

Extension lexicale de définitions grâce à des corpus annotés en sens

Loïc Vial¹, Andon Tchechmedjiev¹, Didier Schwab¹

¹LIG-GETALP, Université Grenoble Alpes, France

E-mail: {loic.vial, andon.tchechmedjiev, didier.schwab}@imag.fr

<http://getalp.imag.fr/WSD>

RÉSUMÉ

Pour un certain nombre de tâches ou d'applications du TALN, il est nécessaire de déterminer la proximité sémantique entre des sens, des mots ou des segments textuels. Dans cet article, nous nous intéressons à une mesure basée sur des savoirs, la mesure de Lesk. La proximité sémantique de deux définitions est évaluée en comptant le nombre de mots communs dans les définitions correspondantes dans un dictionnaire. Dans cet article, nous étudions plus particulièrement l'extension de définitions grâce à des corpus annotés en sens. Il s'agit de prendre en compte les mots qui sont utilisés dans le voisinage d'un certain sens et d'étendre lexicalement la définition correspondante. Nous montrons une amélioration certaine des performances obtenues en désambiguïstation lexicale qui dépassent l'état de l'art.

ABSTRACT

Lexical Expansion of definitions based on sense-annotated corpus

For many natural language processing tasks and applications, it is necessary to determine the semantic relatedness between senses, words or text segments. In this article, we focus on a knowledge-based measure, the Lesk measure, which is certainly among the most commonly used. The similarity between two senses is computed as the number of overlapping words in the definitions of the senses from a dictionary. In this article, we study the expansion of definitions through the use of sense-annotated corpora. The idea is to take into account words that are most frequently used around a particular sense and to use the top of the frequency distribution to extend the corresponding definition. We show better performances on a Word Sense Disambiguation task surpassing state-of-the-art.

MOTS-CLÉS : Extension lexicale, mesure de Lesk, corpus annotés en sens, désambiguïstation lexicale.

KEYWORDS: Lexical Expansion, Lesk measure, sense-annotated corpus, Word Sense Disambiguation.

1 Introduction

Pour un certain nombre de tâches ou d'applications du Traitement Automatique des Langues, il est nécessaire de déterminer la proximité sémantique entre des sens, des mots ou des segments textuels. Dans cet article, nous nous intéressons à une mesure basée sur des savoirs. Pour un état de l'art plus complet, le lecteur pourra se référer à (Budanitsky & Hirst, 2006), (Pedersen *et al.*, 2005), (Cramer

et al., 2010) ou (Navigli, 2009).

La mesure de Lesk (connue également sous le nom de recouvrement de définitions) est certainement l'une des plus utilisée. La proximité sémantique de deux définitions est évaluée en comptant le nombre de mots communs (pris simplement comme les chaînes de caractères entre les espaces) dans les définitions correspondantes dans un dictionnaire. Plusieurs variantes de cette mesure existent comme, par exemple, l'utilisation des relations dans un réseau lexical, la lemmatisation des mots de la définition, l'utilisation d'un anti-dictionnaire pour filtrer certains mots, etc.

Dans cet article, nous étudions plus particulièrement l'extension de définitions grâce à plusieurs corpus annotés en sens. Il s'agit de prendre en compte les mots qui sont utilisés dans le voisinage d'un certain sens et d'étendre lexicalement la définition correspondante.

Nous présentons dans un premier temps les mesures de proximité sémantique et plus particulièrement celle de Lesk au centre de ces travaux. Nous abordons les corpus annotés en sens et expliquons comment nous nous en servons pour enrichir les définitions des sens. Enfin nous évaluons notre approche sur une tâche de désambiguïsation lexicale et montrons une améliorations nette des résultats obtenus.

2 Proximité sémantique

2.1 Généralités

Une mesure de proximité sémantique permet d'estimer à quel point deux sens de mots (ou mots, ou segments textuels, ou textes) sont proches sémantiquement l'un de l'autre. On s'attend, par exemple, à ce que *'docteur'* et *'hôpital'* soient évalués comme plus proches que *'chien'* et *'avion'*. Il existe des dizaines de mesures peut-être même des milliers si on considère leurs variantes. On peut distinguer plusieurs domaines d'arrivée :

- $[0, 1]$: ce sont des similarités pour lesquelles une valeur tendant vers 1 indique des sens proches alors qu'une valeur tendant vers 0 indique des sens éloignés. C'est le cas, par exemple des mesures vectorielles comme dans LSA (Deerwester et al., 1990) ou Word2Vec (Mikolov et al., 2013) ;
- $[0, \pi/2]$ ou $[0, 90]$: un angle mesuré en radians ou en degrés comme c'est le cas, par exemple, pour les vecteurs d'idées (Schwab, 2005). Un angle proche de 0° (0 radian) correspond alors à des sens voisins et un angle proche de 90° ($\pi/2$ radians) correspond à des sens très éloignés ;
- \mathbb{N}^+ , un nombre entier positif, comme c'est le cas pour les mesures comme celle de Lesk (Lesk, 1986).

2.2 Évaluation de la proximité sémantique

Il est communément admis qu'il existe trois manières d'évaluer des mesures de proximité sémantique (Budanitsky & Hirst, 2006) :

- d'un point de vue théorique, par l'étude de leurs propriétés mathématiques (similarité mathématique, distance . . .) ;
- par la comparaison avec le jugement humain sur des ensembles de couples de mots évalués ;
- par l'étude des performances obtenues par une application particulière grâce à ces mesures.

Dans cet article, nous utilisons cette dernière approche en évaluant nos mesures sur une tâche de désambiguïsation lexicale.

2.3 Mesures de Lesk

Dans cette section, nous décrivons la mesure de Lesk et quelques unes de ses variantes classiques.

2.3.1 Mesure de Lesk originale

Il y a 30 ans, Lesk (1986) a proposé, un algorithme très simple qui évalue la proximité sémantique entre deux sens (S_1, S_2) comme le nombre de mots communs dans les définitions correspondantes ($D(s_1), D(s_2)$) issues d'un dictionnaire :

$$sim_{lesk}(s_1, s_2) = | D(s_1) \cap D(s_2) |$$

2.3.2 Variantes de la mesure de Lesk

La mesure de Lesk est ainsi sensible à la présence ou l'absence des mots dans les définitions. En effet, la mesure tient seulement compte des correspondances exactes entre les formes de surface des mots des définitions. Si des mots manquent ou si des synonymes sont utilisés, tout ou partie des correspondances potentielles ne pourront être trouvées. Comme les définitions issues des dictionnaires sont souvent assez concises, il est plus difficile de distinguer des différences fines entre définitions. Les variantes de la mesure de Lesk les plus encourageantes sont ainsi basées sur l'enrichissement des définitions. Nous ne nous intéresserons donc pas ici, ni à d'autres manières de calculer le score, ni à la lemmatisation ou la racinisation des mots, ni à l'utilisation d'antidictionnaires.

L'extension la plus classiquement utilisée est le Lesk étendu (appelé également adapté) de Banerjee & Pedersen (2002). Cette mesure nécessite une ressource composée de définitions pour les sens de mots mais également de liens reliant sémantiquement ces sens. Il s'agit ainsi d'enrichir la définition initiale du sens par les mots des définitions des sens qui lui sont liés, soit :

$$Lesk_{etendu}(s_1, s_2) = \sum_{\forall (R_1, R_2) \in RELPAIRS^2} (| D(R_1(s_1)) \cap D(R_2(s_2)) |)$$

où RELPAIRS est l'ensemble des liens reliant sémantiquement les sens s_1 et s_2 . Cette extension a été utilisée dans de nombreux travaux ((Vasilescu *et al.*, 2004) ou (Schwab *et al.*, 2011)) ainsi que l'ensemble des autres articles cités dans cette section par exemple) et son efficacité n'est plus discutée, seuls les types des relations utilisées diffèrent.

Miller *et al.* (2012) utilisent (1) un Lesk étendu et (2) un thésaurus distributionnel pour étendre les définitions en y ajoutant les termes associés à chacun des mots de la définition. Les résultats sont les meilleurs pour les définitions les plus longues testées (extensions de 100 mots).

Baldwin *et al.* (2010) comparent plusieurs manières d'étendre les définitions sur un dictionnaire japonais dont les définitions sont annotées en sens. Ils enrichissent ainsi chaque définition (1) par les

définitions des sens des mots qui la composent et (2) comme dans Lesk étendu par les définitions des sens liés dans le réseau lexical (uniquement les relations synonymes, hyperonyme, hyponyme). Les résultats du Lesk utilisant les définitions les plus enrichies sont ici aussi les meilleures sur un corpus japonais annoté en sens.

Il semble ainsi que pour la mesure de Lesk plus grandes sont les définitions, meilleurs sont les résultats obtenus. Nous étudions dans cet article l'enrichissement des définitions de WordNet à l'aide de plusieurs corpus annotés en sens de cette même base lexicale.

3 Extension de définitions par corpus annotés en sens

Dans cet article, nous utilisons *Princeton WordNet* (Fellbaum, 1998), une base lexicale pour l'anglais dans laquelle les sens de mots (les *synsets*, des ensembles de synonymes) sont décrits par une définition et sont reliés entre eux par des relations (hyperonymie, hyponymie, antonymie, etc.). Chaque entrée lexicale de WordNet est ainsi liée à un ou plusieurs *synsets* qui correspondent à autant de sens pour ce mot. Par habitude, on nomme *cat#n#3*³ le troisième sens du nom *cat*.

3.1 Corpus annotés en sens pour l'anglais

Il existe plusieurs corpus pour l'anglais annotés en sens. Nous ne présentons ici que ceux qui sont annotés avec des sens du Princeton WordNet :

- La *Defense Science Organisation* (Ng & Lee, 1996) a produit un corpus non disponible librement. 192 800 mots ont été annotés avec des *synsets* du WordNet. L'annotation se concentre sur 121 noms (113 000 occurrences) et 70 verbes (79 800 occurrences) qui ont été choisis parmi les plus fréquents et les plus ambigus de l'anglais. Selon les auteurs, la couverture correspond à environ 20% des occurrences de noms et de verbes en anglais.
- Le *SemCor* (Miller *et al.*, 1993) est un sous-ensemble du Corpus de Brown (Francis & Kučera, 1964). Sur les 700 000 mots de ce dernier, environ 230 000 sont annotés avec des *synsets* du WordNet. L'annotation porte au total sur 352 textes. Pour 186 d'entre eux, 192 639 mots (soit l'ensemble des noms, verbes, adjectifs et adverbes) sont annotés. Sur les 166 autres, seulement 41 497 verbes sont annotés.
- Le *Groningen Meaning Bank* (Basile *et al.*, 2012) inclut des annotations en sens mais aussi les parties du discours, les entités nommées, les rôles thématiques et les sens. Il a été construit semi-automatiquement grâce à une combinaison d'outil de TALN, d'experts et de crowdsourcing (jeux avec un but). Sa dernière version¹ (2.2.0) est sortie le 4 juillet 2014 et inclut 10 000 documents et 1 354 149 mots dont 666 562 sont annotés en sens (soit 49,22% de couverture).
- Le corpus des définitions de WordNet² qui contient les définitions de WordNet annotées en sens. Il contient 1 504 077 mots dont 458 825 sont annotés (couverture de 30,5%).
- Les corpus des campagnes d'évaluation de désambiguïsation lexicale pour l'anglais *SemEval-SensEval*. Ces corpus sont composés de peu de textes et dépassent rarement les 5000 mots.

Dans cet article, nous étendons nos définitions grâce aux quatre premiers et comparons leurs résultats sur la tâche 7 de la campagne *SemEval 2007*.

1. <http://gmb.let.rug.nl/data.php>

2. <http://wordnet.princeton.edu/glosstag.shtml>

SC	DSO	WNGT	GMB	Nombre d'annotations	Nombre de mots uniques annotés	Nombre de sens uniques annotés	Nombre de sens/mot
X				234 136	21 153	35 399	1.67
	X			177 478	188	1 421	7.55
X	X			411 614	21 153	35 620	1.68
		X		444 781	38 752	65 687	1.69
X		X		678 917	46 233	82 538	1.78
	X	X		622 259	38 752	65 845	1.69
X	X	X		856 395	46 233	82 589	1.78
			X	665 391	13 192	15 395	1.16
X			X	899 527	26 322	42 134	1.6
	X		X	842 869	13 192	16 583	1.25
X	X		X	1 077 005	26 322	42 353	1.6
		X	X	1 110 172	42 336	72 453	1.71
X		X	X	1 344 308	49 088	87 189	1.77
	X	X	X	1 287 650	42 336	72 604	1.71
X	X	X	X	1 521 786	49 088	87 240	1.77

TABLE 1 – Statistiques sur les différentes combinaisons de corpus.

La table 1 récapitule les statistiques des différentes combinaisons de corpus.

3.2 Extension de définitions

Notre méthode consiste à étendre les définitions grâce aux plus fréquents voisins d'un sens dans un ou des corpus annotés en sens. Nous considérons ici comme voisin, un mot trouvé dans la même phrase que le sens considéré. Pour un sens de mot donné, elle nécessite ainsi trois étapes :

1. extraire les phrases où se trouve ce sens ;
2. calculer la fréquences d'apparition de chacun des voisins et les trier en fonction de cette fréquence ;
3. étendre la définition de ce sens avec les n voisins les plus fréquents.

Par exemple, considérons que nous voulons étendre la deuxième définition du nom *stone* (*stone#n#2*) de 10 mots. Nous extrayons dans un premier temps les phrase des corpus qui contiennent ce sens. Nous classons l'ensemble de ses voisins en fonction de leur fréquence d'apparition puis nous sélectionnons les 10 premiers que nous rajoutons à *stone#n#2* de WordNet.

3.3 Données produites

Nous avons produit le dictionnaire qui sera notre référence c'est-à-dire sans extension basée sur des corpus. Il s'agit des définitions enrichies des définitions liées dans WordNet (extension à la (Banerjee & Pedersen, 2003)). Nous utilisons également un anti-dictionnaire³ pour filtrer certains mots des définitions et nous appliquons une racinisations sur les mots restants grâce à *Snowball Stemmer*⁴.

3. La plus longue liste de <http://www.ranks.nl/stopwords>.

4. <http://snowball.tartarus.org>

Nous avons également produit les 90 dictionnaires correspondant aux combinaisons des quatre corpus annotés en sens utilisés (SemCor, corpus de la *Defence Science Organization*, corpus des définition de WordNet, corpus *Groningen Meaning Bank*) et des six longueurs d'extensions testées (de 0 à 300 mots par pas de 50).

Un dépôt disponible à l'adresse

<https://github.com/getalp/WSD-TALN2016-Vialetal> contient :

- les deux meilleurs dictionnaires ;
- les sorties des 2730 exécutions (30 exécutions pour chacune des 91 combinaisons) ;
- le code permettant de calculer des mesures de proximité sémantique et de réaliser la désambiguïsation lexicale de textes en anglais.

Cette mise à disposition permettra :

- l'utilisation des mesures de proximité sémantique par la communauté ;
- l'exploitation de ces dictionnaires enrichis pour d'autres applications ;
- la reproductibilité des résultats.

4 Évaluation des mesures basées sur les définitions étendues

Comme nous le disions dans la partie 2.2, il existe trois manières d'évaluer une mesure de proximité sémantique. Nous évaluons ici nos mesures grâce à la troisième méthode, une tâche de désambiguïsation lexicale.

4.1 Désambiguïsation lexicale

La tâche de désambiguïsation lexicale (*Word Sense Disambiguation*) consiste à trouver pour chaque mot d'un texte le sens le plus approprié parmi un inventaire pré-défini. Par exemple, dans la phrase "*Je vois la montagne à travers ma fenêtre.*", l'algorithme devrait choisir le sens de «fenêtre» qui correspond à la menuiserie plutôt que celui qui correspond à l'interface graphique. Pour évaluer la désambiguïsation, il existe des campagnes dédiées (senseval puis semeval) qui permettent de produire des corpus annotés avec lesquels il est possible de tester et de comparer l'efficacité des algorithmes. Nous utilisons ici le corpus annoté en sens de WordNet issu de la tâche 7 de SemEval 2007.

4.1.1 Le corpus de la tâche 7 de SemEval 2007

Le corpus a été créé pour la tâche *gros grain* de la campagne d'évaluation *Semeval 2007* (Navigli *et al.*, 2007). Les organisateurs fournissent un inventaire de sens plus grossiers que ceux de WordNet. En effet, pour chaque terme, les sens considérés comme proches (par exemple, "*neige/précipitation*" et "*neige/couverture*" ou "*porc/animal*" et "*porc/viande*") sont groupés. Le corpus est composé de 5 textes de genres divers (journalisme, critique littéraire, voyage, informatique, biographies) dont il faut annoter les 2269 mots. Le nombre moyen de sens par mot est de 6,19 ; ramené à 3,1 pour l'inventaire de sens grossiers. Les compétiteurs étaient libres de se servir de cet inventaire (sens grossiers connus *a priori*) ou non (sens grossiers connus *a posteriori*). Dans le premier cas, le nombre de choix à faire pour chaque mot est réduit et la tâche moins compliquée. Dans le second cas, les sens annotés sont jugés corrects s'ils sont dans le bon groupement, une sorte d'erreur acceptable. Notre objectif est

de tester un système en vue d'une utilisation dans un cadre applicatif réel or l'inventaire de sens grossiers n'est vérifié manuellement que pour les 2269 mots utilisés dans le corpus d'évaluation, nous ne l'utilisons donc pas. Dans les expériences présentées ici, nous nous situons ainsi dans un cas de sens connus *a posteriori*. Les résultats sont analysés par les formules classiques :

$$\text{Précision} = \frac{\text{sens correctement annotés}}{\text{sens annotés}} \quad \text{Rappel} = \frac{\text{sens correctement annotés}}{\text{sens à annoter}}$$

$$\text{mesure} - F1 = \frac{2 \times P \times R}{P + R}$$

4.1.2 Désambiguïsation lexicale basée sur des similarités

Nous voulons évaluer des mesures de proximité sémantique en comparant leurs performances sur une tâche de désambiguïsation lexicale. Les algorithmes utilisant de telles mesures sont appelés algorithmes basés sur des similarités. Ils font partie de la famille plus large des algorithmes basés sur les savoirs (dictionnaires, bases lexicales, encyclopédies, ...).

Algorithmes locaux et algorithmes globaux Les algorithmes basés sur des similarités sont composés d'un ou parfois plusieurs algorithmes locaux et d'un algorithme global. Les algorithmes locaux correspondent aux mesures de proximité sémantique telles que nous les avons présentées en section 2 et permettent d'estimer la proximité entre deux sens de mots du texte. L'algorithme global, lui, propage ces mesures locales aux niveaux supérieurs (syntagmes, phrases, paragraphes, texte selon l'algorithme choisi) afin de désambiguïser l'ensemble du texte. Parmi les algorithmes globaux on trouve, entre autres, des algorithmes génétiques (Gelbukh *et al.*, 2003), des recuits simulés (Cowie *et al.*, 1992), des algorithmes à colonies de fourmis (Schwab *et al.*, 2011) (Schwab *et al.*, 2012) ou encore, depuis peu, des algorithmes à colonies d'abeilles (Abualhaija & Zimmermann, 2016).

La qualité de la désambiguïsation est essentiellement déterminée par les algorithmes locaux. En effet, l'algorithme global permet de parcourir plus ou moins efficacement l'espace afin de maximiser le score global calculé grâce aux algorithmes locaux. Ainsi, une mesure de proximité sémantique parfaite parviendrait à refléter la proximité qu'il existe entre les sens possibles pour les mots du texte et aiderait au mieux l'algorithme global à trouver le ou les sens les plus appropriés pour chaque mot d'un texte. Dit autrement, en un temps infini, tous les algorithmes globaux trouveraient la ou les solutions qui maximisent la cohérence sémantique du texte selon le ou les algorithmes locaux utilisés et, c'est donc de la performance de ces derniers que vient principalement la qualité de la désambiguïsation lexicale.

Dans la suite de cet article, nous souhaitons comparer les performances des 91 mesures Lesk correspondant aux 91 dictionnaires présentés dans la section 3.3, ce sont autant d'algorithmes locaux. Chacun sera utilisé avec le même algorithme global, une variante de l'algorithme des coucous (*Cuckoo Search Algorithm*), une heuristique décrite par (Yang & Deb, 2009) qui utilise le principe du vol de Lévy pour parcourir efficacement l'espace de recherche.

Algorithme global : algorithme des coucous Dans cet algorithme, on représente une solution possible à notre problème comme un vecteur associant un sens à chacun des mots d'un texte. Ce

vecteur est considéré comme la position dans l'espace de recherche d'un nid de coucou.

On démarre avec un ensemble de n nids de coucou, puis, à chaque itération, on choisit aléatoirement un nid de coucou x et on lui fait effectuer un vol de Lévy, c'est-à-dire un déplacement aléatoire dans l'espace en suivant une distribution de Lévy. On choisit ensuite aléatoirement un deuxième nid y et on remplace sa solution si la nouvelle solution du nid x est meilleure que celle ci. Enfin, à chaque fin d'itération, on supprime un pourcentage p des nids de coucou ayant les plus mauvaises solutions pour les remplacer par des nouveaux nids créés aléatoirement.

La loi de Lévy a la caractéristique d'avoir une queue lourde, ce qui permet aux coucous d'explorer intensivement l'espace local dans lequel ils se situent tout en effectuant parfois des "sauts" plus loin dans l'espace pour éviter de tomber dans un extremum local.

L'algorithme originel nécessite ainsi 4 paramètres : le nombre de nids de coucou n , le pourcentage de mauvais nids de coucou à remplacer p , et les paramètres de position μ et d'échelle c de la distribution de Lévy utilisée.

Cependant, notre implémentation est une variante de l'algorithme original qui va décider automatiquement du nombre de coucous n et des paramètres μ et c de la distribution de Lévy en essayant de maximiser la fonction objectif (ie. maximiser les similarités entre les mots). Il n'y a donc aucun paramétrage particulier à réaliser et donc strictement aucun apprentissage n'est nécessaire.

Notons également que cet algorithme permet d'annoter l'ensemble du texte. Nous avons ainsi la précision, le rappel et le F-mesure qui sont égaux ⁵.

4.2 Évaluation

Nos données sont analysées selon deux axes : les corpus annotés en sens utilisés (SemCor noté également SC, corpus de la *Defence Science Organization* noté DSO, corpus des définition de WordNet noté WNGT, corpus *Groningen Meaning Bank* noté GMB) et la longueur de l'extension ⁶ (de 0 à 300 mots par pas de 50).

La recherche par coucous est probabiliste et ses résultats diffèrent (de l'ordre de 1%) entre deux exécutions. Il est ainsi nécessaire d'étudier la distribution des résultats afin de caractériser correctement les performances de l'algorithme. Pour chaque combinaison, nous effectuons 30 exécutions. Grâce à un test de Shapiro-Wilk, nous savons que les résultats ne suivent pas une loi de distribution normale. Ainsi, nous utilisons un test de Wilcoxon-Mann-Whitney (Wilcoxon, 1945; Mann & Whitney, 1947) pour vérifier la significativité ($p < 0.01$) paire à paire de chaque résultat. Au vu des conditions d'expérience, la variable U est comprise entre 450 et 900, une valeur de 450 signifie que l'on ne peut pas départager les résultats tandis qu'une valeur proche de 900 indiquera une très forte séparation des résultats. La table 2 présente les moyennes obtenues pour chaque combinaison de corpus et chaque extension. La figure 1 présente visuellement ces mêmes moyennes.

Constatons tout d'abord que quel que soit le corpus utilisé pour étendre les définitions, les résultats obtenus sont bien meilleurs que sans extension (au minimum, 1,54% de gain). En revanche, on ne peut pas dire que plus l'extension est importante, meilleurs sont les résultats.

5. Rappelons que si 100% du corpus est annoté alors $P=R=F$ puisque les sens annotés sont égaux aux sens à annoter ($P=R$) et dans ce cas $F = \frac{2 \times P \times P}{P+P} = \frac{2 \times P^2}{2P} = P$

6. Dans cette section, lorsque nous employons le mot extension, nous parlons d'extension à l'aide de corpus, l'ensemble des dictionnaires bénéficiant des mêmes extensions à la (Banerjee & Pedersen, 2003) par ailleurs (voir 3.3).

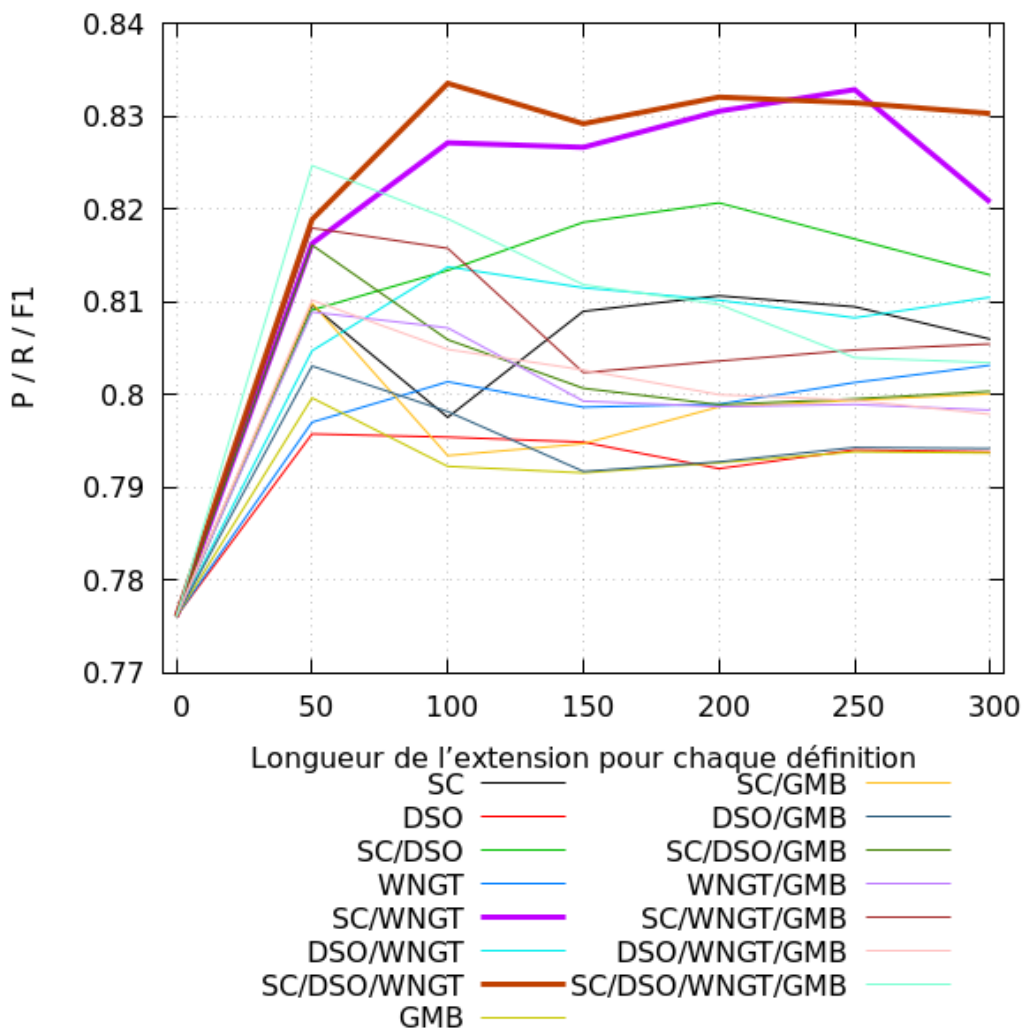


FIGURE 1 – Graphique des moyennes des P/R/F1 pour chacune des extensions réalisées. SC = SemCor, DSO = corpus de la *Defence Science Organization*, WNGT = corpus des définition de WordNet, GMB = corpus *Groningen Meaning Bank*.

SC	DSO	WNGT	GMB	0	50	100	150	200	250	300
X				77.61%	80.96%	79.75%	80.89%	81.06%	80.94%	80.59%
	X			77.61%	79.57%	79.54%	79.48%	79.20%	79.40%	79.38%
X	X			77.61%	80.91%	81.33%	81.85%	82.06%	81.67%	81.28%
		X		77.61%	79.70%	80.13%	79.86%	79.89%	80.12%	80.31%
X		X		77.61%	81.62%	82.71%	82.66%	83.05%	83.28%	82.07%
	X	X		77.61%	80.47%	81.37%	81.15%	81.01%	80.82%	81.04%
X	X	X		77.61%	81.88%	83.35%	82.91%	83.20%	83.14%	83.02%
			X	77.61%	79.96%	79.22%	79.15%	79.26%	79.38%	79.37%
X			X	77.61%	80.98%	79.34%	79.47%	79.86%	79.93%	80.00%
	X		X	77.61%	80.30%	79.81%	79.17%	79.27%	79.42%	79.41%
X	X		X	77.61%	81.61%	80.59%	80.06%	79.89%	79.95%	80.03%
		X	X	77.61%	80.88%	80.71%	79.92%	79.86%	79.89%	79.83%
X		X	X	77.61%	81.79%	81.57%	80.23%	80.36%	80.48%	80.54%
	X	X	X	77.61%	81.01%	80.48%	80.26%	79.99%	79.93%	79.78%
X	X	X	X	77.61%	82.46%	81.89%	81.18%	80.97%	80.39%	80.34%

TABLE 2 – Moyenne des scores F1 pour chacune des extensions réalisées. Les scores en gras sont les meilleurs pour chaque combinaison. Les deux scores en rouge sont significativement meilleurs que les autres. En revanche la différence entre les deux n’est pas significative ($p = 0,64$, $U = 481$). SC = SemCor, DSO = corpus de la *Defence Science Organization*, WNGT = corpus des définition de WordNet, GMB = corpus *Groning Meaning Bank*.

On distingue clairement deux combinaisons meilleures que les autres : SemCor/DSO/WNGT avec une extension de 100 et SC/WNGT avec une extension de 250. La première améliore la moyenne de 5,74% et la seconde de 5,67% les résultats obtenus avec définitions sans extensions. Les résultats des algorithmes qui les utilisent sont significativement meilleurs que les autres ($p < 0,01$). En revanche la différence entre les deux n’est pas significative ($p = 0,64$, $U = 481$).

Il est également possible d’utiliser ces résultats pour comparer individuellement la qualité des corpus. Utilisé seul, le SemCor obtient toujours des meilleurs résultats que les 3 autres corpus seuls. D’ailleurs, les 4 combinaisons qui arrivent en tête ont été créées grâce au SemCor.

Le DSO seul obtient les plus mauvais résultats, mais on peut l’expliquer du fait que c’est le corpus qui contient le moins de mots différents annotés. Sa qualité est pourtant réelle puisque si on l’ajoute au couple SC/WNGT, on obtient l’un des deux meilleurs résultats pour une extension de 150 mots.

Le WNGT et le GMB seuls sont, eux, à peu près équivalents. Rappelons toutefois que le WNGT associé au SC obtient le meilleur résultat pour l’extension de 250 mots, et qu’associé au DSO il obtient en moyenne le 5ème meilleur résultat.

4.3 Comparaison à d’autres systèmes

Les résultats obtenus avec ces deux systèmes (83,35% et 83,28%) les classeraient en première et deuxième position des systèmes évalués sur le corpus de SemEval 2007 portés à notre connaissance devant le système arrivé en tête de la compétition avec 82,5%. NUS-PT (Chan *et al.*, 2007) est un système qui utilise des corpus annotés similaires à ceux de nos systèmes (SemCor, DSO et Extended WordNet une autre annotation en sens des définitions de WordNet). Trois différences avec notre approche : l’utilisation de ressources supplémentaires (corpus parallèles anglais-chinois) et surtout

connaissance des sens *a priori* qui est une tâche plus simple mais plus artificielle⁷ que l'approche *a posteriori* que nous suivons dans ces travaux. Enfin une dernière différence qui, elle, sort du thème principal de cet article, ce système est de type supervisé ce qui implique un coût de calcul pour la désambiguïsation bien supérieur à celui nécessaire pour une désambiguïsation non-supervisée telle que la réalisent les systèmes que nous présentons ici.

Dans l'état de l'art, le système présentant les meilleurs résultats sur ce corpus d'évaluation est S2C de Chen *et al.* (2014). Il s'appuie principalement sur une représentation vectorielle des mots et des sens. Dans un premier temps un modèle type Word2vec (Mikolov *et al.*, 2013) est entraîné à partir du Wikipedia anglais, puis un vecteur est créé pour chaque sens de WordNet à partir de la moyenne des vecteurs des mots présents dans leur définition. Le système va ensuite trouver un sens pour chacun des mots d'un document en prenant le sens pour lequel la similarité entre son vecteur et les vecteurs des mots de la même phrase est maximisée. Si cette similarité ne dépasse pas un certain seuil de confiance, la stratégie de repli est de choisir le sens le plus fréquent. S2C obtient 82,6% avec son meilleur jeu de paramètres.

Pour être complets, comme nous le notions dans la section 4.1.2, l'algorithme des coucous est stochastique et ses résultats diffèrent de l'ordre de 1% à chaque exécutions. Ainsi, environ 84,3% des exécutions utilisant le SC/WNGT-250 dépassent les scores de ces deux systèmes tandis que 100% des exécutions utilisant le SemCor/DSO/WNGT-100 les dépassent.

5 Conclusion

Dans cet article, nous avons présenté une technique d'amélioration de la mesure de similarité de Lesk par l'extension des définitions à l'aide de corpus annotés en sens. Notre technique consiste à ajouter à la définition d'un sens les mots qui se trouvent le plus souvent dans les mêmes phrases que lui.

Nous avons montré qu'une telle extension améliore significativement les résultats obtenus sur une application : la désambiguïsation lexicale. Les meilleures extensions permettent même d'atteindre une performance comparable à celle des meilleurs systèmes supervisés, tout en conservant une simplicité de calcul et une rapidité d'exécution propre aux systèmes non-supervisés.

Nous avons détaillé la construction de nos dictionnaires, qui sont disponibles pour la communauté et qui contient toutes les définitions de WordNet étendues. Cette ressource peut être utilisée par exemple pour calculer la similarité entre des sens, ce qui peut aider dans diverses applications comme la recherche d'information, le résumé automatique de texte ou l'accès lexical.

D'autres pistes d'améliorations peuvent être envisagées. D'autres extensions lexicales sont possibles comme, par exemple, en utilisant des corpus non-annotés par la méthode de (Miller *et al.*, 2012). L'analyse des performances peut aussi être analysé en suivant la méthode de (Schwab *et al.*, 2015). Aussi, la participation de certains corpus à notre extension donne parfois de meilleurs résultats tandis que l'ajout d'autres à tendance à les dégrader. Il serait intéressant dans un futur travail d'en observer la raison plus en détails.

7. Voir section 4.1.1

Références

- ABUALHAIJA S. & ZIMMERMANN K.-H. (2016). D-bees : A novel method inspired by bee colony optimization for solving word sense disambiguation. *Swarm and Evolutionary Computation*, p.-.
- BALDWIN T., KIM S., BOND F., FUJITA S., MARTINEZ D. & TANAKA T. (2010). A reexamination of mrd-based word sense disambiguation. **9**(1), 4 :1–4 :21.
- BANERJEE S. & PEDERSEN T. (2002). An adapted lesk algorithm for word sense disambiguation using wordnet. In *CICLing 2002*, Mexico City.
- BANERJEE S. & PEDERSEN T. (2003). Extended gloss overlaps as a measure of semantic relatedness. In *In Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence*, p. 805–810.
- BASILE V., BOS J., EVANG K. & VENHUIZEN N. (2012). Developing a large semantically annotated corpus. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC 2012)*, p. 3196–3200, Istanbul, Turkey.
- BUDANITSKY A. & HIRST G. (2006). Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, **32**(1), 13–47.
- CHAN Y. S., NG H. T. & ZHONG Z. (2007). Nus-pt : Exploiting parallel texts for word sense disambiguation in the english all-words tasks. In *Proceedings of the 4th International Workshop on Semantic Evaluations, SemEval '07*, p. 253–256, Stroudsburg, PA, USA : Association for Computational Linguistics.
- CHEN X., LIU Z. & SUN M. (2014). A unified model for word sense representation and disambiguation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 1025–1035, Doha, Qatar : Association for Computational Linguistics.
- COWIE J., GUTHRIE J. & GUTHRIE L. (1992). Lexical disambiguation using simulated annealing. In *COLING 1992*, volume 1, p. 359–365, Nantes, France.
- CRAMER I., WANDMACHER T. & WALTINGER U. (2010). *WordNet : An electronic lexical database*, chapter Modeling, Learning and Processing of Text Technological Data Structures. Springer.
- DEERWESTER S. C., DUMAIS S. T., LANDAUER T. K., FURNAS G. W. & HARSHMAN R. A. (1990). Indexing by latent semantic analysis. *Journal of the American Society of Information Science*, **41**(6).
- FELLBAUM C. (1998). *WordNet : An Electronic Lexical Database (Language, Speech, and Communication)*. The MIT Press.
- FRANCIS W. N. & KUČERA H. (1964). *A Standard Corpus of Present-Day Edited American English, for use with Digital Computers (Brown)*. Rapport interne, Brown University, Providence, Rhode Island.
- GELBUKH A., SIDOROV G. & HAN S. Y. (2003). Evolutionary approach to natural language wsd through global coherence optimization. *WSEAS Transactions on Communications*, **2**(1), 11–19.
- LESK M. (1986). Automatic sense disambiguation using mrd : how to tell a pine cone from an ice cream cone. In *Proceedings of SIGDOC '86*, p. 24–26, New York, NY, USA : ACM.
- MANN H. B. & WHITNEY D. R. (1947). On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics*, **18**(1), 50–60.
- MIKOLOV T., SUTSKEVER I., CHEN K., CORRADO G. S. & DEAN J. (2013). Distributed representations of words and phrases and their compositionality. In C. BURGESS, L. BOTTOU,

- M. WELLING, Z. GHAHRAMANI & K. WEINBERGER, Eds., *Advances in Neural Information Processing Systems 26*, p. 3111–3119. Curran Associates, Inc.
- MILLER G. A., LEACOCK C., TENGI R. & BUNKER R. T. (1993). A semantic concordance. In *Proceedings of the workshop on Human Language Technology, HLT '93*, p. 303–308, Stroudsburg, PA, USA : Association for Computational Linguistics.
- MILLER T., BIEMANN C., ZESCH T. & GUREVYCH I. (2012). Using distributional similarity for lexical expansion in knowledge-based word sense disambiguation. In *Proceedings of COLING 2012*, p. 1781–1796, Mumbai, India : The COLING 2012 Organizing Committee.
- NAVIGLI R. (2009). Wsd : a survey. *ACM Computing Surveys*, **41**(2), 1–69.
- NAVIGLI R., LITKOWSKI K. C. & HARGRAVES O. (2007). Semeval-2007 task 07 : Coarse-grained english all-words task. In *SemEval-2007*, p. 30–35, Prague, Czech Republic.
- NG H. T. & LEE H. B. (1996). Integrating multiple knowledge sources to disambiguate word sense : an exemplar-based approach. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics, ACL '96*, p. 40–47, Stroudsburg, PA, USA : Association for Computational Linguistics.
- PEDERSEN T., BANERJEE S. & PATWARDHAN S. (2005). *Maximizing Semantic Relatedness to Perform WSD*. Research report, University of Minnesota Supercomputing Institute.
- SCHWAB D. (2005). *Approche hybride pour la modélisation, la détection et l'exploitation des fonctions lexicales en vue de l'analyse sémantique de texte*. PhD thesis, Université Montpellier 2.
- SCHWAB D., GOULIAN J. & GUILLAUME N. (2011). Désambiguïstation lexicale par propagation de mesures sémantiques locales par algorithmes à colonies de fourmis. In *Traitement Automatique des Langues Naturelles (TALN)*, Montpellier, France.
- SCHWAB D., GOULIAN J., TCHECHMEDJIEV A. & BLANCHON H. (2012). Ant Colony Algorithm for the Unsupervised Word Sense Disambiguation of Texts : Comparison and Evaluation. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2012)*, Mumbai (India).
- SCHWAB D., TCHECHMEDJIEV A., GOULIAN J. & SÉRASSET G. (2015). *Language Production, Cognition, and the Lexicon*, chapter Comparisons of Relatedness Measures Through a Word Sense Disambiguation Task, p. 221–243. Springer International Publishing : Cham.
- VASILESCU F., LANGLAIS P. & LAPALME G. (2004). Désambiguïstation de corpus monolingues par des approches de type lesk. In *Actes de TALN 2004, Traitement Automatique des Langues Naturelles*, Fès, Maroc.
- WILCOXON F. (1945). Individual Comparisons by Ranking Methods. *Biometrics Bulletin*, **1**(6), 80–83.
- YANG X.-S. & DEB S. (2009). Cuckoo search via lévy flights. *Proc. of World Congress on Nature and Biologically Inspired Computing*, p. 210–214.