Recherche de « périsegments » dans un contexte d'analyse conceptuelle assistée par ordinateur : le concept d'« esprit » chez Peirce

Davide Pulizzotto[1,2] José A. Lopez[1,3] Jean-François Chartier[1] Jean-Guy Meunier[1] Louis Chartrand[1] Francis Lareau[1] Tan Le Ngoc[1]
(1) LANCI-UQAM, 405 Rue Sainte-Catherine Est, H2L 2C4 Montréal, Canada
(2) Doctorat sémiologie, UQAM, 405 Rue Sainte-Catherine Est, H2L 2C4 Montréal, Canada
(3) Département de mathématiques, UQTR, 3351 Boulevard des Forges, G9A 5H7 Trois-Rivières, Canada
davide.pulizzotto@gmail.com, josalelg@hotmail.com

RÉSUMÉ
En sciences humaines et plus particulièrement en philosophie, l'analyse conceptuelle (AC) est une pratique fondamentale qui permet de décortiquer les propriétés d'un concept. Lors de l'analyse d'un un corpus textuel, le principal défi est l'identification des segments de texte qui expriment le concept. Parfois, ces segments sont facilement reconnaissables grâce à une unité lexicale attendue, appelée forme canonique. Toutefois, ce n'est pas toujours le cas. Cet article propose une chaîne de traitement pour la découverte d'un certain nombre de segments périphériques, dits périsegments. Pour illustrer le processus, nous réalisons des expérimentations sur le concept d'« esprit » dans les Collected Papers de Ch. S. Peirce, en obtenant une précision moyenne supérieure à 83%.

ABSTRACT
Search of "perisegments" in computer-assisted conceptual analysis : the concept of "mind" in Peirce.

In humanities and especially in philosophy, Conceptual Analysis (AC) is a fundamental practice for studying concept's proprieties. When this analysis is executed on a textual corpus, the challenge is to identify segments of text expressing the concept. Sometimes, these are easily retrievable in reason of one expected lexical item, that we call canonical form. However, this is not always the case. This article outlines a processing chain for peripheral segments, or perisegments, discovery. This process is illustrated by some experiments on the concept of "mind" in the Collected Papers of Ch. S. Peirce, obtaining more than 83% of the average precision.

MOTS-CLÉS : philosophie, Peirce, esprit, analyse conceptuelle, assistée par ordinateur, concordance, périsegment, composantes sémantiques, k-moyennes, apprentissage automatique, SVM.

KEYWORDS: philosophy, Perice, mind, conceptual analysis, computer-assisted, concordance, KWIC, perisegment, semantic component, k-means, machine learning, SVM.

1 Introduction

Dans les humanités et plus particulièrement en philosophie, une méthode d'analyse conceptuelle (AC) consiste généralement à parcourir un corpus de textes d'un ou plusieurs auteurs afin de découvrir

les diverses dimensions ou propriétés d'un processus mental complexe appelé le concept. Étudier le concept d'« évolution » et ses propriétés dans l'ouvrage de Darwin, ou celui de « beauté » dans les romans français de la moitié du XIX siècle, sont des exemples d'AC. Il demeure toutefois difficile d'établir une méthode d'AC reconnue et largement utilisée en sciences humaines, surtout à cause des multiples et divergentes théories du *concept* (Laurence & Margolis, 2003). Par conséquent, il existe plusieurs méthodes et approches pour conduire une AC (Beaney, 2015).Toutefois, si on adopte une approche linguistique et sémantique et l'hypothèse que le concept se manifeste surtout à travers des expressions linguistiques, nous pouvons affirmer que, généralement, l'AC identifie les dimensions d'un concept à partir d'une ou plusieurs *formes canoniques* et d'une analyse des segments de texte où elles sont présentes. Dans sa forme la plus simple, la forme canonique coïncide avec le mot qui véhicule le mieux le concept à l'étude. Par exemple, le mot « évolution » constitue la forme canonique du concept d'« évolution » chez Darwin. Cependant, l'ensemble des segments qui contiennent la forme canonique d'un concept peut ne pas être exhaustif et ne pas traduire toute la complexité sémantique que possède chacune des dimensions d'un concept. Vraisemblablement, ces dernières peuvent s'exprimer aussi au moyen de segments périphériques qui ne contiennent pas le mot « évolution ». En effet, dans son ouvrage le plus connu, Darwin n'utilise le mot « évolution » que huit fois (Sainte-Marie *et al.*, 2010). Pourtant, il est reconnu que sa théorie est centrée sur le concept d'« évolution ». Ceci représente une des difficultés les plus importantes pour l'analyse conceptuelle en sciences humaines, notamment en philosophie.

Cet article propose une chaîne de traitement qui assiste informatiquement une analyse conceptuelle traditionnelle et où se pose principalement le problème de la sélection d'un certain nombre de segments périphériques. Nous appellerons «périsegment» ce dernier type de segment, «concordance» l'ensemble des segments contenant la forme canonique et «composantes sémantiques» les dimensions ou propriétés du concept. La méthode se fonde sur l'hypothèse suivante : les périsegments peuvent être trouvés à partir de la *signature* des composantes du concept, lesquelles sont déterminées à partir de la concordance. Dans cette recherche, nous menons une expérimentation sur le concept d'« esprit » dans les *Collected Papers* de Ch. S. Peirce et nous montrons qu'un certain nombre de périsegments pertinents à des fins d'analyse conceptuelle peuvent être extraits à l'aide de l'ordinateur. La principale contribution de cet article réside dans le transfert des connaissances du traitement automatique des langues (TAL), de la recherche d'information (RI) et de l'apprentissage automatique (AA) vers le domaine des humanités numériques (HN) et plus particulièrement, vers la philosophie. L'article est divisé en trois parties : une première qui répertorie les travaux les plus directement liés à notre sujet, une deuxième qui détaille la méthode et les étapes de notre expérimentation et une partie finale qui résume les résultats et discute de l'importance d'une telle chaîne de traitement pour la philosophie et les HN.

2 Travaux reliés

Depuis plusieurs années, on trouve des recherches en philosophie qui utilisent des techniques développées en intelligence artificielle et intégrées dans une méthode générale d'analyse de texte (Bynum & Moor, 1998).Certains chercheurs ont utilisé les modèles topiques pour extraire les arguments dans des textes philosophiques du 19e siècle (Lawrence *et al.*, 2014) ; d'autres ont utilisé des outils de l'AA et du TAL pour répondre à des questions philosophiques, comme l'expression des relations causales (Girju & Moldovan, 2002) ou la perception subjective du temps (Schwartz *et al.*, 2015). Une autre catégorie de travaux utilise la RI pour la création d'une « ontologie dynamique » pour la

philosophie (Buckner *et al.*, 2007). Il existe aussi des projets sur la suggestion de documents (SalVe2) ou sur le traçage des citations (PhiloZoo) qui s'adressent explicitement aux philosophes.

En se servant des certaines techniques standard, notre recherche s'inscrit dans le cadre d'une analyse conceptuelle assistée par ordinateur en philosophie. Dans ce contexte, des outils (Forest & Meunier, 2005) ou des chaînes de traitement pour l'analyse d'un concept ont été développés. Le premier à utiliser ce genre de méthodes est probablement McKinnon, qui a étudié le concept de « destin » chez Kierkegard (McKinnon, 1973). Plus récemment, d'autres recherches ont exploré le concept de « langage » dans l'œuvre de Bergson (Danis, 2012; Estève, 2008), celui d'« évolution » dans l'œuvre de Darwin (Sainte-Marie *et al.*, 2010), le concept d'« esprit » chez Peirce (Meunier & Forest, 2009) et celui de « management » chez le philosophe Matsushita (Ding, 2013).

La majorité de ces travaux ont utilisé la concordance (Pincemin, 2007) pour l'analyse conceptuelle. À notre connaissance, il n'existe aucun travail traitant de la question de la découverte des périsegments, limitant ainsi l'analyse conceptuelle à l'étude de la forme canonique (concordance).

3 Méthode et expérimentation

De manière générale, notre chaîne de traitement se sépare en trois grandes étapes : 1) construction d'une *concordance* à partir de la forme canonique du concept étudié, 2) identification des *composantes sémantiques* présentes dans la concordance et 3) recherche des *périsegments*. Dans la première, nous choisissons une *forme canonique* du concept à l'étude et nous divisons le corpus en deux sous-corpus, la concordance et l'ensemble des périsegments potentiels. Ensuite nous générons une représentation vectorielle des deux sous-corpus et nous appliquons un algorithme de clustering sur la concordance pour l'identification des composantes sémantiques. La troisième étape a pour objectif la sélection d'un certain nombre de périsegments qui alimenteront l'analyse conceptuelle. Pour ce faire, nous extrayons la *signature* de chaque composante sémantique au moyen d'un processus d'apprentissage supervisé. Nous réduisons ensuite la dimensionnalité de l'espace sémantique généré et, enfin, nous calculons la similarité entre les signatures et tous les périsegments potentiels.

3.1 Corpus, segmentation et concordance

Les *Collected Papers* est la plus large collection d'écrits de Ch. S. Peirce, contenant huit volumes édités par *Havard University Press* entre 1931 et 1956. Étant un père fondateur du pragmatisme américain, Peirce demeure encore aujourd'hui une référence incontournable en philosophie. En nous inspirant de la manière classique de débuter une AC en philosophie, nous identifions dans la chaîne de caractères « mind » (« esprit » en anglais) la *forme canonique* la plus simple du concept que nous voulons étudier.

Pour procéder à une analyse conceptuelle assistée par ordinateur, une étape préalable est nécessaire ; elle consiste à segmenter le corpus en unités textuelles. La version du corpus que nous utilisons (Peirce, 1994) offre déjà une segmentation en paragraphes. Toutefois, nous préférons travailler avec une nouvelle segmentation parce que la longueur de ces paragraphes est irrégulière et, souvent, trop grande. Ils auraient donc alourdi le processus final d'évaluation et, surtout, ils auraient contenu trop d'information non pertinente pour l'analyse conceptuelle. En identifiant chaque phrase contenant la chaîne de caractères « mind », nous formons des segments de trois phrases en retenant la phrase

précédente et celle qui suit. Nous regroupons ces segments de trois phrases dans une concordance, qui forme le sous-corpus *C*, composé de 1 323 segments *c*. Le sous-ensemble du corpus qui ne fait pas partie de *C* est nommé *P*.Il est segmenté aussi en segments d'au maximum trois phrases (des segments d'une ou deux phrases peuvent apparaître) et il représente l'ensemble des périsegments potentiels, contenant ainsi 14 963 segments *p*. Enfin, le corpus complet *D* contient 16 286 segment *d*.

3.2 Identification de composantes sémantiques : modèle vectoriel et clustering

L'identification des composantes sémantiques correspond à l'obtention de plusieurs sous-ensembles de segments *c* en fonction de leur similarité sémantique. Cette opération est comparable à la découverte des différents usages de la forme canonique du concept étudié. Pour ce faire, dans un contexte d'assistance informatique, il est nécessaire de générer une représentation vectorielle du corpus *D*. Nous effectuons le traitement linguistique de base (tokenisation des segments, suppression des mots vides et lemmatisation) et nous construisons l'espace vectoriel, dans lequel chaque segment *d* est représenté par un vecteur unique qui rend compte de la fréquence de ses mots. Puisque nous voulons faire ressortir les mots les plus discriminants, la pondération TF-IDF est utilisée (Robertson, 2004). Nous obtenons ainsi une matrice segments-mots, composée de 13 903 colonnes qui correspondent aux mots et de 16 286 lignes qui correspondent aux segments *d*.

L'identification des composantes sémantiques du concept est réalisé par une opération d'apprentissage non supervisé. Celle-ci produit essentiellement des *clusters* de segments *c*, regroupés en fonction de leur similarité sémantique. Nous avons opté pour un algorithme de type *hard clustering* afin d'obtenir un partitionnement net des segments et d'éviter ainsi des chevauchements entre clusters. Parmi ce type de méthodes, nous avons retenu l'algorithme des k-moyennes (Pedregosa *et al.*, 2011) parce qu'il est reconnu pour ses bonnes performances sur des données textuelles (Steinbach *et al.*, 2000) et pour sa large utilisation (Jain, 2010).

Pour initialiser l'algorithme, nous devons établir la valeur de la variable *k* qui détermine le nombre de clusters à produire. Afin d'éviter des biais dans les résultats, nous avons calculé l'indice silhouette (Rousseeuw, 1987) pour chaque partitionnement possible de 2 à 50 *k* (figure 1). L'indice évalue la *cohésion* interne d'un cluster et sa *séparabilité* par rapport aux autres. Plus les clusters sont détectables, meilleure est la partition. Ceci nous a permis d'établir les trois meilleurs partitionnements, parmi lesquelles nous choisissons, par souci de simplicité, le moins complexe, ce qui correspond à $k = 12$.

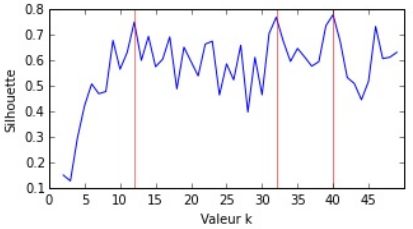

FIGURE 1 – Indice silhouette de 2 à 50 *k*

À la fin du processus d'apprentissage non supervisé, chaque cluster regroupe les segments *c* qui partagent des informations sémantiques similaires et qui contiennent la forme canonique. Ces sous-ensembles sont appelés les composantes sémantiques du concept d'« esprit » chez Peirce parce qu'ils contiennent des informations spécifiques pour chaque différent usage de la forme canonique.. Le cluster n° 1, par exemple, réunit les segments qui traitent de l'« association d'idées » (figure 2a), le cluster n° 2 ceux de « signe en relation à l'esprit », le cluster n° 3 ceux de « matière », le n° 4 ceux de « croyances », etc. Deux des 12 clusters ainsi générés ne correspondent pas à une véritable composante sémantique et n'ont donc pas été retenus dans l'étape finale de recherche et d'évaluation des périsegments.

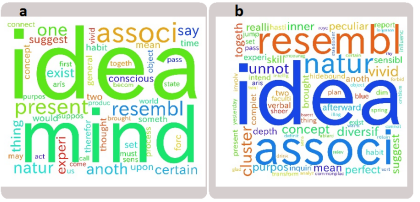

FIGURE 2 – Représentation graphique de mots pour le cluster n° 1, selon la fréquence TF/IDF (a) et les poids du vecteur *w* (b).

3.3 Recherche de périsegments : extraction des signatures et sélection des candidats

Pour trouver un certain nombre de périsegments pertinents, nous devons d'abord sélectionner des candidats parmi les 14 963 potentiels qui seront ensuite évalués par des experts. La sélection des candidats est effectuée à travers un modèle classique de recherche d'informations, dans lequel on calcule la similarité sémantique entre un vecteur de requête et les périsegments potentiels. L'étape préliminaire consiste donc dans la construction d'un ou plusieurs vecteurs de requête. Pour ce faire, nous profitons des composantes sémantiques identifiées préalablement. Puisque chaque cluster regroupe des segments *c* partageant des informations sémantiques similaires, nous voulons sélectionner et ordonner ces informations en générant un seul vecteur qui puisse les « résumer ». En d'autres termes, pour chaque composante sémantique, nous générons un seul vecteur, appelé sa *signature*, et nous l'utilisons pour la recherche des périsegments.

Pour construire ces vecteurs, plusieurs possibilités sont envisageables, comme l'utilisation du centroïde construit par le k-moyennes. Toutefois, nous avons opté pour l'utilisation d'un algorithme d'apprentissage supervisé, la machine à vecteurs de support linéaire (SVM), qui est systématiquement appliqué aux clusters de la concordance, afin d'obtenir un modèle par composante sémantique. Ce choix est justifié surtout par le fait qu'une telle méthode nous permet d'évaluer les performances des modèles construits. En outre, plusieurs travaux ont déjà utilisé le *vecteur des poids w* (en anglais, *weights vector*) construit par la SVM pour sélectionner et ordonner efficacement les variables les plus importantes d'un groupe de données (Brank *et al.*, 2002; Guyon *et al.*, 2002; Rakotomamonjy, 2003; Chang & Lin, 2008). Dans un contexte de classification de données textuelles, ce vecteur contient le poids de chaque mot selon sa pertinence pour la classification. Pour cette raison, nous

	Cluster 1			*Cluster 2*			*Cluster 3*		
	baseline : moy. F_1 = 0.47			baseline : moy. F_1 = 0.89			baseline : moy. F_1 = 0.94		
	Précision	**Rappel**	$\mathbf{F_1}$	**Précision**	**Rappel**	$\mathbf{F_1}$	**Précision**	**Rappel**	$\mathbf{F_1}$
0	0.81	0.89	0.85	0.98	0.99	0.99	1	0.99	1
1	0.81	0.70	0.75	0.88	0.77	0.82	0.86	0.92	0.89
Moy.	0.81	0.81	0.81	0.97	0.97	0.97	0.99	0.99	0.99

TABLE 1 – Rapport de classification pour les clusters n° 1, n° 2 et n° 3. La *baseline* classe chaque segment dans la classe majoritaire. 0 = classe "tous" ; 1 = classe "un", moy = moyenne ponderée. Pour des raisons d'espace, nous ne pouvons pas présenter tous les clusters.

le considérons comme étant représentatif des segments *c* observés, modélisant ainsi la signature de chaque composante sémantique.

Techniquement, nous procédons comme suit. La concordance est séparée de manière aléatoire en un sous-corpus d'apprentissage (70%) et un sous-corpus de test (30%). En suivant une stratégie de classification *un-contre-tous* (Rifkin et Klautau, 2004), nous utilisons une SVM pour discriminer les segments de chaque cluster par rapport à tous les autres. Les paramètres des modèles sont déterminés au moyen d'une validation croisée en dix échantillons et en suivant les indications fournies par Ng (Ng, 2011). Enfin, nous évaluons les modèles optimaux sur le sous-corpus de test et nous extrayons les vecteurs *w*, en ne retenant que leurs valeurs positives. Une signature pour chaque composante sémantique est ainsi obtenue (figure 2b). Les résultats de la classification sur les sous-corpus de test montrent que la classification par SVM est plus performante que la *baseline* (table 1), ce que nous permet d'utiliser avec efficacité les modèles générés.

Pour augmenter les performances de la recherche des périsegments, nous réduisons la dimensionnalité de l'espace sémantique global (Turney & Pantel, 2010) à 300 dimensions au moyen d'une décomposition en valeurs singulières (SVD) (Schütze & Pedersen, 1993). Enfin, nous utilisons la métrique cosinus pour sélectionner les dix candidats les plus similaires à chaque signature. Nous obtenons ainsi dix périsegments pour chaque composante sémantique. Autrement dit, pour chaque cluster exprimant une composante sémantique du concept par une forme canonique, on trouve plusieurs périsegments qui expriment ces composantes mais par des formes non canoniques.

4 Résultats, évaluation, conclusion

Un comité de trois experts a annoté chacun des périsegments candidats selon la pertinence, en suivant un protocole composé de trois catégories : directement lié à l'« esprit » (1) ; indirectement lié à l'« esprit » (2) ; non pertinent (3). Seuls les périsegments faisant partie des catégories (1) et (2) sont considérés comme pertinents. La table 2 résume donc les résultats pour les dix premiers périsegments propres aux dix composantes sémantiques retenues pour la validation. L'analyse des périsegments trouvés met en évidence que la majorité d'entre eux font partie de la deuxième catégorie. Les périsegments non pertinents peuvent être de deux types : ceux qui sont totalement non pertinents mais qui ont été récupérés en raison d'une utilisation ambigüe des mots propres à la composante du concept ; ceux qui laissent émerger un aspect de la composante qui ne peut pas être rattaché au concept à l'étude.

Composante	1	2	3	4	5	6	7	8	9	10
Précision	86,6%	93,3%	80 %	63,3%	90%	76,6%	86,6%	83,3%	96,6%	80%

TABLE 2 – Précision pour les dix périsegments de chaque composante sémantique. La précision moyenne est 83,63%.

Le principal obstacle de l'analyse conceptuelle assistée à partir de textes de haut niveau théorique, comme les textes philosophiques, dépend principalement de la nature des composantes du concept. Ainsi, comme l'a déjà souligné Deleuze et Guattari, le concept est souvent constitué d'autres concepts (Deleuze & Guattari, 2005), ce qui augmente le niveau d'abstraction des segments et périsegments à analyser. Par exemple, les moins bons résultats du cluster n° 4 proviennent du fait que la composante « matière » est un concept très général, utilisé par Peirce dans plusieurs contextes et traduit par un lexique plus ambigu que les autres composantes. Cependant, les résultats montrent qu'en général, la méthode permet de trouver des périsegments *pertinents* à des fins d'analyse conceptuelle et au moyen d'une assistance informatique, offrant ainsi une voie à poursuivre pour la recherche en ce domaine.

Notre travail a donc une nature exploratoire et utilise un corpus tout à fait original, mais ceci amène des obstacles au niveau de l'évaluation. À notre connaissance, il n'existe aucun corpus philosophique qui a été préalablement annoté à des fins d'évaluation de chaînes de traitement pour l'assistance à l'analyse conceptuelle. Cette absence nous oblige à utiliser des méthodes plutôt standards pour l'exploration des corpus en philosophie, limitant ainsi l'évaluation de notre méthode au calcul de sa précision. Des stratégies alternatives seront cependant envisagées dans des travaux futurs.

L'utilisation de l'ordinateur apporte sans aucun doute des avantages en termes de temps et de ressources utilisées pour ce genre de recherche, mais elle permet surtout la construction d'une approche empirique hybride pour l'AC qui vient renouveler, à la fois des points de vue de la méthodologie et de la théorie, la philosophie et les sciences humaines. Afin que le transfert des connaissances du « numérique » vers les humanités soit réalisé, nous croyons que cette approche doit se concrétiser dans des méthodes qui reproduisent partiellement les pratiques des chercheurs en sciences humaines. En suivant certaines étapes classiques de l'analyse conceptuelle en philosophie, telle l'élaboration d'une concordance, notre démarche contribue à la construction d'une stratégie performante pour l'analyse conceptuelle assistée par ordinateur.

Références

BEANEY M. (2015). Analysis. In E. N. ZALTA, Ed., *The Stanford Encyclopedia of Philosophy*. Spring 2015 edition.

BRANK J., GROBELNIK M., MILIC-FRAYLING N. & MLADENIC D. (2002). Feature selection using support vector machines. In *Proc. of the Third International Conference on Data Mining Methods and Databases for Engineering, Finance, and Other Fields* : Citeseer.

BUCKNER C., NIEPERT M. & ALLEN C. (2007). InPhO : The Indiana philosophy ontology. *APA Newsletters-Newsletter on Philosophy and Computers*, **7**(1), 26–28.

BYNUM T. W. & MOOR J. (1998). *The digital phoenix : How computers are changing philosophy*. Blackwell publishers edition.

CHANG Y.-W. & LIN C.-J. (2008). Feature ranking using linear SVM. *Causation and Prediction Challenge Challenges in Machine Learning*, **2**, 47.

DANIS J. (2012). *L'analyse conceptuelle de textes assistée par ordinateur (LACTAO) : une expérimentation appliquée au concept d'évolution dans l'œuvre d'Henri Bergson*. PhD thesis, Montréal, Université du Québec à Montréal.

DELEUZE G. & GUATTARI F. (2005). Qe'est-ce qu'un concept ? In *Qu'est-ce que la philosophie ?*, p. 21–37. Paris : Minuit.

DING X. (2013). A Text Mining Approach to Studying Matsushita's Management Thought. In *Proceedings of The Fifth International Conference on Information, Process, and Knowledge Management*, p. 36–39.

ESTÈVE R. (2008). Une approche lexicométrique de la durée bergsonienne. In *Actes des journées de la linguistique de corpus*, volume 3, p. 247–258.

FOREST D. & MEUNIER J.-G. (2005). NUMEXCO : A Text Mining Approach to Thematic Analysis of a Philosophical Corpus. *CH Working Papers*, **1**(1).

GIRJU R. & MOLDOVAN D. I. (2002). Text mining for causal relations. In *FLAIRS-02 Proceedings*, p. 360–364.

GUYON I., WESTON J., BARNHILL S. & VAPNIK V. (2002). Gene selection for cancer classification using support vector machines. *Machine learning*, **46**(1-3), 389–422.

JAIN A. K. (2010). Data clustering : 50 years beyond K-means. *Pattern recognition letters*, **31**(8), 651–666.

LAURENCE S. & MARGOLIS E. (2003). Concepts and Conceptual Analysis. *Philosophy and Phenomenological Research*, **LXVI**(no. 2), 253–282.

LAWRENCE J., REED C., ALLEN C., MCALISTER S., RAVENSCROFT A. & BOURGET D. (2014). Mining arguments from 19th century philosophical texts using topic based modelling. In *Proceedings of the First Workshop on Argumentation Mining*, p. 79–87 : Citeseer.

MCKINNON A. (1973). The conquest of fate in Kierkegaard. *CIRPHO*, **1**(1), 45–58.

MEUNIER J. G. & FOREST D. (2009). Lecture et analyse conceptuelle assistée par ordinateur : premières expériences. In *Le Priol et Dèscles (dir.), Annotations automatiques et recherche d'information*. Paris : Hermès.

NG A. (2011). Stanford University's Machine Learning Course. Lecture 62 - Regularization and Bias Variance. `https://fr.coursera.org/learn/machine-learning/lecture/4VDlf/regularization-and-bias-variance`. Consulté le : 31-05-2016.

PEDREGOSA F., VAROQUAUX G., GRAMFORT A., MICHEL V., THIRION B., GRISEL O., BLONDEL M., PRETTENHOFER P., WEISS R., DUBOURG V., VANDERPLAS J., PASSOS A., COURNAPEAU D., BRUCHER M., PERROT M. & DUCHESNAY E. (2011). Scikit-learn : Machine Learning in Python. *Journal of Machine Learning Research*, **12**, 2825–2830.

PEIRCE C. S. (1994). *The Collected Papers of Charles Sanders Peirce*. Virginia, U.S.A. : InteLex Corp. Charlottesville, electronic edition edition.

PINCEMIN B. (2007). Concordances et concordanciers : de l'art du bon KWAC. In *XVIIe colloque d'Albi Lagages et signification-Corpus en Lettres et Sciences sociales : des documents numériques à l'interprétation*, p. 33–42.

RAKOTOMAMONJY A. (2003). Variable Selection Using Svm Based Criteria. *J. Mach. Learn. Res.*, **3**, 1357–1370.

ROBERTSON S. (2004). Understanding inverse document frequency : on theoretical arguments for IDF. *Journal of Documentation*, **60**(5), 503–520.

SAINTE-MARIE M., MEUNIER J.-G., PAYETTE N. & CHARTIER J.-F. (2010). Reading Darwin between the lines : a computer-assisted analysis of the concept of evolution in the Origin of species. In *10th International Conference on Statistical Analysis of Textual Data*.

SCHÜTZE H. & PEDERSEN J. (1993). A vector model for syntagmatic and paradigmatic relatedness. In *Proceedings of the 9th Annual Conference of the UW Centre for the New OED and Text Research*, p. 104–113 : Citeseer.

SCHWARTZ H. A., PARK G., SAP M., WEINGARTEN E., EICHSTAEDT J., KERN M., BERGER J., SELIGMAN M. & UNGAR L. (2015). Extracting Human Temporal Orientation in Facebook Language. In *Proceedings of the The 2015 Conference of the North American Chapter of the Association for Computational Linguistics-Human Language Technologies*.

STEINBACH M., KARYPIS G. & KUMAR V. (2000). A Comparison of Document Clustering Techniques. *KDD workshop on text mining*, **400**, 1–2.

TURNEY P. D. & PANTEL P. (2010). From frequency to meaning : Vector space models of semantics. *Journal of artificial intelligence research*, **37**(1), 141–188.