ; [fled] from ; Hungary ; during World War II)
– ([Chilly Gonzales] ; [has] ; parents)

Sentence EM 5 – "*They pointed out that the method had been 'proposed many times in special circumstances' by earlier authors.*"

Annotations
– (They/(Arthur Dempster, Nan Laird, and Donald Rubin) ; pointed out that ;
(the method)/(The EM algorithm) had been "proposed many times in special circumstances" by earlier authors)
– ((the method)/(The EM algorithm) ; had been proposed by ; earlier authors ;
in special circumstances) [attributed]
– (earlier authors ; proposed ; (the method)/(The EM algorithm) ;
in special circumstances) [attributed]

Sentence FI 2 – "*A police statement did not name the man in the boot, but in effect indicated the traveler was State Secretary Samuli Virtanen, who is also the deputy to Foreign Minister Timo Soini.*"

Annotations
– (A police/(Finnish police) statement ; did not name ;
(the man in the boot)/(Samuli Virtanen))
– ((the man in the boot)/(Samuli Virtanen) ; was ; Samuli Virtanen) [attributed]
– ((the traveler)/(Samuli Virtanen) ; was ; Samuli Virtanen) [attributed]
– (Samuli Virtanen ; [is] ; State Secretary)
– (Samuli Virtanen ; is ; the deputy to Foreign Minister Timo Soini)
– (Samuli Virtanen ; is ; [a] deputy)
– (Timo Soini ; [is] ; Foreign Minister)
– (Timo Soini ; [has] ; [a] deputy)

Sentence CE 4 – "*The International Monetary Fund, for example, saw 2017 global growth at 3.4 percent with advanced economies advancing 1.8 percent.*"

Annotations
– (The International Monetary Fund ; saw ; 2017 global growth ; at 3.4 percent)
– (The International Monetary Fund ; saw ; advanced economies ;
advancing 1.8 percent ; [in] 2017)
– (2017 global growth ; [was] ; 3.4 percent)
– (advanced economies ; [advanced] ; 1.8 percent ; [in] 2017) [attributed]

Figure 4.2 – Sample annotations from WiRe57, from four of the documents. Reformulated words are enclosed in [brackets] and coreference information is indicated with forward slashes and parentheses, as detailed in the guidelines.

are Wikipedia articles (Chilly Gonzales, the EM algorithm, and Tokyo) and two are newswire articles (taken from Reuters, hence the Wi-Re name).

Two annotators (authors of this paper) first independently extracted tuples from the documents, based on a first version of the annotation guidelines which quickly proved insufficient to reach any significant agreement. The two sets of annotations were then merged, and the guidelines rectified along the way in order to resolve the issues that arose. After merging, a quick test on a few additional sentences from a different document shown a much improved agreement, more than half of extractions matching exactly and the remaining missing a few details. The guidelines are detailed in the next sections.

4.3.2 Annotation Principles

In keeping with past literature, our guiding principles for the annotation were as follows.

The first, obvious purpose of extracted information is to be **informative**. Fader et al. (2011) mention how extracting (*Faust ; made ; a deal*) instead of the correct (*Faust ; made a deal with ; the devil*) would be pointless. Further, anaphoric mentions being so ubiquitous and being void of meaning outside the context of their original sentence, we resolve anaphora in our extractions.

Moreover and following Stanovsky et Dagan (2016), extracted tuples should each be **minimal**, in the sense that they should convey the smallest standalone piece of information, though that piece must be completely expressed. This means that some facts must be extracted as n-ary relations[9]. The MinIE system Gashteovski et al. (2017) in particular addresses this issue and "minimizes its extractions by identifying and removing parts that are considered overly specific".

The annotation shall be **exhaustive**, in the sense of capturing as much of the information expressed in the text as possible. This is to measure absolute recall for a system, a notoriously difficult evaluation metric for Open IE.

[9]Some systems — namely CSD-IE (Bast et Haussmann, 2013) and NestIE (Bhutani et al., 2016) — explore nesting extractions, but we didn't adopt this strategy.

This in turn raises the issue of **inference** : some information is merely suggested by the text, rather than explicitly expressed, and should not be annotated. Light inference, in the form of reformulation, is helpful to make use of the information extracted, but full-fledged inference should be processed by a dedicated program, and is not part of the Open IE task.

Past authors mention this issue : from Wu et Weld (2010) : "The extractor should produce one triple for every relation stated explicitly in the text, but is not required to infer implicit facts." Stanovsky et Dagan (2016) say : "an Open IE extractor should produce the tuple (*John ; managed to open ; the door*) but is not required to produce the extraction (*John ; opened ; the door*)". In this resource we do also annotate (*John ; [opened] ; the door*), marking the reworded relation as inferred (which in turn makes it optional to find when scoring).

4.3.3 Annotation Guidelines

Our full annotation guidelines are shared at `https://github.com/rali-udem/WiRe57`. We present its major points here.

Extracted tuples should reflect all meaningful relationships found in the source text. Typically, this means that there are multiple tuples for a given sentence. A number of times, two arguments are connected in a sentence but the relation that links them is implicit (e.g. *Paris, France*, *the North Atlantic Treaty Organization (NATO)*, *the Nature paper* or *the Turing paper*, etc.). In this case, we annotate a somewhat arbitrary relationship (such as *is in*, *stands for*, *published in* and *published by* respectively), the tokens of which are thus inferred. This is the case for 39% of our tuples.

Some OIE systems similarly attempt to hallucinate some or part of relations. Notably, ClausIE wrongly extracts (*New Delhi ; is ; India*) and MinIE gets right (*Paris ; is in ; France*). Ollie adds some "be" auxiliaries to otherwise nominal relations, as in *Barack Obama, former president of the United States, [...]*, which OpenIE 4 also infers. Yet, we acknowledge that most work in Open IE rely on explicit predicate tokens as in Mesquita et al. (2013), and don't try to elicit relations

further. At scoring time, systems are not penalized for not finding inferred words, or not finding inferred relations. If the whole predicate of a tuple is inferred, a predicted tuple is scored on its token overlap with the arguments only.

We suggest "platinum" annotations, including inferred words, to be a very high standard for extractors, while the gold standard for the task, recall-wise, is based only on words found in the original sentences.

Noun phrases can be rich in elements of information. To solve the problem of finding the granularity level to use when including argument NPs, we extract two tuples, one as generic as possible and the other as specific as possible, for the same relation. Adjectives and other elements of meaning that can be easily separated from the noun phrase to create other tuples are so split. Only elements that cannot become part of the most specific noun phrase.

For instance, the sentence *"Solo Piano is a great album of classical piano compositions."* would yield 3 tuples : the split adjective (*Solo Piano ; is ; great*), the generic (*Solo Piano ; is ; [an] album*) and the specific (*Solo Piano ; is ; [an] album of classical piano compositions*).

When predicates contain nouns or other elements (e.g. *Tokyo is the capital of Japan.*), we annotate the richer relationship (*Tokyo ; is the capital of ; Japan*) rather than the more basic (*Tokyo ; is ; the capital of Japan*). This allows tuple relations to be more meaningful, and more easily compared, clustered, and aggregated with other relations. This also is in line with Reverb.

Like ClausIE and other extractors since, we split conjunctions : *"Andrea lived in both Poland and Italy*" yields both (*Andrea ; lived in ; Poland*) and (*Andrea ; lived in ; Italy*).

4.3.4 Resource

A sample of annotations is pictured in Figure 4.2. The occurring frequency of various phenomena is presented in Table 4.1. Our resource is comprised of 343 relational facts (or tuples), three quarters of them binary relations. One in five have three arguments, sometimes "two objects" as in (*This performance ; has made ;*

Phenomenon	N	%
All tuples	343	100
Anaphora	196	57
Contains inferred words	186	54
Hallucinated parts	135	39
Binary relations	254	74
n-ary, $n = 3$	72	21
n-ary, $n = 4$	16	5
n-ary, $n = 5$	1	0.3
Inferred words	347/2597	13.4

Table 4.1 – Frequencies of various phenomena in WiRe57

some economists ; optimistic) or more frequently a complement as in (*His parents ; had to flee ; from Hungary ; during World War II*). Five percent of them have four arguments or more : for instance (*Tokyo ; ranked ; third ; in the International Financial Centres Development IndexEdit ; twice*) and (*The International Monetary Fund ; saw ; advanced economies ; advancing 1.8 percent ; [in] 2017*).

We found (and resolved) anaphoric phrases in more than half the tuples, as in (*Emperor Meiji ; moved* ; *his/(Emperor Meiji's) seat* ; *to (the city)/Tokyo* ; *from the old capital of Kyoto ; in 1868*). The released dataset contains the raw and anaphora-resolved argument spans.

When solely extracting words from the sentence would not yield clear factual tuples, we reworded or adapted the text into more explicit statements. In this case, we marked the changed (or added) words as inferred (brackets in Figure 4.2). For instance, in sentence CE 4, the relation "[advanced]" was reformulated from the sentence word "*advancing*", and the word [in] was added before "2017". In the resource, each token is accompanied by its index in the sentence if it comes from it, or the "inferred" mark. Inferred words represent 13% of the lot but affect 54% of the tuples.

	# tokens	1↔2	1↔R	2↔R
Sentence 1	24	84.4	90.6	93.8
Sentence 2	19	98.7	98.7	100
Sentence 3	33	78.0	90.9	85.6
Average		**85.2**	**92.8**	**91.9**

Table 4.2 – **Inter-annotator agreement.** Percentage of agreement on the labeling of each sentence token as belonging to four classes. Each annotator's original production differs only slightly from the agreed-on result (columns 1↔R and 2↔R), and the disagreement between both annotators is slightly larger (column 1↔2). The average is computed token-wise.

4.3.5 Inter-Annotator Agreement

After the guidelines were fully settled, three additional sentences from one of the documents were annotated by two annotators (1 and 2) in order to measure inter-annotator agreement. Afterward, annotation discrepancies were resolved in cases of disagreement to produce a merged reference (R). Here, we report the agreement between the two original annotations (1↔2), and between each original annotation and the merged reference (1↔R and 2↔R).

Comparing triples can become quite tricky for many reasons, including missing complements, overlapping spans, etc. We therefore resorted to another scheme, where we reframe the annotation task as taking each annotated token and classifying it as either belonging or not belonging to each of four classes (subject, relation, object, or complementary argument). These classifications can be trivially derived from the triples produced beforehand. For instance, a triple (t_1 t_2; t_3; t_4 t_5) implies that the annotator classified tokens t_1 and t_2 as belonging to the subject class. It then becomes possible to measure an agreement percentage on the full binary labeling grid (obtained automatically from the long-form annotations). We believe the resulting figures (shown in Table 4.2) aptly reflect the level of overall agreement between the annotators. We measure an overall inter-annotator agreement (1↔2) of 85.2% for the three sentences.

Qualitatively, one annotator steered close to the sentence syntax, sometimes

missing some of the meaning obscured by long-winded formulations. The other annotator, on the other hand, tended to be overly specific, including some nonessential complements, and making longer-ranged inferences that fall out of the scope of this task. Some possessive and passive constructions were also overlooked.

4.4 Evaluation of Existing Systems

4.4.1 Scorer

An important step when measuring extractors performance is the scoring process. Matching a system's output to a reference isn't trivial. As detailed in Section 4.2.2, because it didn't penalize overlong extractions, we could game the basic evaluation method of the QA-SRL OIE benchmark with a trivial extractor.

Our scorer computes precision and recall of a system's predicted tuples at the token level. Precision is, briefly put, the proportion of extracted words that are found in the reference. Recall is the proportion of reference words found in the systems' predictions.

More formally, let $G = \{g_1, g_2, \ldots, g_N\}$ be the gold tuples, and $T_{\text{sys}} = \{t_1, t_2, \ldots, t_n\}$ a system's extractions. We denote the parts of a tuple $t = (t^{a_1}; t^r; t^{a_2}; t^{a_3}; \ldots) = (t^{p_k})_{k \in [1,6]}$, where p_1 is the first argument, p_2 is the relation, etc., up to p_6 the fifth argument when it exists (no reference tuple contains more than 5 arguments). Let $t_i^p \cap g_j^p$ be the subset of words shared by parts t_i^p and g_j^p, where parts are considered as bags of words. The length of a tuple is the sum of lengths of its parts, i.e. $|t_i| = |t_i^{a_1}| + |t_i^r| + |t_i^{a_2}| + |t_i^{a_3}| + \cdots = \sum_k |t_i^{p_k}|$.

A predicted tuple t_i may match a reference tuple g_j from the same sentence if it shares at least one word from each of the relation, first and second arguments with it. That is, iff $(w_{a_1}; w_r; w_{a_2})$ exist such that $w_1 \in g_j^{a_1} \cap t_i^{a_1}, w_r \in g_j^r \cap t_i^r$ and $w_2 \in g_j^{a_2} \cap t_i^{a_2}$.

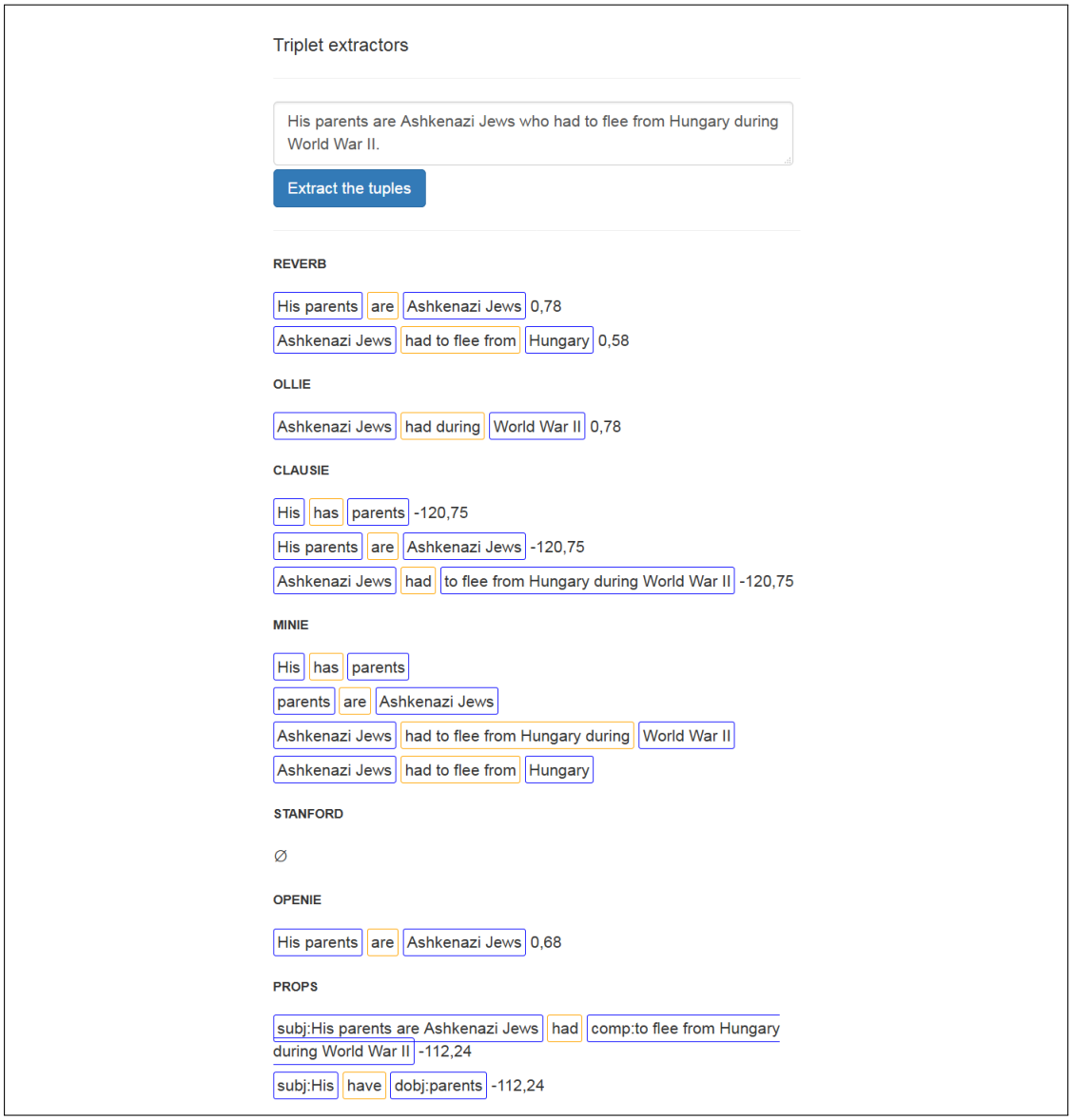


Figure 4.3 – Example output of evaluated OIE systems, on sentence CH 7. This screenshot is of an in-house web application that allows us to submit any sentence for tuple extraction and to visualize the results.

For all tuple pairs that may match, we have the matching scores:

$$\text{precision}(t_i, g_j) = \frac{\sum_k |t_i^{p_k} \cap g_j^{p_k}|}{|t_i|}$$
$$\text{recall}(t_i, g_j) = \frac{\sum_k |t_i^{p_k} \cap g_j^{p_k}|}{|g_j|}$$
$$F_1 = \frac{2\ p\ r}{p + r}.$$

We match predicted tuples with reference ones by greedily removing from the potential match pool the pair with maximum F_1 score, until no remaining tuples match. Let $m(.)$ be the matching function such that t_i matches with $g_{m(i)}$ (and conversely $t_{m(j)}$ matches g_j), assuming that $|t_i \cap g_{m(i)}| = 0$ if there is no match for t_i.

Hence, the overall performance metrics of an extractor are its token-weighted precision and recall over all tuples, i.e.

$$\text{precision}_{\text{sys}} = \frac{\sum_i^n \left(\sum_k |t_i^{p_k} \cap g_{m(i)}^{p_k}| \right)}{\sum_i^n |t_i|}$$
$$\text{recall}_{\text{sys}} = \frac{\sum_j^N \left(\sum_k |t_{m(j)}^{p_k} \cap g_j^{p_k}| \right)}{\sum_j^N |g_j|}$$
$$F_{1\text{sys}} = \frac{2\ p_{\text{sys}}\ r_{\text{sys}}}{p_{\text{sys}} + r_{\text{sys}}}.$$

To avoid penalizing systems for not finding them, neither the words annotated as inferred, nor the coreference information, are used in this evaluation (g_j is the non-resolved version of the tuple, and inferred words are not included in recall denominators). Future work can look into evaluating OIE systems that mean to resolve anaphoric mentions.

	Tuples	Match	Exact match	Prec. of matches	Recall of matches	Prec.	Recall	F1
Reverb (Fader et al., 2011)	79	54	13	.83	.77	**.569**	.121	.200
Ollie (Mausam et al., 2012)	145	74	8	.73	.81	.347	.175	.239
ClausIE (Del Corro et Gemulla, 2013)	223	121	**24**	.74	.84	.401	.298	.342
Stanford (Angeli et al., 2015b)	371	99	2	.79	.65	.210	.188	.198
OpenIE 4 (Mausam, 2016)	101	74	5	.68	.84	.501	.182	.267
PropS (Stanovsky et al., 2016)	184	69	0	.59	.80	.222	.162	.187
MinIE (Gashteovski et al., 2017)	252	**134**	10	.75	.83	.400	**.323**	**.358**

Table 4.3 – Performance of available OpenIE systems (in chronological order) on our reference. Precision and recall are computed at the token level. Systems with lower precision of matches are penalized for producing overlong tuples. High precision and recall of matches overall show that our matching function (one shared word in each of the first three parts) works correctly. Inferred words are required for exact matches.

4.4.2 Results

In order to experiment with the seven systems described in this paper, we installed them as a single web service on one computer. A client application need only submit a sentence and a list of OIE system names to perform extraction. All tuples are in turn served as JSON objects following a unique, documented pattern, no matter the OIE system used. This considerably facilitates the development of clients, shielded from the different tuple formats, various coding languages (Scala, Java, Python), and other quirks of the OIE systems presented here. It also allowed us to visualize the tuples using a web application, pictured in Figure 4.3. Moreover, because the various extractors run as servers, they load their respective resources (1 or 2 minutes) only once, when the service is launched, and are then always quick to respond to a given extraction task (a few seconds). Otherwise, the user would have had to wait for the resources to load each time when querying the extractors.

While creating such a framework is a significant effort, it ultimately saved us a lot of time when writing the clients. It also provided a common frame of reference for all collaborators in our lab. Typically, we used the default configuration for each OIE system, but we tweaked the available flags in order to favor exhaustiveness,

when such flags were present and properly documented. When additional information did not fit into a traditional tuple (arg1; rel; arg2), e.g. MinIE's quantities, we resorted to simple schemes to faithfully cast that information into a tuple.

Table 4.3 details the performance of available OIE systems against our reference. MinIE produces a large number of correct tuples, and performs best, especially recall-wise. The conservative choices made by Reverb achieve a relatively high precision, though it lacks in comprehensiveness. Ollie improves recall over Reverb, and Open IE 4 improves precision over Ollie. Stanford Open IE produces a very large number of tuples, hindering its precision (it is possible that limiting its verboseness through configuration would improve this).

4.5 Conclusion

In this paper, we set out to create additional resources useful to researchers in Open Information Extraction. We distribute these resources freely.

First and foremost, we provide a manually crafted, tentative reference for the task. It consists of 343 manually extracted facts, including some implicit relations, over 57 sentences. A quarter of them are n-ary relations and coreference information is included in over half of them. We believe that such a benchmark is valuable because it offers a common frame of reference allowing OIE systems to be tested and compared fairly, a task we carried out on seven OIE systems. This also entailed the creation of a scoring algorithm and program, which we release along with the data. We assess the Reverb, Ollie, ClausIE, Stanford Open IE, OpenIE 4, and MinIE systems against our reference, using a fine-grained token-level scorer. We find the MinIE system to perform best.

Naturally, such an annotation effort requires one to attempt to "pin down" the task of OIE by confronting real-life data. We provide guidelines that propose such a definition. While by no means definitive or exhaustive, these guidelines have at least the merit of being sufficiently clear to yield an annotated dataset with a reasonable inter-annotator agreement. At the same time, we believe they are not

too overwrought, and rather invite further contributions by other researchers. The thorniest issues are the fine line between useful reformulation of information to a canonical form and ill-advised inference, and how to trim and annotate complex noun-phrase arguments. These difficulties can affect the manual annotation process. Interestingly, these issues are also likely to arise when building automated OIE systems, which is the ultimate goal in this research field after all.

A Annexe - Directives d'annotation pour WiRe57

This companion document to the WiRe57 Open Information Extraction (OIE) benchmark explains in detail the principles and guidelines used to produce the benchmark. It can also be employed by any researcher willing to annotate sentences with their reference (manual) tuples in order to create their own benchmark.

A.1 General Principles

Briefly put, the goal of the annotation task for which we provide guidelines in this document is to produce reference tuples (extractions) from a text. We seek to manually extract all meaningful relationships occurring in the text : in other words, we want to be as **exhaustive** as possible. At the same time, we want to limit the amount of inference and reasoning necessary to produce these reference tuples. Obviously, "limited inference" is ambiguous, but we hope that the examples we provide here will help clarify this admittedly arbitrary choice on our part.

It is worth noting that these guidelines are a compromise between, on the one hand, the need for exhaustive and unambiguous annotations of tuples in sentences, and, on the other hand, an effort to simplify the annotation process for a human being.

The rest of the document explains the annotation guidelines we used to build WiRe57, along with examples for each rule. For further examples, the reader can also simply consult the WiRe57 annotated benchmark itself.

A.2 Annotation Guidelines

In the following, we represent tuples as parenthesized elements, following the pattern (arg_1, rel, arg_2, arg_3, ...) where arg_i is the argument at position i and rel is the relationship expressed explicitly or implicitly in the source text between the arguments. The number of arguments can be one or more. For instance, (*Kyle, read, the book*) or (*Kyle, fell*).

Typically, the tokens found in the tuples (e.g. *Kyle*) are found verbatim in the original text. However, this is not always the case. When one or more tokens in a tuple are not directly extracted from the underlying text, they are enclosed in square brackets, like in the example below.

Jewish musician Bob Dylan	(Bob Dylan, [is], Jewish)
	(Bob Dylan, [is], [a] musician)

Our examples will always take this form: The source text on the left and the tuples that illustrate the guideline being discussed on the right. Sometimes, an example tuple list is not exhaustive, i.e. some other tuples could have been derived from the text, but were not shown in order to alleviate the presentation.

A.2.1 Exhaustiveness

The tuples should reflect all meaningful relationships found in the source text. Typically, this means that there are multiple tuples for a given sentence, although this is by no means a requirement. It also allows a single word to be involved in more than one tuple.

When in doubt, we recommend that the annotator err on the side of exhaustiveness.

Kyle and Mary read the book, and then left for school.	(Kyle, read, the book)
	(Mary, read, the book)
	(Kyle, left for, school)
	(Mary, left for, school)

The annotator must also consider relationships and facts that may not be expressed explicitly through a verb, i.e. *non-verbal* relations. Typically, the relation in these tuples will be inferred and written enclosed in square brackets, e.g. [happened in] or [stands for].

1923 Kantō earthquake	(Kantō earthquake, [happened in], 1923)
The United Nations (UN)	(UN, [stands for], The United Nations)

A.2.2 Noun Phrases

Often, noun phrases involved in a tuple can be rich in multiple elements of information. For instance, in the sentence *"Solo Piano I is an album of classical piano compositions."*, the phrase *"an album of classical piano compositions"* should yield two tuples when annotating: one using its simplest form *"an album*" and another using its complete form *"an album of classical piano compositions*". We therefore extract two tuples, one very generic and another, more specific, for the same relation.

Solo Piano I is an album	(Solo Piano I, is, an album)
of classical piano compositions.	(Solo Piano I, is, an album of classical piano compositions)

The complete, longer form of a noun phrase does not include adjectives and other elements of meaning that can be easily separated from the noun phrase to create other tuples. For instance, in the following example, the adjective *great* need not appear in either the simple or extended form of the noun phrase, because it can easily be isolated in another triple (*Solo Piano I, [is], great*).

Solo Piano I is a great album	(Solo Piano I, is, [an] album)
of classical piano compositions.	(Solo Piano I, is, [an] album of classical piano compositions)
	(Solo Piano I, [is], great)

A.2.3 Rich Relations

When relationships are expressed with verb phrases containing nouns or other elements (e.g. *"Tokyo is the capital of Japan.*"), we prefer annotating the richer

relationship (*Tokyo, is the capital of, Japan*) rather than the more basic (*Tokyo, is, the capital of Japan*). This allows tuple relations to be more meaningful, and more easily compared, clustered, and aggregated with other relations.

Tokyo is home to Fuji TV.	(Tokyo, is home to, Fuji TV) *essential tuple*
	(Tokyo, is, home to Fuji TV) *discouraged tuple*
Tokyo's population is over 13 million.	(Tokyo's population, is over, 13 million) *essential tuple*
Their first release, the LP Thriller, was moderately successful.	(Thriller, was moderately successful)

A.2.4 Prepositions in annotations

When a relationship verb calls for a preposition, the preposition is placed after the verb in the tuple's relation slot (*Kyle, left for, school*), or just before the argument when there are more than one preposition involved in the tuple (*Kyle, left for, school, on Tuesday*).

A.2.5 Minimality

The tuples should split coordinate elements when appropriate, so as to have as many tuples as there are elements of information. Care should be taken when considering a split, as some coordinated elements should be kept joined (see second and third examples below).

He has Cornish as well as Welsh ancestry.	(He, has, Cornish ancestry)
	(He, has, Welsh ancestry)
This dog is black and brown.	(The dog, is, black and brown)
The printer alternates between portrait and landscape.	(The printer, alternates between, portrait and landscape)
He wears red shoes or black shoes.	(He, wears, red shoes or black shoes)

Some elements may not be explicitly coordinated with a conjunction, yet they should be separate elements of meaningful information, belonging to distinct tuples. Noun phrases are good examples of this.

Flubber is a critically acclaimed cult film.	(Flubber, is, a cult film)
	(Flubber, is, critically acclaimed)

A.2.6 Possessives

Possessives are a special case of inferred relations where the relation is "[has]".

Mary's dog is brown.	(Mary's dog, is, brown)
	(Mary, [has], [a] dog)
The prefecture of this city	(this city, [has], [a] prefecture)
The GOP is American.	(The GOP, is, American)
Its leaders include Ronald Reagan.	(Its/(The GOP's) leaders, include, Ronald Reagan)
	(The GOP, [has], leaders)

A.2.7 Rephrasing

Complex verbal phrases are rephrased more simply, yielding two tuples: one using the original phrasing, and another using the simplified one.

Sam managed to convince John.	(Sam, managed to convince, John)
	(Sam, [convinced], John)
The Kellers had to flee from Germany.	(The Kellers, had to flee from, Germany)
	(The Kellers, [fled] from, Germany)

A.2.8 Passive and Active Voice

When a verb is used at the passive voice, it yields two triples, the original one and another one, rephrased at the active voice.

The apple was eaten by Kyle.	(The apple, was eaten by, Kyle)
	(Kyle, [ate], the apple)

A.2.9 Tuples where more than two arguments are necessary

For a given relationship, the annotator must try to identify all arguments that are essential or very important to the meaning of the resulting tuple. Typically, this means adding place and time arguments to a given verb that calls for them. The order of the arguments after the second argument does not reflect their relative importance when conveying the meaning of the tuple.

Note that the preposition *to* is placed in the relation slot while *in* is in arg_3, in keeping with section A.2.4.

In 1869, the 17-year-old Emperor Meiji moved to Edo.	(Meiji, moved to, Edo, in 1869)
He has more apples than her.	(He, has, more apples, than her)
Emperor Meiji moved this seat to Tokyo from the old capital of Kyoto in 1868.	(Emperor Meiji, moved, this seat, to Tokyo, from the old capital of Kyoto, in 1868)

When an argument adds little information or specificity to a tuple, it can be dropped. This happens typically with adverbs of manner.

Bob liked to play piano from time to time. (Bob, liked to play, piano)

When multiple arguments compete for a single slot in a tuple (through an *and* or an *or*, for instance), they produce as many tuples as there are competing arguments.

Bob played the piano and the sax in the 80s.	(Bob, played, the piano, in the 80s) (Bob, played, the sax, in the 80s)
Bob both hated and loved this theory.	(Bob, hated, this theory) (Bob, loved, this theory)

A.2.10 Tuples calling for a single argument

Some relationships have a single argument, typically the subject of the verb, which results in tuples of the form (arg_1, rel). In the example below, *"grow in popularity*" is a non-compositional phrase that cannot be expressed as ~~(X, grow in, popularity)~~ and therefore does not call for any argument.

Sangster grew in popularity. (Sangster, grew in popularity)

A.2.11 Attribution and Speculation

The truth value of some statements and relationships in a given text are influenced by either attribution (e.g. "according to X") or speculation (e.g. "If Y holds, then Z."). The annotation guidelines we propose here attempt to capture this, albeit superficially, by decorating such attributed and speculative tuples with an additional Boolean flag.

This means that the annotation process must be arranged so that the annotator can add the value of such a flag for any given tuple they produce. Typically, this takes the form of an additional field the annotator can tick when they deem the tuple to be attributed/speculative.

An Apple Valley, California man plans on launching himself in a homemade rocket to prepare for obtaining evidence that the Earth is flat, according to The Washington Post.	(the Earth, is, flat) [✓ attributive/speculative]
The Earth is round, according to sane people.	(The Earth, is, round) [✓ attributive/speculative]
If everything goes well, the Sun shines on you.	(The Sun, shines on, you) [✓ attributive/speculative]

When a sentence has the form *"X said that Y*" or equivalent, the annotator produces the triple (*X, said that, Y*), regardless of the syntactic or semantic complexity of Y.

The idea is that all is pretty much on track for growth that will be stronger than in 2017.	(The idea, is that, all is pretty much on track for growth that will be stronger than in 2017)

A.2.12 Pan-document Anaphora Resolution (optional, but highly recommended)

The authors of these guidelines are convinced that anaphora resolution is an integral part of open information extraction, whether it be carried out by machines or by humans. A large proportion of nouns and noun phrases involved in tuples are not explicitly mentioned, but rather used indirectly through anaphora. For WiRe57, for instance, we found that 57% of tuples fall into this category.

Anaphora resolution allows the tuples to convey immediate meaning, rather than having to be processed (imperfectly) by anaphora resolution pipelines. Consequently, we found it somewhat simplifies the assessment of their quality as the annotator produces them.

The antecedent of a given anaphora can be anywhere in the document, or can be deduced from prior knowledge, *as long as this antecedent is used consistently for all its anaphoric mentions.*

We use the forward slash to separate the mention from its antecedent, like so: "she/Mary". When there are multiple words in either the mention or the antecedent, parentheses surround the multi-word expressions, like so: "she/(Mary Simpson)" or "(the girl)/(Mary Simpson)".

John likes Mary.	(John, likes, Mary)
But he shies away from the beautiful girl.	(he/John, shies away from, (the beautiful girl)/Mary)

Possessive adjectives are also resolved by using an antecedent in a possessive form (e.g. *John's*).

John plays football, but his friends do not.	(John, plays, football) (His/(John's) friends, do not, [play] football)

A.2.13 Verbal Tenses

The verbs in the tuples should employ the same verbal tense as that found in the original text. When a verb is inferred, however, its tense should be the one the writer would have used, had they had to make explicit the relation. In the example below, note the past for "took place in" and the present for [has].

The meeting took place in June, a day after	(The meeting, took place in, June)
Virtanen's co-ruling Finns party had elected	([the] Finns party, [has], new leaders)
anti-immigration hardliners as its new leaders.	

A.2.14 Don't extract a generic plural from a sentence singular

The annotator must refrain from inferring generic truths from isolated examples, for instance by converting a single fact in a sentence (*a McGill-trained pianist*) to a stronger statement (*McGill, trains, pianists*). It would amount to the logical sin of inferring general truths from isolated examples, as in *"The Sun rose today."* yielding (*The Sun, rises*). It is the job of the inference system to abstract upon single examples, when appropriate.

CHAPITRE 5

PRÉDICTION DE NOUVEAUX TRIPLETS : CLASSIFICATION

William Léchelle et Phillippe Langlais. Revisiting the Task of Scoring Open IE Relations. Dans *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japon, May 2018. European Language Resources Association (ELRA).

Dans les chapitres précédents, nous avons évalué les productions de systèmes d'EIL, envisagées comme des listes de faits. Dans les chapitres qui suivent, nous allons nous intéresser à une composante sous-étudiée mais pas moins importante de l'EIL : la **fonction de confiance**.

Chaque extraction produite par un système se voit attribuer un score de confiance, mesure approximative du crédit que le système accorde à cette information particulière. Comme nous allons le voir (en particulier section 5.2.3), cette confiance peut être fondée sur diverses sources d'informations : la phrase d'origine, le fait lui-même, d'autres faits extraits, une base de connaissances établie, etc. À un haut niveau, la fonction de confiance permet deux choses : filtrer les extractions candidates des systèmes d'EIL, et suggérer et prédire de nouveaux faits à partir de bases de connaissances existantes.

Les systèmes d'EIL étant imparfaits, certains triplets produits sont bruités ou ininformatifs, pendant qu'une fraction significative des éléments d'information présents sont omis. Cette double lacune en précision et en rappel est pourquoi la fonction de confiance est si importante : si l'on disposait d'un score reflétant adéquatement la probabilité qu'une extraction soit correcte, on pourrait obtenir facilement soit une base de connaissance de taille restreinte mais de bonne qualité, soit une large couverture des relations exprimées dans le texte, au prix d'une moindre précision. Sans fonction de confiance, toutes les extractions sont sur le même plan, autant les reformulations triviales de la phrase qui sont généralement exactes, que

les correspondances avec des schémas d'extractions rares et alambiqués, qui sont plus douteux ; autant les informations vues plusieurs fois à propos d'entités connues que des combinaisons improbables d'entités incompatibles. Le système d'EIL de Stanford est un bon exemple de ce problème : la plupart des extractions ont le même score de 1, le programme manque parfois des informations faciles à extraire, tout en produisant un relatif océan de bruit sous la forme de très nombreuses productions redondantes (voir figure 2.1).

Dans ce chapitre, on ramène la fonction de score à un choix binaire entre faits « corrects » ou non, appliquée à des requêtes, nouveaux faits candidats à l'intégration dans la base de connaissances. Ces candidats pourraient provenir d'un utilisateur posant des questions, d'un système d'inférence tentant de raisonner, ou d'un extracteur produisant de nombreux faits correspondant syntaxiquement à des patrons d'extraction possibles, nécessitant une étape de filtrage ou de validation sémantique.

Dans le chapitre suivant, on se rapprochera du paradigme standard de la prédiction de liens, et l'on proposera de modéliser une base de connaissances produites par EIL, de façon analogue aux modèles existants en complétion de bases de connaissances. Autrement dit, on évaluera des fonctions de confiance habituellement appliquées à des bases de connaissances construites manuellement, sur des bases produites par EIL.

Contexte

On cherche à améliorer la qualité de la sortie d'un système d'EIL, en en filtrant les extractions les plus absurdes. De nombreuses productions de Reverb et d'Ollie sont *absconses*, parfois hautement contextuelles, parfois absurdes, globalement informatives. Dans le modèle d'Ollie, on souhaite les identifier pour les éliminer à la sortie du moteur syntaxique de reconnaissance de patrons, pour augmenter la précision du système.

Le travail d'Angeli et Manning (2013), propose un modèle complexe pour clas-

sifier un fait-requête comme compatible, ou semblable, à une base de connaissances existante. Remarquablement, les auteurs évaluent leur approche avec Reverb, et ConceptNet. Avec le même objectif, nous proposons un modèle beaucoup plus simple, et tout aussi efficace, s'appuyant sur un modèle de langue. Lors des expériences, j'ai remarqué que la définition précise du protocole d'évaluation de la tâche a une influence significative sur les résultats, ce qui vaut la peine d'être documenté. Les différents résultats présentés dans l'article résultent d'une exploration naturellement diffuse dans toutes les directions autour d'un protocole d'évaluation novateur.

Finalement, cet article servira principalement à formaliser le protocole avancé par Angeli et Manning (2013), ce qui ouvre la voie à en tisser les liens avec le protocole d'évaluation de la prédiction de liens dans les bases de connaissances (voir le chapitre suivant). Fondamentalement, les différences principales entre les deux sont la nature de la base de connaissances, assemblée avec soin dans le cas de Freebase ou WordNet, extraite automatiquement avec une part de bruit pour Reverb, et la source des requêtes (en prédiction de liens, on attribue un score à toutes les complétions possibles d'un fait partiel, on attribue à chaque étape autant de scores qu'il y a d'entités dans le corpus). Au chapitre suivant, on s'affranchira de la première différence, en évaluant les méthodes de prédiction de liens sur les sorties de l'EIL.

Ce qui rend la tâche étudiée dans cet article un peu artificielle et difficile à résoudre manuellement, c'est la façon dont les requêtes sont produites : d'un côté, on considère (par nécessité) que toutes les productions de Reverb sont correctes, et d'un autre, on produit aléatoirement des faits probablement incohérents, qu'on suppose faux.

Contributions

J'ai commencé par explorer le protocole avancé par Angeli et Manning (2013), dans l'optique de filtrer les extractions bruitées. Après quelques expériences, mon

directeur de recherche a suggéré de comparer un modèle de langue, plus simple, à leur approche, ce que j'ai implémenté avec la librairie KenLM. À fin de comparaison, j'ai réimplémenté l'essentiel de la méthode d'Angeli et Manning (2013). J'ai ensuite mené les expériences autour du protocole d'évaluation, et écrit l'article. Plus tard, j'ai mené d'autres expériences sur les mêmes données, donnant lieu aux sections 5.4.4 et 5.4.5. Mon collègue Fabrizio Gotti a participé à l'exercice pour cette dernière section. J'ai enfin produit une affiche, présentée par mon collègue Abbas Ghaddar à la conférence LREC en mon lieu.

Impact

Prédire si un nouveau fait est plausible ou non est intéressant, pour affiner le processus d'extraction, purifier une base de connaissances existante, ou étendre la réponse à des requêtes qui ne font pas partie de la base. Pour cette tâche, Angeli et Manning (2013) ont proposé une méthode d'évaluation qui consiste à discriminer des faits générés aléatoirement (vraisemblablement erronés), et des faits extraits de texte par une méthode d'extraction, vraisemblablement corrects.

Nous utilisons un modèle de langue pour cette tâche, qui arrive à des performances équivalentes, sur un jeu de données similaire. Le modèle de langue est plus facile à entrainer à l'aide d'outils existants, à stocker et diffuser, et répond plus rapidement à des requêtes.

Nous tournons notre attention sur le protocole d'évaluation en lui-même, pour constater que la façon dont les exemples négatifs sont générés a un impact considérable sur la difficulté de la tâche.

Nous analysons les erreurs de notre système, et elles sont raisonnables : certains triplets générés artificiellement (considérés comme faux) sont en réalité plausibles, et certains triplets véritablement extraits (considérés comme vrais) sont erronés. Les erreurs du modèle de langue sont en partie dues à ce bruit, et en partie explicables de par son fonctionnement. Un système plus élaboré tenant compte des entités nommées et de leurs types permettrait d'atteindre de meilleures performances.

Pour quantifier la part de bruit dans la génération des données, nous avons mesuré les performances d'annotateurs humains. Ceux-ci réussissent à classifier correctement 80% des faits, ce qui est légèrement supérieur aux performances obtenues par les algorithmes (environ 75%).

5.1 Introduction

Much recent effort has been put into building Knowledge Bases (KBs), either manually curated (Freebase[1], Cyc[2]) or automatically produced (YAGO[3], Knowledge Vault[4]), ranging from logically consistent linked data in OWL (SUMO[5]) to little-structured sets of textual relations extracted from text (NELL[6]) with Open IE systems (Reverb[7], Ollie[8], ClausIE[9], Stanford Open IE[10], CSD-IE[11]). However large they may be, typical KBs are greatly incomplete, and many relevant facts are missing (West et al., 2014).

Because an exhaustive coverage of the information that ought to be part of the KB is a very desirable feature, KB completion (inference of missing facts from known ones) is a rapidly growing field (West et al., 2014; Nickel et al., 2015; Wang et al., 2015; Toutanova et al., 2016).

In the context of Open Information Extraction (OIE), our aim is to assign a score to an arbitrary unseen query fact[12], judging its *plausibility* as a member of the KB. This task is important for three reasons : first, it extends the coverage of the existing KB probabilistically to any query, greatly improving upon the closed-world

[1]Bollacker et al. (2008)
[2]Lenat (1995)
[3]Suchanek et al. (2007)
[4]Dong et al. (2014)
[5]Pease et al. (2002)
[6]Mitchell et al. (2015)
[7]Fader et al. (2011)
[8]Mausam et al. (2012)
[9]Del Corro et Gemulla (2013)
[10]Angeli et al. (2015b)
[11]Bast et Haussmann (2013)
[12]A "fact" is a relation phrase linking two or more argument phrases. The arguments need not be named entities.

assumption that facts not known to be true are false. Second, as the extraction of information in the open domain is a relatively noisy process, a confidence score helps detecting extraction errors, and makes for higher-quality automatically generated KBs. Last, adjusting the confidence threshold of extracted facts allows to tune as desired the trade-off between precision and recall of the extraction process.

The task has attracted little attention since it was introduced by Angeli et Manning (2013), most that we know by Li et al. (2016). The authors propose to assign high KB-membership probability to facts that have *support facts* existing in the KB, which are similar to them (based on common phrases and word similarity).

We propose a new baseline for this task, in the form of a language model. Whereas the method of Angeli et Manning (2013) is fairly complicated to implement, and requires indexing the KB in various ways for intermediate computations, a trained language model is very compact, straightforward to implement and train, and fast to process requests at use time. We train the model on automatically extracted facts (including some noise) from the same corpus, i.e. the knowledge base, taken as a list of sentences. We experiment with language model features and show that a linear classifier gives good results at the task of recognizing actual extracted facts, perhaps unsurprisingly.

We then go back over the experimental protocol proposed by Angeli et Manning (2013) and consider the way negative examples are automatically generated. We find that this procedure has a significant impact on the difficulty of the task. At last, we go back to the goal of improving existing extractions by picking out the noise. Instead of an automatically generated test set, we measure the ability of the models to identify the remaining wrong extractions in the RV15M high-quality dataset (presented in Section 5.4.1).

5.2 Related Work

5.2.1 Knowledge Base Completion

Much work in Knowledge Base Completion (KBC) has been done in recent years (Bordes et al., 2013; Riedel et al., 2013; García-Durán et al., 2015; Feng et al., 2016; Trouillon et al., 2016), on tasks very similar to ours, mostly focusing on Freebase, and other such large manually curated KBs (WordNet, NCI-PID (Schaefer et al., 2008), etc.).

The major difference between our approach and most KBC works is the predefined schema of the KB. The arguments of the relations curated in Freebase are mostly named entities, and the relations to be gathered were defined when building the KB. FB15k, a popular Freebase dataset, covers 1345 predicates, though only 401 have more than 100 occurrences (Yang et al., 2014). NELL captures about 150 relations, and WordNet about 20. By contrast OIE seeks to extract all the relations expressed in text, resulting in hundreds of thousands of relation predicates (even though many are synonyms). The RV15M dataset used in this work has 660k distinct relation strings.

Toutanova et al. (2015) embed surface textual patterns in the same vector space as the KB relations, which is similar to the implicit embedding of all predicates in the same vector space as we do. Yet their work only predicts relations based on the 237 predicates of the FB15k-237 dataset, whereas we predict confidence scores for all relations, including predicates never seen during training.

5.2.2 Angeli and Manning (2013)

In this work we reexplore in depth the task set up by Angeli et Manning (2013), and it is their work that is most similar to ours. They seek to probabilistically extend a KB to arbitrarily any query fact, in the sense that any candidate fact has a KB-membership probability (or confidence score). Indeed, this is a sensible way of considering a knowledge base system that must perform inferences.

To this aim, they compare a query fact to a set of candidate support facts from

the KB. The candidate facts need to share 2 phrases (arguments or relation) with the query fact, and is allowed to differ by the third part. The query fact has a high KB-membership score if it is similar enough to its closest support facts (that is, the differing part is similar enough). The algorithm is as follows:

- Gather candidate facts that can support the query fact: candidates must share two of (argument 1, relation, argument 2) with the query, these phrases having at least the same head word. Stricter criteria are used as long as there is a sufficient number of candidates.
- Compute the distances between the query fact and each of its supporting facts, using 11 distance metrics, based on both distributional similarity and the WordNet thesaurus — cosine, Jaccard, etc.
- The highest similarities are used as weights in a linear classifier, whose job is to aggregate the similarity values across candidates and distance metrics.

5.2.3 OIE Systems Confidence

Contrarily to most OIE systems in which the confidence score is often an afterthought, Fader et al. (2011) went to great lengths to develop the confidence function of Reverb (see their Section 4.2). They manually labeled the extractions from 1000 Web sentences as correct or incorrect, and trained a classifier using features about the *original sentence* to assign a confidence score to the extraction process. The confidence function of Ollie is based on the frequency of the syntactic pattern that was used to extract a given fact. ClausIE simply returns the confidence score of the underlying dependency parser, as its rules rely directly on it.

In contrast to our work, the confidence scores produced by OIE systems rely on the original sentence from which they were extracted, and on how well the extraction procedures could handle the input sentences. Our goal is to judge the quality of query facts as they stand, regardless of the sentences they come from.

For instance, with the Chomsky sentence *Colorless green ideas sleep furiously* as input, past OIE evaluation would consider (*ideas, sleep, furiously*) to be a correct extraction, whereas the goal of our system is to reject any "fact" coming from that sentence on the grounds that they do not make sense.

5.3 Revisiting the Task Setup

Our end goal is to improve the OIE process by pruning out the erroneous or empty facts produced. We frame this as a classification problem, seeking to distinguish correct useful facts from ill-constructed or void statements.

5.3.1 Task Protocol

A KB is constructed by running an open information extractor over a textual corpus. Even though there is some noise, extracted facts are assumed to be correct, and samples of them are set aside from the KB to constitute positive examples of unseen facts for the classification task.

For negative examples, artificial facts are constructed by replacing one part of a genuine extraction with that of another. Let (a_1, r, a_2) and (a'_1, r', a'_2) be two genuine facts from the KB, then one negative example is picked between (a'_1, r, a_2), (a_1, r', a_2) and (a_1, r, a'_2). We will show in Section 5.4.3 that this choice is a very significant parameter of the experimental setup.

Classifiers are trained to discriminate the positive from the negative examples. The performance metric is the classification accuracy.

5.3.2 Approaches

ArgSim is a weak baseline for the task, measuring the cosine similarity between a_1 and a_2 in a fact. Arguments (effectively bags of words) are represented by the average of their individual words' embeddings. This performs well on ConceptNet, per Li et al. (2016), but captures little information on OIE facts. This is both reported by Angeli et Manning (2013) and replicated in our experiments.

The full algorithm of Angeli et Manning (2013), presented in Section 5.2.2, is denoted **AM-system** in Table 5.1. We reimplemented the **count** and **cos** methods used in their evaluation, which are both simplified versions of their approach (the former coarse, and the latter close in principle to the full-fledged scoring function).

As has been noted by Stanovsky et al. (2015), OIE output can be used as training material for other tasks such as text comprehension, word similarity and analogy. This is because OIE produces a distinctive intermediate representation of the sentence, from which complementary features (to that of dependency parse or lexical representations) can be extracted.

Moreover, in the confidence scoring function of Reverb, several features capture how completely the extraction covers the sentence's tokens. In short, the most typical correct extractions look like short declarative sentences, like (*Hudson*, *was born in*, *Hampstead*), or (*Hampstead*, *is a suburb of*, *London*). Then, it seems natural to train a language model on confidently extracted facts, and to expect from correct unseen extractions that they fit well and cause low perplexity. This implements the assumption that an unseen fact is plausible iff it resembles a short and natural-sounding sentence.

The most basic implementation of this idea (**LM-basic**) is to straightforwardly use the probability of concatenated $(a_1.r.a_2)$ as its score. Next, we notice that given the way negative facts are constructed, all arguments and relation spans are probable as they stand (since they come from genuine extractions). What makes the fact incorrect is that the parts do not fit together. Therefore, we use as a score the (log) probability of the whole fact minus that of each individual part. We call this model **LM-junctions**. With $\{a_1, r, a_2\}$ the fact expressing that the relation r holds between the two arguments a_1 and a_2, and p the language model probability function : $\text{score}(\{a_1, r, a_2\}) = \log\, p(a_1.r.a_2) - \log\, p(a_1) - \log\, p(r) - \log\, p(a_2)$.

We trained the language models with KenLM (Heafield, 2011), using the knowledge base itself as a corpus, each fact being considered as a sentence. We trained 5-gram models (the default), doing as little parameter tuning as possible.

Further, we train a linear classifier based on linguistic modeling-based features

(**LM-SVM**). We implemented the SVM with scikit-learn (Pedregosa et al., 2011)[13]. It uses 20 features such as the individual log-probabilities of the various parts, the log-probabilities of various bigrams and trigrams focused on the argument-relation junctions, and arithmetic operations over those values.

The classifier is trained on 10k genuine extractions as positive examples, and 10k artificially constructed ones as negative examples, counts picked to be the same as those in (Angeli et Manning, 2013).

5.4 Experiments

5.4.1 Dataset

We used Reverb-15M, a shared[14] dataset of high-quality binary assertions extracted by Reverb on the ClueWeb09 corpus. In order to obtain a high-precision dataset, its authors filtered the extractions by Reverb's confidence function (with a 0.9 threshold), stop words, and frequency, along with certain syntactic criteria.[15] We used the normalized (lemmatized) version of the tuples. Taken as a text corpus, the RV-15M dataset is 98M tokens long.

Angeli et Manning (2013) used a similar set of extractions : the authors ran Reverb over ClueWeb09 themselves, filtered out extractions scoring under 0.5 per Reverb's confidence function, and retained the first billion extractions, which results in a KB of 500 million unique facts. Their dataset is thus larger, and noisier, than the one we used.

5.4.2 Classification

Table 5.1 shows the performance of our approach, along with the methods **count** and **cos** of Angeli et Manning (2013).

[13]We shortly experimented with Logistic Regression and Random Forest classifiers, giving slightly inferior results. We also tried several available SVM kernels, the default RBF kernel giving slightly better results than the others.

[14]Available at `http://reverb.cs.washington.edu/`

[15]See `reverb.cs.washington.edu/README_data.txt`

	Reverb-15M (AUPRC)	Reverb-500M†
Random	50.0 (0.500)	50.0
ArgSim	53.0 (0.283)	52.6
AM-count	61.6 (0.399)	52.3
AM-cos	64.2 (0.462)	70.6
AM-system		74.2
LM-basic	65.4 (0.471)	
LM-junctions	73.6 (0.589)	
LM-SVM	76.3 (0.628)	

Table 5.1 – Classification accuracy of the scoring methods on Reverb-15M, evaluated on the automatically generated test set. Area under the precision-recall curve is indicated in parentheses. † Results on Reverb-500M (a similar, larger and noisier dataset) published in (Angeli and Manning 2013), are reproduced for comparison.

The most elementary language modeling idea demonstrably captures some useful signal for the task. By focusing on the probability of the argument-relation junctions, the language model improves to near state-of-the-art performance. By training a linear classifier on top of the language modeling features, we can gain 3 additional points in precision, surpassing previous state-of-art performance.

Examples of correct tuples scored highly by our model are (*Austin Airways, was an airline based in, Canada*) and (*Children, are welcome in, the Curriculum Lab*).

Reversely, (*A knight, can turn into, a serious problem*) and (*Hand, be in, The Los Angeles Times*) are examples of incorrect tuples scored highly. The first argument of the former was originally *'A bad sunburn'* and the relation of the latter was *'has also written for'*. Both facts are plausible and show the limits of the automatic evaluation procedure (*be in* being accidentally close in meaning to *has also written for*, for a journalist and a newspaper).

Correct tuples scored poorly include (*A bounty Hunter, sent to kill ; Gust*) and (*Casey, also works on, pairNIC*). Our model finds high perplexity in infrequent proper nouns. A Named Entity Recognition module would certainly improve the model, so that *<PERSON> also works on <MISC>* is seen at training time (or

training data featuring the entities of interest if the KB focuses on a certain domain). Bad tuples correctly identified include (*81.45% of total wagers, can also avail the services of, bettors*) and (*A trellis, can also refer to, permission of the artist a structure*), in which relation and second argument formerly were *'returned to'* and *'a structure'* respectively.

5.4.3 Impact of Negative Examples Sampling Method

One important task parameter is the way negative examples are constructed. With (a_1, r, a_2) and (a_1', r', a_2') two genuine extractions, then one of (a_1', r, a_2), (a_1, r, a_2'), or (a_1, r', a_2) will be used as a negative example. We examine the impact of choosing one of the former two (changing an argument) versus picking the latter (changing the relation[16]). In Figure 5.1, we vary the fraction of relation changes over argument changes (a "knob" of the task setup), and measure the ensuing precision of systems.

Angeli et Manning (2013) use a particular scheme : out of the 3 negative example candidates, they pick the one that has the largest cosine similarity with the original fact (a_1, r, a_2). This is with the stated purpose of training the classifier to discriminate between more similar examples, supposedly a better learning setup.

In practice, this means changing the relation in about 75% of cases, similarly to setting the knob at 0.75 in Figure 5.1 [17].

In practice, the fact most similar to the original tends to be the one with the swapped relation, because relational phrases are more often similar to each other than argument phrases. This in turn, we suppose, is because phrases are treated as bags of words and represented by their average word embedding, and relation phrases often share common verbs such as *be*, *make*, etc. and prepositions, whereas argument phrases are more distinct. Between $d(a_1, a_1')$, $d(r, r')$ and $d(a_2, a_2')$, $d(r, r')$

[16] Or, from a different perspective, changing *both* arguments with respect to the other fact, since facts used for this construction are picked at random.

[17] The difference is that our graph shows performance for a certain proportion of relation changes picked at random, whereas Angeli and Manning select a particular set of relation changes, that amounts to 75%. This is why points were made to be off the regression lines in the chart.

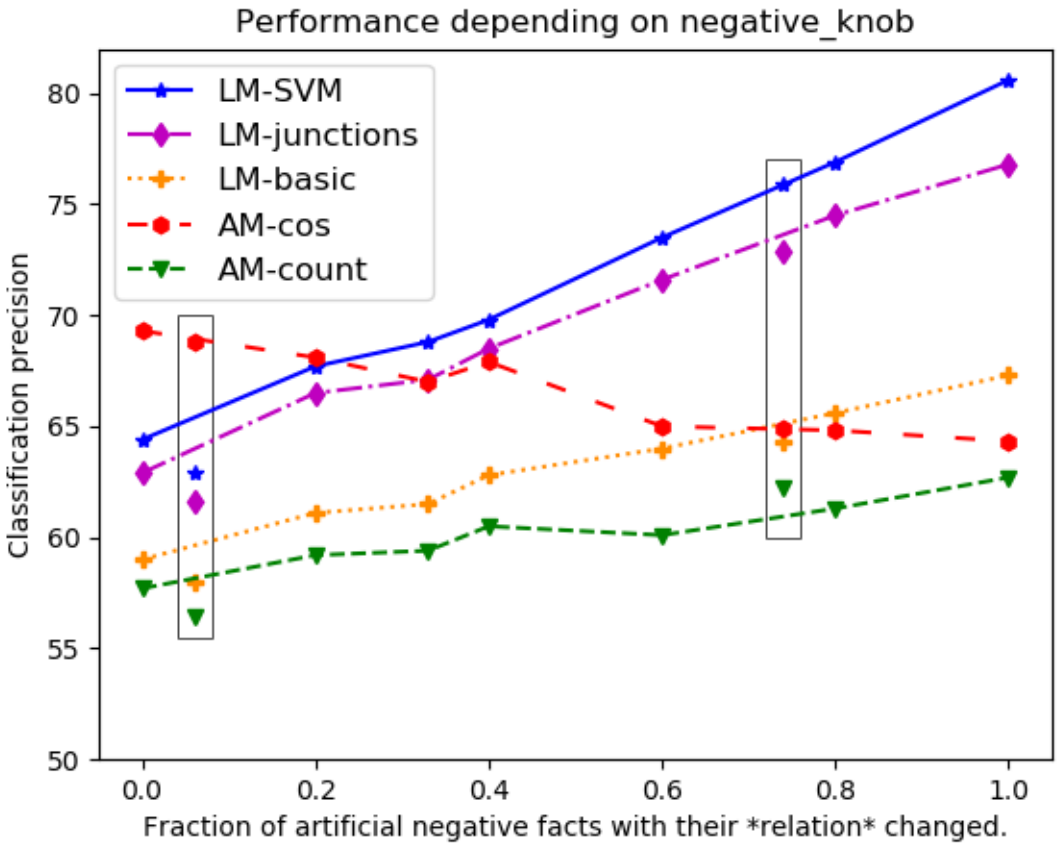


Figure 5.1 – Performance depending on the proportion of negative examples where the *relation* was changed, rather than one of the arguments. 0.33 corresponds to picking at random, which we recommend. The method used by Angeli and Manning is not equivalent but corresponds to a value close to 0.75.

is the largest of the three in only 7% of cases (this choice of negative examples is the column of values at 0.07 on the x-axis).

Looking at Figure 5.1, we can see that **swapping the relation phrase makes the task easier** : except for AM-cos, all systems perform better, in a linear fashion. From a language modeling perspective, this is easy to explain : swapping the relation introduces two breaks inside the short sentence, where words may not fit together, instead of just one when an argument is replaced. When building artificial facts in this way, we recommend picking one of the three candidate negatives at random (i.e. setting the knob on 0.33). For more difficult learning tasks[18], further research could leverage other sources of information to produce better distractors as negative examples. This could be argument types, or using the fact that certain relations have only one correct value (e.g. *<person>, be born on, <date>*), so that differing values are known to be false.

5.4.4 Results on Manually Annotated Tuples

Our classification task up to now assumed all extracted facts to be correct, and all randomly generated facts to be wrong. In practice this is not always the case, as some noise remains in the high-quality Reverb dataset, and some randomly assembled facts turn out to have interpretations that make them right, at least plausible. For instance (*a knight, can turn into, a serious problem*) would be true in medieval times, or in chess commentary.

Therefore, in an effort to come nearer the original task of recognizing impossible from plausible OIE extractions, we experiment with a small set of manually annotated facts. It is a delicate issue to decide what constitutes a correct extraction: see for instance Section 3.1 in Stanovsky et Dagan (2016). We annotated 430 tuples manually as:

- good extractions (45%) : capturing *some* information, and at least sometimes true. Some examples are (*Blackberry picking, is a great introduction*

[18]For instance, distinguishing facts that are actually true in the world from facts that are false or sometimes false, or from facts that are merely plausible.

	Reverb-15M-manual
ArgSim	63.5
AM-count	47.7
AM-cos	57.5
LM-basic	52.6
LM-junctions	51.3

Table 5.2 – Scores of unsupervised models on the manually annotated test set.

to, foraging), (*A new computer ; only costs around ; 500$*), and (*Blood pressure, is influenced by, dietary fibre*).

- unsure (30%) : typically only true in their original context which is lost, neither wrong nor useful in a vacuum. For instance (*Alfred, was against, garages*), (*Archaeology, answers this question with, confidence*), or (*Access ; is limited to ; official business*).

- incorrect extractions (25%) : nonsense or false, often due to upstream parser errors or noisy source text. E.g. (*5 Mts, Walk From, Wembley Stadium*), (*Atomic Kitten, released, the*) and (*A whole day ; set aside for ; literary pursuits*).

We annotate the tuples regardless of the sentences from which the facts were extracted : we label the facts as they stand and not the extraction process. The *unsure* labels (covering 25% of examples) were ignored and models had to classify correct and incorrect extractions. The negative examples were oversampled to balance the dataset.

The models used are the same as in Section 5.4.2, except that the SVM cannot be trained, as there is no training data (all facts are positive examples in the automatic task setup). Results are presented in Table 5.2. Among genuine extractions (all scored highly by Reverb's confidence function), current models have a hard time recognizing ill-formed or nonsensical extractions, and perform worse than on the automatically generated test set.

Some incorrect facts are scored highly because some particular pattern was often repeated over the Web and systematically misinterpreted by the OIE system. One example is *"Cross listed with <another class>."*, occurring frequently on certain university curriculum pages, from which Reverb extracts (*Cross, listed with,* e.g. *AIST2340*). This is correct from a language modeling perspective, even though it's a wrong fact.

5.4.5 Human performance

In an attempt to gauge the impact of false positives (genuine extractions by Reverb that turn out to be erroneous) and false negatives (artificially assembled facts that turn out to be plausible), two human annotators, both NLP practitioners and including one author, manually performed the task on 200 facts. Half of them were not lemmatized. As in Table 5.1, the negative sampling method was that of (Angeli et Manning, 2013).

Both judges achieved just 80% accuracy at discriminating genuine and artificial facts, on both the lemmatized and unlemmatized versions of the task. Agreement was also 80%.

Examples of highly ambiguous facts on which both annotators were mistaken include:

- (*Zaire ; will be maimed by ; betrayal*)
- (*Jean ; is a native of ; New York*)
- (*Peas ; here take advantage of ; ringtones*)
- (*Cooking school ; have changed a bit in ; the Los Angeles area*)

The first and third are genuine extractions (positive examples in the automated task) while the second and fourth were assembled at random (negative examples). Such facts constitute 10% of the test set.

Examples of facts on which annotators disagreed (one being mistaken) are:

- (*shimer college ; be establish in ; mt*)
- (*the north node ; will be in ; pisces*)
- (*Specific attention ; will be given to ; THE MAN*)
- (*Zoroastrianism ; is in even ; worse shape*)
- (*Kara ; is vice president of ; buying*)

Out of those five, only the third is artificial, and others are genuine extractions (the first two being lemmatized, as in the automatic evaluation). Such facts constitute 20% of the test set.

Overall, it is as if 60% of the automatically generated test-set was reliably recognizable as genuine or artificial, humans performing no better than chance on the remaining 40% (hence the resulting 80% performance).

5.5 Conclusion

We revisit the task of judging the plausibility of a new candidate fact to extend a knowledge base, in the context of OIE — arbitrary relations between unrestricted noun phrases. Correctly assessing the validity of an unknown fact is highly valuable, both as a way to refine KBs built automatically, and to implicitly enhance finite stored knowledge for it to answer an order of magnitude more queries.

We propose a new baseline for this task, based on language modeling, which achieves state-of-the-art performance. Indeed, archetypal correctly extracted information resembles short declarative sentences. We show that the way artificial negative examples are sampled has a large and robust impact on the difficulty of the task. We manually find genuine yet incorrect extractions and show that while our system does capture some useful signal, picking up wrong extractions from a high quality dataset remains a challenging task.

We examine the sort of facts that our model gets "right" despite the test set generation method being wrong, and the sort of facts on which it performs poorly.

Future work extending the language model beyond off-the-shelf programs with named-entity recognition would improve its performance.

CHAPITRE 6

PRÉDICTION DE LIENS DANS LES BASES DE CONNAISSANCES OUVERTES

William Léchelle, Jason Jiechen Wu et Phillippe Langlais. Link Prediction in Open Domain Knowledge Bases. (en préparation).

Contexte

La façon dont les modèles de prédiction de liens sont évalués est intéressante : un sous-ensemble de la base est mis de côté comme ensemble de test. Pour chaque triplet de test, on fournit « sujet ; relation ; ? » au modèle, qui doit prédire l'objet de la relation (et réciproquement avec « ? ; relation ; objet »). Cette prédiction prend la forme d'un classement de tous les objets possibles (contenus dans la base), et la performance du modèle est le rang qu'il a assigné à la réponse correcte, d'après le triplet de test d'origine.

Dans ce chapitre, on assimile la fonction de confiance du paradigme de l'EIL à la fonction de classement intrinsèque aux modèles de prédiction de liens. À la suite des travaux de Bordes et al. (2011, 2013), de nombreux modèles ont été proposés pour modéliser les graphes de connaissances. Cette modélisation consiste à être capables de différencier des faits inconnus mais plausibles, de suggestions fausses ou déraisonnables. Dans le classement mentionné plus haut, les objets du haut du classement qui viendraient plausiblement compléter une requête « sujet ; relation ; ? » ont un score élevé, le bas du classement, un score faible. Comme la fonction de confiance, ce score permettrait de filtrer les extractions produites par l'EIL pour purifier les connaissances résultantes.

Bien que ces modèles de prédictions de liens soient généralement appliqués à des bases de connaissances de taille modérée, on applique ici TransE, un modèle classique de complétion, à la base de données Reverb-15M, résultant d'une soigneuse

curation des extractions du système Reverb sur le corpus Clueweb.

L'idée m'était venue dès le début de mon doctorat de simplifier les bases de connaissances produites par l'EIL, en simplifiant par synonymie la grande varitété de formes de surface de relations produites par les extracteurs, en des relations plus symboliques, dont le sens serait désambiguisé. Chaque relation symbolique couvrirait et représenterait plusieurs formes de surface fortement synonymes, et serait plongée dans un espace vectoriel aux fins de prédictions et computations. Malheureusement, la qualité des rassemblements est très difficile à quantifier, comme le démontre la complexité de l'évaluation de projets similaires comme OntoLearn (Velardi et al., 2013) ou RELLY (Grycner et al., 2015). L'objectif initial (comme au chapitre précédent, dans le cadre de la classification de faits candidats) était de revenir informer l'étape d'extraction de faits des systèmes d'EIL, grâce aux rassemblements produits sur la base d'une première extraction. Sans façon d'évaluer précisément la qualité de l'extraction, il était toutefois impossible de mesurer l'impact d'un tel apprentissage, et cette idée est restée en suspens pendant que je travaillais à une évaluation automatique de l'EIL.

Contributions

Je me suis familiarisé avec la littérature en complétion de base de connaissances assez tôt, grâce à la vague de popularité du modèle TransE, avant d'explorer l'idée d'appliquer leurs techniques au paradigme de l'EIL.

Sous ma supervision, Jason Wu a adapté la librairie OpenKE pour réaliser les expériences avec les modèles TransE et DistMult. J'ai défini les plages d'hyperparamètres que nous avons explorées, dans la limite du temps nécessaire au calcul de chaque expérience (ces modèles étant particulièrement longs à entrainer). Philippe Langlais a eu l'idée d'explorer la différence de performances entre les relations plus et moins fréquentes. J'ai conçu et implémenté les regroupements de relations, ainsi que les variantes dans la façon de les construire. J'ai mené les analyses de résultats, avec l'assistance de Jason. Enfin, j'ai écrit l'article.

Impact

En étendant l'utilisation des modèles de prédiction de liens aux bases de connaissances produites par EIL, nous envisageons d'employer un tel modèle comme filtre contribuant à l'extraction (autrement dit, comme remplacement à la fonction de confiance).

Ce qui ressort de cet article, c'est la grande variabilité dans la performance du modèle TransE : d'excellent à prédire les objets des relations fréquentes (connaissant le sujet), à moyen pour prédire les sujets (connaissant l'objet), jusqu'à être moins performant qu'un modèle simpliste à prédire les objets des relations rares.

Deux conclusions apparaissent. D'abord, qu'il serait effectivement viable d'employer un tel modèle comme fonction de confiance, dès lors qu'une petite base de connaissances (d'amorçage) est disponible pour l'entrainer (celle-ci devant être du même domaine et raisonnablement représentative de la base de connaissance plus grande que l'on cherche à construire). De la même façon que la traduction automatique valide des candidats de traduction provenant de la source, par un modèle de langue de la langue cible, un système générique d'extraction d'information validerait des faits candidats provenant de la source, par un modèle de connaissances du domaine ciblé.

Ensuite, qu'il serait utile de se pencher davantage sur la performance « non supervisée » des modèles de prédiction de liens, dans le cas des grandes bases de connaissances riches en éléments peu connectés. Il est bien connu que les relations courantes et les entités fréquentes sont raisonnablement faciles à modéliser (et les performances de prédiction sur Freebase et WordNet s'améliorent d'année en année), mais le plus grand défi consiste à extraire et prédire les relations marginales.

6.1 Introduction

Link Prediction (also called Knowledge Graph Completion) is a well-established research domain, in the case of well-structured Knowledge Bases (KBs) such as Freebase, WordNet, YAGO or DBpedia. Following the early works of Bordes et al.

(2011, 2013), a myriad of models have been developed to model the knowledge graphs, that is to say, to differentiate unknown but plausible facts, from wrong or unreasonable suggestions. This allows to validate new candidate facts into the KB, and to make suggestions for the inclusion of new likely pieces of information, as well as to answer as of the probability of an inference query. It is necessary, because even large KBs tend to be sparse, containing a small fraction of the set of facts they aspire to cover (West et al., 2014).

In the continuity of Angeli et Manning (2013) and Li et al. (2016), we consider this task in the context of Open IE. Open IE – open domain information extraction – extracts relational tuples from large corpora, in a scalable way and without domain-specific training (Mausam et al., 2012; Del Corro et Gemulla, 2013). Approaches range from the simple Reverb system (Fader et al., 2011), using regular expressions over POS tags to detect verbal phrases and their arguments, to more complex methods involving the splitting of clauses and dependency parsing (Mausam et al., 2012; Del Corro et Gemulla, 2013), or inspired by semantic role labeling (Mausam, 2016). Recent works (Soderland et al., 2013; Fader et al., 2014; Stanovsky et al., 2015) successfully employ Open IE output as a tool enhancing text understanding. Niklaus et al. (2018) recently published a survey of Open IE.

The KBs produced by Open IE systems are larger than that traditionally studied in Link Prediction by an order of magnitude, and are subject to a higher amount of noise (in the form of unclear, or only sometimes true, facts). FB15k, a popular Freebase dataset, covers 1345 predicates, though only 401 of them have more than 100 occurrences (Yang et al., 2014), and only 237 are not near-synonym or strict inverse of one another (Toutanova et Chen, 2015). NELL captures about 150 relations, and WordNet about 20. By contrast, the RV15M Open IE dataset used in this work has 660k distinct relation strings (even though many are synonyms).

In practice, usage of Open IE productions, often requires to establish a mapping from the free-ranging textual relations to the ("symbolic") relations used by another application. One enlightening example of this is the task of TAC-KBP (Surdeanu, 2013), where one team adapted an Open IE engine to the Knowledge Base Slot

	#entities	#relations	#facts
WordNet	40 943	18	151 442
FB15k	14 951	1345	592 213
FB15k-237	14 541	237	272 115
Aristo KB	N/A	1605	282 594
RV15M	2 263 915	664 746	14 728 268
DefIE	2 398 982	255 881	20 352 903

Table 6.1 – Knowledge Base sizes

Filling task (Soderland et al., 2013). The authors compare what can be achieved with only minimal manual pattern writing (very few patterns per relation), to a slightly more involved effort covering more textual schemas (resulting in slight improvements).

In this work, we apply the established evaluation protocol of Link Prediction models, to the case of KBs produced by Open IE systems. We conduct experiments with the TransE model of Bordes et al. (2013), and compare it to a baseline. In order to consolidate the sparse training data available (textual relations have an average of 22 occurrences, and the median number is 4), we propose to cluster the textual relations by synonymy, as a preprocessing step for both the baseline and embedding models.

6.2 Related Work

Since the foundational work of Bordes et al. (2011) and the TransE (Bordes et al., 2013) model that followed it, a plethora of similar embedding models have been developed. Kadlec et al. (2017) list 29 such methods[1] and provide a full list of references. Nickel et al. (2015) provides a review of these models.

Interestingly, the works by Toutanova et al. (2015, 2016) learn to represent textual relations, found between entity pairs in the ClueWeb12 corpus, as well as knowledge base relations and entities.

[1] TransH, TransR, TranSparse, NTN, DistMult, RESCAL, TransD, TransG, CTransR, RTransE, PTransE, STransE, etc.

Li et al. (2016) frame the problem of extending the coverage of the commonsense knowledge base ConceptNet[2], as that of knowledge base completion. Unlike previous works but following ConceptNet, their model allows for *arbitrary phrases* to be tuple arguments, rather than solely entities as in Freebase. While our goals are identical to theirs, we extend the methodology to the larger and less curated knowledge base produced by Reverb, that contains many more relations (ConceptNet only contains 34 relations, 22 of which appear in their test set).

6.3 Methods

Evaluation protocol

A sample is left aside of the knowledge base for testing purposes. For each relation (arg_1, rel, arg_2) in the test set, the model is provided with the partial fact (arg_1, rel, ?) as a query, and must predict the correct arg_2. In practice models are made to *rank* all possible arg_2s, and the model performance is measured by the rank of the original answer[3]. Symmetrically models must also predict arg_1s from (?, rel, arg_2) queries.

Evaluation metrics typically are the Mean Rank (MR) and Mean Reciprocal Rank (MRR) of correct predictions. The mean rank is most affected by the worse predictions (biggest model mistakes), whose very high rank bring the average down. The mean reciprocal rank, in turn, is most affected by the best predictions, that contribute most to the average. We will report the head (predicting arg_1), tail (arg_2), and average mean rank for our experiments, as the reciprocal rank is not significative.

[2](Speer et Havasi, 2012) - `http://conceptnet5.media.mit.edu/`

[3]Typically, other correct answers from the knowledge base that the model might rank higher than the original one are excluded for scoring purposes, this is the "filtered" evaluation setup, as opposed to "raw", in which they are retained. This distinction makes no difference for this work as the ranks reported are much higher than the number of alternative answers.

Dataset

As a knowledge base, we used RV15M[4], a shared dataset of high-quality binary relations extracted by Reverb on the ClueWeb09 corpus. In order to obtain a high-precision dataset, its authors filtered the extractions by Reverb's confidence function (with a 0.9 threshold), stop words, and frequency, along with certain syntactic criteria. We used the normalized (lemmatized) version of the tuples. The dataset contains almost 15M triples, covering 2.2M entities linked by 660k textual relations.

Since many nominally distinct relation strings really represent the same semantic relation, we looked to cluster them together, what is detailed in Section 6.4.3.

6.4 Experiments

6.4.1 Frequency-based baseline

The simplest baseline consists in always predicting entities in their order of decreasing frequency in the train set, regardless of the query. This is the "overall frequency" model. A better frequency-based baseline predicts entities that were seen co-occurring with the query relation, in the train set, in their order of decreasing co-occurrence frequency with the relation[5].

The results of baseline models are presented in Table 6.2. Taking the relation into account when computing frequencies improves mean rank by about 8%. Entities that were not seen at all during training are considered to have the rank `number of entities in training KB +1`. This happens for 4% of cases in the test set, and weighs heavily in the final result.

This approach is naturally enhanced by a clustering of synonym relations, that drastically improves the frequency counts for scarcely occurring relations, and enriches those of the frequent relations with the variety of less frequent synonym

[4]Available at `http://reverb.cs.washington.edu/`

[5]Note that the target entity cannot co-occur with both the query argument and relation, since the test triple has been removed from the training KB.

Model	arg_1 MR	arg_2 MR	Average
Overall frequency	169512	237961	203736
Frequency baseline	**157385**	**218048**	**187716**

Table 6.2 – Mean Rank performance of the baseline link prediction model. The total number of entities is 2263915, so the average MR is about 8%. This is comparable to the `Unstructured` baseline used by Bordes et al. (2013), which obtains a raw MR of 1074 out of the 14951 entites of Fb15k, about 7%.

forms. Such clusters will be introduced in Section 6.4.3.

6.4.2 Embedding Model

We used the TransE model (Bordes et al., 2013), that initiated the recent interest in Knowledge Base Link Prediction. This model learns low-dimensional vectors for all entities and relations, assuming and optimizing for the fact that relations are *translations* in the embedded space, and therefore if (a_1, r, a_2) is a true fact, $\widehat{a_2} \approx \widehat{a_1} + \widehat{r}$, where $\widehat{a_1}$ is the embedding of a_1.

We used the TensorFlow implementation of OpenKE[6] (Han et al., 2018) for our experiments.

	arg_1 MR	arg_2 MR	Average
Baseline	**157385**	218048	187716
TransE	177604	**187844**	**182724**

Table 6.3 – TransE model performance, after parameter tuning. The best parameters in our experiments were `learning rate` = 0.01, 10^5 mini-batches (`batch size` = 147), trained for 500 epochs.

Once tuned, TransE performs on average slightly better than the baseline, being superior at ranking the second argument, and less good at predicting the first one. The most important parameters are the learning rate, the batch size (controlled by the number of batches in our implementation), and the optimization method. Our best results were achieved with a learning rate of 0.01, 10000 batches (corresponding to a batch size of 147), using the Adagrad optimization method. Both SGD and

[6] `github.com/thunlp/openke`

Adam gave worse results than Adagrad. Training long past 50 epochs might have been very slightly beneficial, but at a large cost in processing time. Embedding size doesn't seem to affect the results and we conducted most experiments with an embedding size of 100.

6.4.3 Relation Clusters

Because Open IE relations are directly extracted from text, a staggering amount of surface forms are produced that often do not constitute semantically distinct relations. For instance, a third of surface forms contain an adverb, most of which do not significantly contribute to the relation meaning (e.g. *danced informally at*, *was particularly weakened by*, *may also occur in*). In half of those cases, the similar relation without the adverb (*danced at*, *was weakened by*, *may occur in*) is also present in the knowledge base.

In order to build our clusters, we first represented each relation by the average of its word embeddings. We used the PARAGRAM word embeddings of Wieting et al. (2015a,b), complementing the 50k Paragram-Phrase XXL word embeddings with the 1.7M-large Paragram-SL999 dataset, as recommended by the authors. Remarkably, the PARAGRAM embeddings were trained on a paraphrase database, and with the purpose of being easily combined into phrase embeddings by the averaging operator, which makes them particularly suited for our purpose.

Cluster configuration	#relations	coverage of relation occurrences	minimum relation frequency	#clusters (k)
C1	228216	90%	5	2282
C2	585118	99%	2	10000
C3	585118	99%	2	50000
C4	585118	99%	2	1000

Table 6.4 – **Cluster parameters.** We experimented with varying numbers of clusters, after increasing the coverage to include more than 99% of textual occurrences (corresponding to 88% of relation types), from an initially lower coverage.

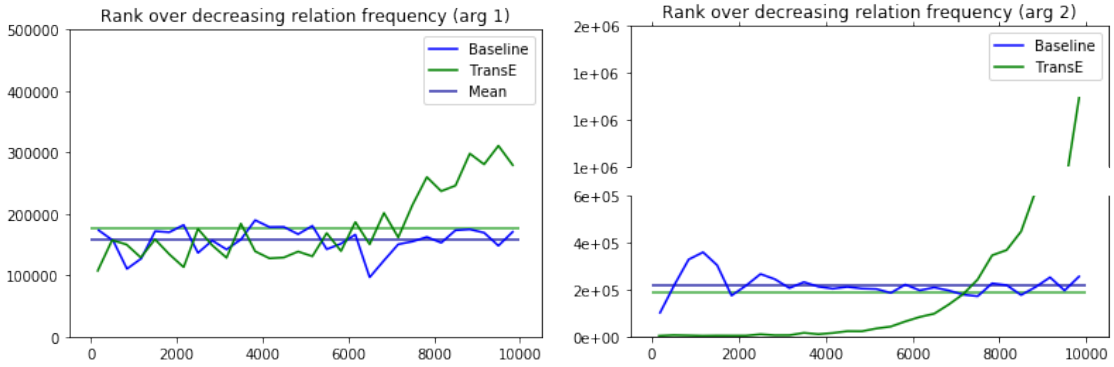
We then clustered the phrase embeddings with the k-means algorithm, imple-

mented with sklearn[7]. Phrase vectors were normalized to unit length, for clustering by Euclidian distance, which the implementation supported, to be equivalent to clustering by cosine distance, which is better suited to NLP. We experimented with several clustering configurations, detailed in Table 6.4, overall reducing the six hundred thousand original relation phrases to a few thousand relation clusters. For prediction, all relations from a cluster are replaced by a cluster representative, prior to training.

Model	arg_1 MR	arg_2 MR	Average
Baseline	157385	218048	187716
Baseline + C1	152629	211438	182033
Baseline + C2	153030	212762	182896
Baseline + C3	153953	214344	184148
Baseline + C4	**151076**	209039	**180057**
Best TransE	177604	**187844**	182724
Best TransE + C	167051	201573	184312

Table 6.5 – Impact of relation clusters on link prediction. The baseline model benefits from a simplification of the relation set. Fewer clusters improve the baseline most. The TransE model improves its subject predictions, but that is at the expense of worse object prediction (with all cluster configurations tested).

The baseline model benefits from a simplification of the relation set, which results in slight gains. Fewer clusters (more simplification of the relation set) help the baseline model most. After relation clustering, the TransE model makes better subject predictions, at the expense of worse object prediction (this happened with all cluster configurations, the best of which is reported in Table 6.5). We believe this happens because some clusters will aggregate relations differing only by the final preposition (e.g. *go to* and *go by*), losing the preposition information, which is a good predictor of the object.



(a) arg_1 (b) arg_2

Figure 6.1 – Rank of correct completion depending on relation frequency (lower is better). On the x-axis, the elements of the test set are sorted by decreasing relation frequency (and binned together to smooth the curve). The baseline model performs averagely across all relations. TransE, on the other hand, predicts the most frequent relation very accurately (particularly for arg_2 prediction), but performs poorly on rare relations.

6.5 Analysis

6.5.1 Distribution of ranks depending on relation frequency

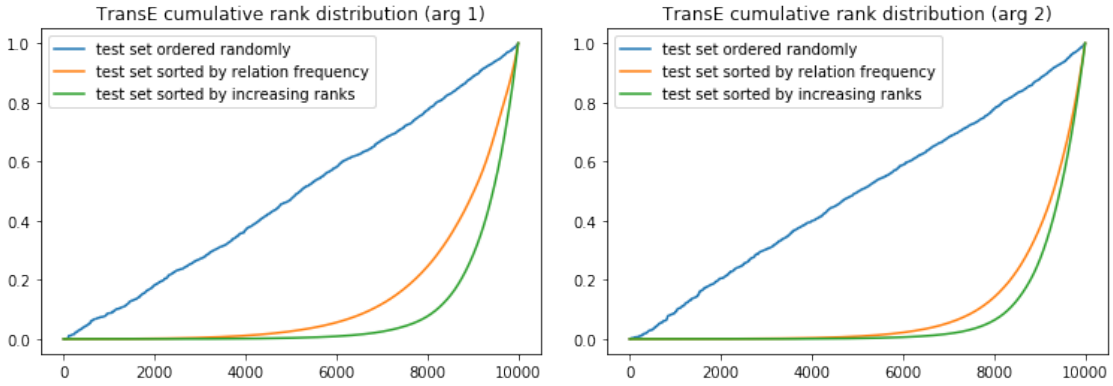
The frequency-based baseline model ranks the first argument, given a relation, best, and is less apt at ranking the second argument. It performs about as well across the range of frequent to rare relations. The TransE model is more imbalanced. At ranking the second argument in particular, it performs remarkably well on the most frequent half of relations, and very poorly on the fifth of relations that are most rare. To a lesser extent, it displays the same pattern on the prediction of the first argument.

This is remindful of what Toutanova et Chen (2015) find about the Freebase dataset : the most frequent relations are easily predictable, simply from the fact that other frequent relations co-occur and correlate very strongly with them (for instance the *"lives in"* and *"was born in"* relations between a person and their country, and family ties that are expressed in symetrical ways such as *"child of"*

[7] `scikit-learn.org` – (Pedregosa et al., 2011)

and *"parent of"*).

6.5.2 Cumulative contribution to total weight



(a) arg_1 (b) arg_2

Figure 6.2 – Cumulative sum of ranks, with varying ordering of the test set, for the TransE model. The bottom 20% of worst predictions weigh heavily in the balance, and correspond to the rarest relations.

The distribution of ranks is such that relatively few bad predictions weigh heavily on the final performance. 75% of test set examples have the correct answer ranked better than 100k, and only add up to 5% of the total rank weight. 50% of the weight comes from 5.5% of the test set with ranks above 1.1M. The worst predictions tightly match the least observed relations, especially for object prediction.

6.6 Conclusion

We broaden the scope of link prediction models, to the realm of Open Information Extraction. We applied the classical link prediction model TransE to RV15M, a knowledge base produced by Reverb, an Open IE system. RV15M is much larger, contains more diversified relations, and is noisier than the typical knowledge bases on which link prediction is run. We compare the embedding model with a simple frequency-based baseline model.

We show that TransE is comparably very good at predicting links mediated by

frequent relations, and drastically worse at learning rare relations. The baseline model, on the other hand, performs as well across the whole spectrum of relation frequencies. Compared to our baseline, TransE performs well at predicting the second argument of triples, whereas the baseline is better at predicting the first argument. Overall, TransE slightly outperforms the baseline.

Additionaly, in order to simplify the numerous relations produced by the extractor, we clustered synonym relations together, using phrase embeddings. In our experiments, this improved the baseline model slightly, but didn't benefit the TransE model.

CHAPITRE 7

CONCLUSION

> To the extent our tuple KB contains facts in this Reference KB (and under the simplifying assumption that these facts are representative of the science knowledge our QA application needs), we say our tuple KB is comprehensive.
>
> Allen Institute for Artificial Intelligence (AI2)

J'ai présenté les protocoles d'évaluation existants pour l'extraction d'information libre. Si l'évaluation manuelle est encore la façon la plus simple de mesurer la performance d'un système, cette méthode est empreinte de subjectivité et ne se prête pas bien à la reproduction par d'autres équipes. Plusieurs ensembles de données voient progressivement le jour, références auxquelles on peut comparer les productions d'un système d'extraction, permettant une évaluation automatique, objective et reproductible, des extracteurs.

Dans un premier temps, je contribue à cet effort, en produisant deux nouvelles références. La première s'appuie sur un système de questions-réponses, dans l'optique de questionner une phrase sur tous les éléments d'information qu'elle contient (chapitre 3). La deuxième se veut le reflet exact des productions d'un système d'extraction idéal (chapitre 4).

Ce qu'il faut retenir de ces travaux, c'est qu'il est désormais possible d'évaluer le rappel des systèmes d'EIL, pour peu que les limites de la tâche (coréférence, frontières de phrase, etc.) soient convenablement définies au préalable. Un effort raisonnable d'annotation manuelle, concentré sur les cas de figures les plus pertinents, permet de mesurer la performance des différents systèmes sur ces cas-là.

La création et l'utilisation automatique de ces données m'amènent à préciser les règles et les paramètres des procédures d'évaluation. Autant que possible, je diffuse le code qui me permet d'obtenir mes résultats, pour faciliter la reproduction de mes expériences. Nous publions également une proposition de directives d'ex-

traction (annexe A), visant à établir nettement ce qui est attendu d'un système d'EIL, au delà des principes de base. Une fois une telle spécification établie par la communauté des chercheurs, le prolongement naturel de ce travail consisterait à produire davantage de ces données de référence, à moindre coût : soit par production participative, soit par transformation de ressources lexicales existantes (par exemple FrameNet).

Dans un deuxième temps, et prenant inspiration d'Angeli et Manning (2013), je fais le lien entre prédiction de liens dans les bases de connaissances et validation de faits candidats à l'extraction. En EIL, peu d'importance est généralement consacrée à la fonction de confiance qui sert à cette validation : bien souvent, le score de l'analyseur syntaxique est utilisé par défaut. Je propose d'utiliser des modèles de connaissances comme sources alternatives de confiance : d'abord un modèle de langue (chapitre 5), puis un modèle de complétion de bases de connaissances (chapitre 6). L'enjeu principal devient celui de la modélisation des éléments rares : l'EIL visant à extraire la longue queue des relations les moins fréquentes, on ne peut pas s'appuyer sur leur connaissance a priori pour entrainer un modèle de validation.

Dans ces chapitres, je veux insister sur l'importance de la fonction de confiance. Grâce à une source exterieure d'information, les sorties des systèmes d'EIL peuvent être bonifiées. Il est encore difficile de concilier les travaux de modélisation de graphes de connaissances, et les vastes bases de connaissances produites par EIL, à cause du grand nombre supplémentaire d'entités, et de leur moindre définition, mais j'ai fait de premiers pas dans cette direction.

D'après ces développements, j'envisage des futurs systèmes d'EIL qu'ils soient conçus, analysés et évalués suivant ces deux étapes. Dans un premier temps, le système de patrons, quel qu'il soit, dans sa configuration la plus permissive, est évalué en rappel. Il est important de comprendre quelle proportion de l'information n'est *même pas considérée*, car c'est une borne supérieure de la performance absolue du système. Des générations successives d'extracteurs viseront à atteindre des faits de référence de plus en plus difficiles à capturer (capturés par des patrons statistiques de moins en moins fréquents, par exemple). Dans un deuxième temps, l'étape d'or-

donnancement de ces faits, le système de modélisation de graphes sémantiques, aura pour fonction de discriminer entre les faits plausibles et ceux qui sont invraisemblables ou inutiles. La combinaison des deux signaux orthogonaux « la structure de la phrase semble indiquer que (*A new computer* ; *only costs around* ; *500$*) » et « d'après d'autres informations sur le commerce des objets, il est vraisemblable que (*A new computer* ; *only costs around* ; *500$*) » est une solide indication de ce fait. À l'inverse, (*Atomic Kitten* ; *released* ; *the*) ou (*Archeology* ; *answers this question with* ; *confidence*) seront rejetés comme invraisemblables sur la base de filtres syntaxiques ou sémantiques. Un nouveau protocole d'évaluation adapté aux bases de connaissances produites par EIL permettra d'évaluer spécifiquement différentes fonctions de confiance : un modèle évolué devrait générer un grand nombre de faits candidats, plus raisonnables que ceux utilisés au chapitre 5, et dont l'évaluation de référence devrait être mieux maitrisée.

Enfin, il me semble nécessaire de tisser des liens entre résolution d'anaphores, liage d'entités (*Entity Linking*), et extraction d'information libre, comme dans la campagne d'évaluation TAC (volets KBP et EDL). L'évaluation de l'EIL est difficile à dissocier des premières tâches, et les systèmes d'extraction bénéficieraient d'une intégration avec des systèmes de liage.

Une des leçons majeures de ce travail est que hors contexte, la plupart des phrases perdent l'essentiel de leur sens. Les enjeux de temporalité, les anaphores et les déictiques couvrent une large fraction de l'information exprimée textuellement, et sont malheureusement hors d'atteinte des systèmes d'analyse actuels. L'extraction d'information est largement une excision d'information, et celle-ci demande une bonne prise en compte du contexte, qui doit s'y prêter, pour être réussie.

BIBLIOGRAPHIE

Eugene Agichtein et Luis Gravano. Snowball: Extracting relations from large plain-text collections. Dans *Proceedings of the Fifth ACM Conference on Digital Libraries*, DL '00, pages 85–94, New York, NY, USA, 2000. ACM. ISBN 1-58113-231-X. URL `http://doi.acm.org/10.1145/336597.336644`.

Alan Akbik et Jürgen Bross. Wanderlust: Extracting semantic relations from natural language text using dependency grammar patterns. Dans *Proceedings of the 2009 Semantic Search Workshop at the 18th International World Wide Web Conference*, SemSearch '09, pages 6–15, Madrid, Spain, April 2009. URL `http://ceur-ws.org/Vol-491/semse2009_7.pdf`.

Alan Akbik et Alexander Löser. Kraken: N-ary facts in open information extraction. Dans *Proceedings of the Joint Workshop on Automatic Knowledge Base Construction and Web-scale Knowledge Extraction*, AKBC-WEKEX '12, pages 52–56, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. URL `http://dl.acm.org/citation.cfm?id=2391200.2391210`.

Alan Akbik et Thilo Michael. The Weltmodell: A Data-Driven Commonsense Knowledge Base. Dans Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Hrafn Loftsson, Bente Maegaard, Joseph Mariani, Asuncion Moreno, Jan Odijk et Stelios Piperidis, éditeurs, *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, Reykjavik, Iceland, may 2014. European Language Resources Association (ELRA). ISBN 978-2-9517408-8-4.

Gabor Angeli, Sonal Gupta, Melvin Johnson Premkumar, Christopher D. Manning, Christopher Ré, Julie Tibshirani, Jean Y. Wu, Sen Wu et Ce Zhang. Stanford's Distantly Supervised Slot Filling Systems for KBP 2014. Dans *TAC-KBP*, 2015a.

Gabor Angeli et Christopher D. Manning. Philosophers are mortal: Inferring the truth of unseen facts. Dans Julia Hockenmaier et Sebastian Riedel, éditeurs,

Proceedings of the Seventeenth Conference on Computational Natural Language Learning, CoNLL 2013, Sofia, Bulgaria, August 8-9, 2013, pages 133–142. ACL, 2013. ISBN 978-1-937284-70-1. URL http://aclweb.org/anthology/W/W13/W13-3515.pdf.

Gabor Angeli, Melvin Johnson Premkumar et Christopher D. Manning. Leveraging linguistic structure for open domain information extraction. Dans *ACL*, 2015b.

Gabor Angeli, Julie Tibshirani, Jean Y. Wu et Christopher D. Manning. Combining distant and partial supervision for relation extraction. Dans *Proceedings of EMNLP*, pages 1556–1567, 2014.

Michele Banko et Oren Etzioni. The Tradeoffs Between Open and Traditional Relation Extraction. Dans *Proceedings of ACL-08: HLT*, pages 28–36, Columbus, Ohio, June 2008. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P/P08/P08-1004.

Hannah Bast et Elmar Haussmann. Open Information Extraction via contextual sentence decomposition. Dans *Semantic Computing (ICSC), 2013 IEEE Seventh International Conference on*, pages 154–159. IEEE, 2013.

Nikita Bhutani, H V Jagadish et Dragomir Radev. Nested Propositions in Open Information Extraction. Dans *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 55–64. Association for Computational Linguistics, 2016. URL http://www.aclweb.org/anthology/D16-1006.

Steven Bird, Ewan Klein et Edward Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly, Beijing, 2009. ISBN 978-0-596-51649-9. URL http://www.nltk.org/book.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge et Jamie Taylor. Freebase: A collaboratively created graph database for structuring human knowledge. Dans *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*, SIGMOD '08, pages 1247–1250, New York, NY, USA, 2008.

ACM. ISBN 978-1-60558-102-6. URL http://doi.acm.org/10.1145/1376616.1376746.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston et Oksana Yakhnenko. TransE: Translating Embeddings for Modeling Multi-relational Data. Dans C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani et K. Q. Weinberger, éditeurs, *Advances in Neural Information Processing Systems 26*, pages 2787–2795. Curran Associates, Inc., 2013. URL http://papers.nips.cc/paper/5071-translating-embeddings-for-modeling-multi-relational-data.pdf.

Antoine Bordes, Jason Weston, Ronan Collobert et Yoshua Bengio. Learning structured embeddings of knowledge bases. Dans *Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence*, AAAI'11, pages 301–306. AAAI Press, 2011. URL http://dl.acm.org/citation.cfm?id=2900423.2900470.

Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam Hruschka et Tom Mitchell. Toward an architecture for never-ending language learning, 2010. URL http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/view/1879.

Matthias Cetto, Christina Niklaus, André Freitas et Siegfried Handschuh. Graphene: Semantically-linked propositions in open information extraction. Dans *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2300–2311. Association for Computational Linguistics, 2018. URL http://aclweb.org/anthology/C18-1195.

Janara Christensen, Mausam, Stephen Soderland et Oren Etzioni. An analysis of open information extraction based on semantic role labeling. Dans *Proceedings of the Sixth International Conference on Knowledge Capture*, K-CAP '11, pages 113–120, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0396-5. URL http://doi.acm.org/10.1145/1999676.1999697.

Marie-Catherine de Marneffe, Timothy Dozat, Natalia Silveira, Katri Haverinen, Filip Ginter, Joakim Nivre et Christopher D. Manning. Universal stanford dependencies: A cross-linguistic typology. Dans *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA), 2014. URL http://www.lrec-conf.org/proceedings/lrec2014/pdf/1062_Paper.pdf.

Luciano Del Corro et Rainer Gemulla. Clausie: Clause-based open information extraction. Dans *Proceedings of the 22Nd International Conference on World Wide Web*, WWW '13, pages 355–366, Republic and Canton of Geneva, Switzerland, 2013. International World Wide Web Conferences Steering Committee. ISBN 978-1-4503-2035-1. URL http://dl.acm.org/citation.cfm?id=2488388.2488420.

Xin Luna Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, Ni Lao, Kevin Murphy, Thomas Strohmann, Shaohua Sun et Wei Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. Dans *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014*, pages 601–610, 2014. URL http://www.cs.cmu.edu/~nlao/publication/2014.kdd.pdf. Evgeniy Gabrilovich Wilko Horn Ni Lao Kevin Murphy Thomas Strohmann Shaohua Sun Wei Zhang Geremy Heitz.

Oren Etzioni. Search needs a shake-up. *Nature*, 476(7358):25–26, 2011. URL https://doi.org/10.1038/476025a.

Oren Etzioni, Michael Cafarella, Doug Downey, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld et Alexander Yates. Unsupervised named-entity extraction from the web: An experimental study. *Artif. Intell.*, 165(1):91–134, juin 2005. ISSN 0004-3702. URL https://doi.org/10.1016/j.artint.2005.03.001.

Oren Etzioni, Anthony Fader, Janara Christensen, Stephen Soderland et Mausam.

Open Information Extraction: The Second Generation. Dans *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume One*, IJCAI'11, pages 3–10. AAAI Press, 2011. ISBN 978-1-57735-513-7. URL http://dx.doi.org/10.5591/978-1-57735-516-8/IJCAI11-012.

Anthony Fader, Stephen Soderland et Oren Etzioni. Identifying relations for open information extraction. Dans *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1535–1545, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-937284-11-4. URL http://dl.acm.org/citation.cfm?id=2145432.2145596.

Anthony Fader, Luke Zettlemoyer et Oren Etzioni. Open question answering over curated and extracted knowledge bases. Dans *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '14, pages 1156–1165, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2956-9. URL http://doi.acm.org/10.1145/2623330.2623677.

Jun Feng, Minlie Huang, Yang Yang et Xiaoyan Zhu. GAKE: graph aware knowledge embedding. Dans Nicoletta Calzolari, Yuji Matsumoto et Rashmi Prasad, éditeurs, *COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan*, pages 641–651. ACL, 2016. ISBN 978-4-87974-702-0. URL http://aclweb.org/anthology/C/C16/C16-1062.pdf.

Alberto García-Durán, Antoine Bordes, Nicolas Usunier et Yves Grandvalet. Combining two and three-way embeddings models for link prediction in knowledge bases. *CoRR*, abs/1506.00999, 2015. URL http://arxiv.org/abs/1506.00999.

Kiril Gashteovski, Rainer Gemulla et Luciano Del Corro. MinIE: Minimizing Facts in Open Information Extraction. Dans *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2630–2640. Association for Computational Linguistics, 2017. URL http://aclweb.org/anthology/D17-1278.

Fabrizio Gotti et Philippe Langlais. Harnessing open information extraction for entity classification in a french corpus. Dans *Canadian AI 2016*. Springer International Publishing Switzerland, Springer International Publishing Switzerland, 2016.

Paul Groth, Mike Lauruhn, Antony Scerri et Ron Daniel. Open information extraction on scientific text: An evaluation. Dans *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3414–3423. Association for Computational Linguistics, 2018. URL `http://aclweb.org/anthology/C18-1289`.

Adam Grycner et Gerhard Weikum. Harpy: Hypernyms and alignment of relational paraphrases. Dans *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2195–2204, Dublin, Ireland, August 2014. Dublin City University and Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/C14-1207`.

Adam Grycner, Gerhard Weikum, Jay Pujara, James Foulds et Lise Getoor. RELLY: Inferring Hypernym Relationships Between Relational Phrases. Dans *Conference on Empirical Methods in Natural Language Processing*, 2015.

Xu Han, Shulin Cao, Lv Xin, Yankai Lin, Zhiyuan Liu, Maosong Sun et Juanzi Li. Openke: An open toolkit for knowledge embedding. Dans *Proceedings of EMNLP*, 2018.

Luheng He, Mike Lewis et Luke Zettlemoyer. Question-answer driven semantic role labeling: Using natural language to annotate natural language. Dans *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 643–653. Association for Computational Linguistics, 2015. URL `http://www.aclweb.org/anthology/D15-1076`.

Kenneth Heafield. Kenlm: Faster and smaller language model queries. Dans *Pro-*

ceedings of the Sixth Workshop on Statistical Machine Translation, Edinburgh, UK, July 2011. Association for Computational Linguistics.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer et Daniel S. Weld. Knowledge-based weak supervision for information extraction of overlapping relations. Dans *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 541–550, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-87-9. URL `http://dl.acm.org/citation.cfm?id=2002472.2002541`.

Heng Ji et Ralph Grishman. Knowledge base population: Successful approaches and challenges. Dans *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1148–1158, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics. ISBN 978-1-932432-87-9. URL `http://dl.acm.org/citation.cfm?id=2002472.2002618`.

Rudolf Kadlec, Ondrej Bajgar et Jan Kleindienst. Knowledge base completion: Baselines strike back. Dans *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 69–74. Association for Computational Linguistics, 2017. URL `http://aclweb.org/anthology/W17-2609`.

Paul Kingsbury et Martha Palmer. From treebank to propbank. Dans *Language Resources and Evaluation*, 2002.

Karin Kipper, Hoa Trang Dang et Martha Palmer. Class-based construction of a verb lexicon. Dans *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, pages 691–696. AAAI Press, 2000. ISBN 0-262-51112-6. URL `http://dl.acm.org/citation.cfm?id=647288.721573`.

William Léchelle, Fabrizio Gotti et Philippe Langlais. WiRe57 : A Fine-Grained

Benchmark for Open Information Extraction. *CoRR*, abs/1809.08962, 2018. URL http://arxiv.org/abs/1809.08962.

William Léchelle et Philippe Langlais. An Informativeness Approach to Open IE Evaluation. Dans Alexander Gelbukh, éditeur, *CICLing*, Lecture Notes in Computer Science. Springer, Springer, 2016.

William Léchelle et Phillippe Langlais. Revisiting the Task of Scoring Open IE Relations. Dans *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japon, May 2018. European Language Resources Association (ELRA).

Douglas B. Lenat. Cyc: A large-scale investment in knowledge infrastructure. *Commun. ACM*, 38(11):33–38, novembre 1995. ISSN 0001-0782. URL http://doi.acm.org/10.1145/219717.219745.

Xiang Li, Aynaz Taheri, Lifu Tu et Kevin Gimpel. Commonsense knowledge base completion. Dans *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*, pages 1445–1455. Association for Computational Linguistics, 2016. URL http://aclweb.org/anthology/P16-1137.

Mausam. Open information extraction systems and downstream applications. Dans *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, pages 4074–4077. AAAI Press, 2016. ISBN 978-1-57735-770-4. URL http://dl.acm.org/citation.cfm?id=3061053.3061220.

Mausam, Michael Schmitz, Robert Bart, Stephen Soderland et Oren Etzioni. Open language learning for information extraction. Dans *Proceedings of Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CONLL)*, 2012.

Yuval Merhav, Filipe Mesquita, Denilson Barbosa, Wai Gen Yee et Ophir Frieder. Extracting information networks from the blogosphere. *ACM Trans. Web*, 6

(3):11:1–11:33, octobre 2012. ISSN 1559-1131. URL http://doi.acm.org/10.1145/2344416.2344418.

Filipe Mesquita, Jordan Schmidek et Denilson Barbosa. Effectiveness and efficiency of open relation extraction. Dans *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 447–457. Association for Computational Linguistics, October 2013.

George A. Miller. Wordnet: A lexical database for english. *Commun. ACM*, 38 (11):39–41, novembre 1995. ISSN 0001-0782. URL http://doi.acm.org/10.1145/219717.219748.

Mike Mintz, Steven Bills, Rion Snow et Dan Jurafsky. Distant supervision for relation extraction without labeled data. Dans *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, pages 1003–1011, Stroudsburg, PA, USA, 2009. Association for Computational Linguistics. ISBN 978-1-932432-46-6. URL http://dl.acm.org/citation.cfm?id=1690219.1690287.

Bhavana Dalvi Mishra, Niket Tandon et Peter Clark. Domain-targeted, high precision knowledge extraction. *TACL*, 5:233–246, 2017. URL https://transacl.org/ojs/index.php/tacl/article/view/1064.

T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis, T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves et J. Welling. Never-ending learning. Dans *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)*, 2015.

T. Mitchell, W. Cohen, E. Hruschka, P. Talukdar, B. Yang, J. Betteridge, A. Carlson, B. Dalvi, M. Gardner, B. Kisiel, J. Krishnamurthy, N. Lao, K. Mazaitis,

T. Mohamed, N. Nakashole, E. Platanios, A. Ritter, M. Samadi, B. Settles, R. Wang, D. Wijaya, A. Gupta, X. Chen, A. Saparov, M. Greaves et J. Welling. Never-ending learning. *Commun. ACM*, 61(5):103–115, avril 2018. ISSN 0001-0782. URL http://doi.acm.org/10.1145/3191513.

Ndapandula Nakashole, Gerhard Weikum et Fabian Suchanek. Patty: A taxonomy of relational patterns with semantic types. Dans *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 1135–1145, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=2390948.2391076.

Maximilian Nickel, Kevin Murphy, Volker Tresp et Evgeniy Gabrilovich. A Review of Relational Machine Learning for Knowledge Graphs: From Multi-Relational Link Prediction to Automated Knowledge Graph Construction. *CoRR*, abs/1503.00759, 2015. URL http://arxiv.org/abs/1503.00759.

Christina Niklaus, Matthias Cetto, André Freitas et Siegfried Handschuh. A survey on open information extraction. Dans *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3866–3878. Association for Computational Linguistics, 2018. URL http://aclweb.org/anthology/C18-1326.

Joakim Nivre, Johan Hall et Jens Nilsson. Maltparser: A data-driven parser-generator for dependency parsing. Dans *In Proc. of LREC-2006*, pages 2216–2219, 2006.

Harinder Pal et Mausam. Demonyms and compound relational nouns in nominal open ie. Dans *Proceedings of the 5th Workshop on Automated Knowledge Base Construction*, pages 35–39. Association for Computational Linguistics, 2016. URL http://aclweb.org/anthology/W16-1307.

Sachin Pawar, Girish K. Palshikar et Pushpak Bhattacharyya. Relation extraction

: A survey. *CoRR*, abs/1712.05191, 2017. URL http://arxiv.org/abs/1712.05191.

Adam Pease, Ian Niles et John Li. The suggested upper merged ontology: A large ontology for the semantic web and its applications. Dans *In Working Notes of the AAAI-2002 Workshop on Ontologies and the Semantic Web*, 2002.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot et E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Hoifung Poon et Pedro Domingos. Unsupervised ontology induction from text. Dans *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 296–305. Association for Computational Linguistics, 2010.

Lin Qiu, Hao Zhou, Yanru Qu, Weinan Zhang, Suoheng Li, Shu Rong, Dongyu Ru, Lihua Qian, Kewei Tu et Yong Yu. QA4IE: A question answering based framework for information extraction. *CoRR*, abs/1804.03396, 2018. URL http://arxiv.org/abs/1804.03396.

Haniyeh Rashidghalam, Mina Taherkhani et Fariborz Mahmoudi. Text summarization using concept graph and babelnet knowledge base. pages 115–119, 04 2016.

Sebastian Riedel, Limin Yao, Benjamin M. Marlin et Andrew McCallum. Relation extraction with matrix factorization and universal schemas. Dans *Joint Human Language Technology Conference/Annual Meeting of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL '13)*, 2013. URL http://www.riedelcastro.org/publications/papers/riedel13relation.pdf.

Benjamin Roth, Tassilo Barth, Michael Wiegand, Mittul Singh et Dietrich Klakow. Effective slot filling based on shallow distant supervision methods. *CoRR*, abs/1401.1158, 2014. URL `http://arxiv.org/abs/1401.1158`.

Josef Ruppenhofer, Michael Ellsworth, Miriam R. L. Petruck, Christopher R. Johnson et Jan Scheffczyk. FrameNet II: Extended theory and practice. Rapport technique, ICSI, 2005. URL `http://framenet.icsi.berkeley.edu/book/book.pdf`.

Swarnadeep Saha et Mausam. Open information extraction from conjunctive sentences. Dans *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2288–2299. Association for Computational Linguistics, 2018. URL `http://aclweb.org/anthology/C18-1194`.

Swarnadeep Saha, Harinder Pal et Mausam. Bootstrapping for numerical open IE. Dans Regina Barzilay et Min-Yen Kan, éditeurs, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 317–323. Association for Computational Linguistics, 2017. ISBN 978-1-945626-76-0. URL `https://doi.org/10.18653/v1/P17-2050`.

Roser Saurí et James Pustejovsky. Are you sure that this happened ? assessing the factuality degree of events in text. *Comput. Linguist.*, 38(2):261–299, juin 2012. ISSN 0891-2017. URL `http://dx.doi.org/10.1162/COLI_a_00096`.

Carl F Schaefer, Kira Anthony, Shiva Krupa, Jeffrey Buchoff, Matthew Day, Timo Hannay et Kenneth H Buetow. Pid: the pathway interaction database. *Nucleic acids research*, 37(suppl_1):D674–D679, 2008.

Rudolf Schneider, Tom Oberhauser, Tobias Klatt, Felix A. Gers et Alexander Löser. Analysing errors of open information extraction systems. Dans *Proceedings of the First Workshop on Building Linguistically Generalizable NLP Systems*, pages

11–18. Association for Computational Linguistics, 2017. URL http://aclweb.org/anthology/W17-5402.

Satoshi Sekine. On-demand information extraction. Dans *Proceedings of the COLING/ACL on Main Conference Poster Sessions*, COLING-ACL '06, pages 731–738, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1273073.1273167.

Yusuke Shinyama et Satoshi Sekine. Preemptive information extraction using unrestricted relation discovery. Dans *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, HLT-NAACL '06, pages 304–311, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics. URL https://doi.org/10.3115/1220835.1220874.

Stephen Soderland, John Gilmer, Robert Bart, Oren Etzioni et Daniel S. Weld. Open information extraction to KBP relations in 3 hours. Dans *Proceedings of the Sixth Text Analysis Conference, TAC 2013, Gaithersburg, Maryland, USA, November 18-19, 2013*. NIST, 2013. URL http://www.nist.gov/tac/publications/2013/participant.papers/UWashington.TAC2013.proceedings.pdf.

Robyn Speer et Catherine Havasi. Representing general relational knowledge in conceptnet 5. Dans *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC-2012)*. European Language Resources Association (ELRA), 2012. URL http://www.lrec-conf.org/proceedings/lrec2012/pdf/1072_Paper.pdf.

Gabriel Stanovsky et Ido Dagan. Creating a large benchmark for open information extraction. Dans *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Austin, Texas, November 2016. Association for Computational Linguistics.

Gabriel Stanovsky, Ido Dagan et Mausam. Open ie as an intermediate structure for semantic tasks. Dans *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 303–308, Beijing, China, July 2015. Association for Computational Linguistics. URL `http://www.aclweb.org/anthology/P15-2050`.

Gabriel Stanovsky, Jessica Ficler, Ido Dagan et Yoav Goldberg. Getting more out of syntax with props. *CoRR*, abs/1603.01648, 2016. URL `http://arxiv.org/abs/1603.01648`.

Fabian M. Suchanek, Gjergji Kasneci et Gerhard Weikum. Yago: A core of semantic knowledge. Dans *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 697–706, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-654-7. URL `http://doi.acm.org/10.1145/1242572.1242667`.

Mihai Surdeanu. Overview of the tac2013 knowledge base population evaluation: English slot filling and temporal slot filling. Dans *Proceedings of the TAC-KBP 2013 Workshop*, 2013. URL `http://clulab.cs.arizona.edu/papers/kbp2013.pdf`.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati et Christopher D Manning. Multi-instance multi-label learning for relation extraction. Dans *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465. Association for Computational Linguistics, 2012.

Andrey Timofeyev et Ben Choi. Knowledge based automatic summarization. pages 350–356, 01 2017.

Kristina Toutanova et Danqi Chen. Observed versus latent features for knowledge base and text inference. Dans *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*, pages 57–66. Association

for Computational Linguistics, 2015. URL http://aclweb.org/anthology/W15-4007.

Kristina Toutanova, Danqi Chen, Patrick Pantel, Hoifung Poon, Pallavi Choudhury et Michael Gamon. Representing text for joint embedding of text and knowledge bases. Dans Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin et Yuval Marton, éditeurs, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 1499–1509. The Association for Computational Linguistics, 2015. ISBN 978-1-941643-32-7. URL http://aclweb.org/anthology/D/D15/D15-1174.pdf.

Kristina Toutanova, Victoria Lin, Wen-tau Yih, Hoifung Poon et Chris Quirk. Compositional learning of embeddings for relation paths in knowledge base and text. Dans *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016. ISBN 978-1-945626-00-5. URL http://aclweb.org/anthology/P/P16/P16-1136.pdf.

Théo Trouillon, Johannes Welbl, Sebastian Riedel, Éric Gaussier et Guillaume Bouchard. Complex embeddings for simple link prediction. *CoRR*, abs/1606.06357, 2016. URL http://arxiv.org/abs/1606.06357.

Paola Velardi, Stefano Faralli et Roberto Navigli. Ontolearn reloaded: A graph-based algorithm for taxonomy induction. *Computational Linguistics*, 39(3):665–707, 2013. URL http://dx.doi.org/10.1162/COLI_a_00146.

Quan Wang, Bin Wang et Li Guo. Knowledge base completion using embeddings and rules. Dans *IJCAI*, pages 1859–1866, 2015.

Robert West, Evgeniy Gabrilovich, Kevin Murphy, Shaohua Sun, Rahul Gupta et Dekang Lin. Knowledge base completion via search-based question answering.

Dans *WWW*, 2014. URL http://www.cs.ubc.ca/~murphyk/Papers/www14.pdf.

Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins et Benjamin Van Durme. Universal Decompositional Semantics on Universal Dependencies. Dans *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1713–1723, Austin, Texas, November 2016. Association for Computational Linguistics. URL https://aclweb.org/anthology/D16-1177.

John Wieting, Mohit Bansal, Kevin Gimpel et Karen Livescu. From paraphrase database to compositional paraphrase model and back. *TACL*, 3:345–358, 2015a. URL https://tacl2013.cs.columbia.edu/ojs/index.php/tacl/article/view/571.

John Wieting, Mohit Bansal, Kevin Gimpel et Karen Livescu. Towards universal paraphrastic sentence embeddings. *CoRR*, abs/1511.08198, 2015b. URL http://arxiv.org/abs/1511.08198.

William Léchelle, Jason Jiechen Wu et Phillippe Langlais. Link Prediction in Open Domain Knowledge Bases. (en préparation).

Fei Wu et Daniel S. Weld. Open information extraction using wikipedia. Dans *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 118–127, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1858681.1858694.

Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao et Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *CoRR*, abs/1412.6575, 2014. URL http://arxiv.org/abs/1412.6575.

Alexander Yates, Michael Cafarella, Michele Banko, Oren Etzioni, Matthew Broadhead et Stephen Soderland. Textrunner: Open information extraction

on the web. Dans *Proceedings of Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, NAACL-Demonstrations '07, pages 25–26, Stroudsburg, PA, USA, 2007. Association for Computational Linguistics. URL http://dl.acm.org/citation.cfm?id=1614164.1614177.

Sheng Zhang, Rachel Rudinger et Ben Van Durme. An Evaluation of PredPatt and Open IE via Stage 1 Semantic Role Labeling. Dans *Proceedings of the 12th International Conference on Computational Semantics (IWCS)*, Montpellier, France, September 2017.

Alisa Zhila et Alexander Gelbukh. Open information extraction for spanish language based on syntactic constraints. Dans *Proceedings of the ACL 2014 Student Research Workshop*, pages 78–85, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/P/P14/P14-3011.

Jun Zhu, Zaiqing Nie, Xiaojiang Liu, Bo Zhang et Ji-Rong Wen. Statsnowball: A statistical approach to extracting entity relationships. Dans *Proceedings of the 18th International Conference on World Wide Web*, WWW 09, pages 101–110, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-487-4. URL http://doi.acm.org/10.1145/1526709.1526724.