

Exploitation de modèles neuronaux en vue de la détection automatique des nominations émergentes

Yumeng Ding

▶ **To cite this version:**

Yumeng Ding. Exploitation de modèles neuronaux en vue de la détection automatique des nominations émergentes. Sciences de l'Homme et Société. 2021. dumas-03485460

HAL Id: dumas-03485460

https://dumas.ccsd.cnrs.fr/dumas-03485460

Submitted on 17 Dec 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Exploitation de modèles neuronaux en vue de la détection automatique des nominations émergentes

YUMENG

DING

Sous la direction de Oliver Kraif

Stage co-encadré par Agata Jackiewicz

Laboratoires : LIDILEM - PRAXILING

UFR LLASIC

Département Sciences du Langage

Mémoire de master 2 Sciences du Langage – 20 crédits

Parcours : Industries de la Langue

Année universitaire 2020-2021

Exploitation de modèles neuronaux en vue de la détection automatique des nominations émergentes

YUMENG

DING

Sous la direction de Oliver Kraif

Stage co-encadré par Agata Jackiewicz

Laboratoires : LIDILEM - PRAXILING

UFR LLASIC

Département Sciences du Langage

Section Industries de la Langue

Mémoire de master 2 Sciences du Langage – 20 crédits

Parcours : Industries de la Langue, orientation Recherche

Année universitaire 2020-2021

Remerciements

Mes études en France touchent à leur fin. Pendant ces deux ans de master, j'ai eu des expériences mémorables, mais aussi des difficultés dans mes études et dans la vie. Je tiens à remercier les personnes qui m'ont beaucoup aidée, mes parents, mes professeurs et mes amis.

Un grand merci à Olivier Kraif, mon tuteur de stage et de mémoire, à Agata Jackiewicz, mon encadrant de stage et à Claude Ponton, mon enseignant référent pour avoir pris le temps de me diriger, pour leurs conseils et leur soutien.

J'aimerais également remercier à mes amis de classe du Master Industries de la Langue qui m'ont aidée au cours de mon master pour leur soutien et leurs encouragements.

Pour finir, je tiens à remercier mes parents, Hongwu et Deli, pour leur compréhension, leur soutien et leurs encouragements dans tous les moments difficiles.

Financement

Ce stage a été financé grâce à la chaire MIAI « Artificial Intelligence and Language »[1], ainsi que le serveur ayant permis de réaliser les expérimentations. Un grand merci à Laurent Besacier et François Portet pour leur soutien dans ce projet.

[1] https://miai.univ-grenoble-alpes.fr/research/chairs/perception-interaction/artificial-intelligence-language-850480.kjsp?RH=6499587813011763

Déclaration ANTI-PLAGIAT

1. Ce travail est le fruit d'un travail personnel et constitue un document original.
2. Je sais que prétendre être l'auteur d'un travail écrit par une autre personne est une pratique sévèrement sanctionnée par la loi.
3. Personne d'autre que moi n'a le droit de faire valoir ce travail, en totalité ou en partie, comme le sien.
4. Les propos repris mot à mot à d'autres auteurs figurent entre guillemets (citations).
5. Les écrits sur lesquels je m'appuie dans ce mémoire sont systématiquement référencés selon un système de renvoi bibliographique clair et précis.

PRENOM : ……YUMENG……

NOM : ……DING……

DATE : ……19 août 2021……

Sommaire

Remerciements 2

Sommaire 4

Introduction 6

Partie 1 - Contexte et Problématique 10

CHAPITRE 1. CONTEXTE ET PROBLEMATIQUE 11

1. CONTEXTE 11
2. PROBLEMATIQUE 12

Partie 2 - État de l'art 18

CHAPITRE 2. MODELISATION 19

1. APPROCHES POUR LE REPERAGE DES NOMINATIONS 19
2. METHODOLOGIE DE REPERAGE ET D'ANALYSE DE LA NOMINATION EMERGENTE 22
3. LES METHODES TEXTOMETRIQUES 25
4. APPROCHES DISTRIBUTIONNELLES POUR LE GLISSEMENT SEMANTIQUE 29

Partie 3 - Expérimentation basée sur l'apprentissage profond 40

CHAPITRE 3. CONSTRUCTION DU CORPUS 41

1. RECOLTE DES DONNEES 42
2. CORPUS FINAL 45
3. ANNOTATION DU CORPUS 46

CHAPITRE 4. JEUX DE DONNEES 50

1. JEUX DE DONNEES POUR LES CLASSIFIEURS 50
2. JEUX DE DONNEES POUR LE MODELE MLM 55

CHAPITRE 5. CONSTRUCTION ET ENTRAINEMENT DU CLASSIFIEUR 57

1. JEUX DE DONNEES 57
2. CLASSIFIEUR BASE SUR LE FNN 58
3. CLASSIFIEUR BASE SUR LE FLAUBERT 65
4. CLASSIFIEUR FINAL 68

CHAPITRE 6. MODELE MLM DE BERT 69

1. EXPERIENCES 69
2. RESULTATS 70

Partie 4 - Résultats et discussions 72

CHAPITRE 7. ELARGISSEMENT DES DONNEES 73

1. CONSTRUCTION DU CORPUS ELARGI 73
2. RESULTAT DE L'EVALUATION 75
CHAPITRE 8. DISCUSSIONS 78
Conclusion 80
Bibliographie 82
Sitographie 85
Glossaire 86
Sigles et abréviations utilisés 87
Table des illustrations 88
Table des annexes 90
Table des matières 94

Introduction

« La ville durable est-elle une ville qui dure ? », « comment les villes peuvent-elles devenir durable ? ». Si l'on rencontre la première fois l'expression « ville durable » dans les journaux, on pourra avoir du mal à interpréter la notion. « La ville durable est une ville se développant durablement… ». Normalement, nos questions trouvent leur réponse dans les explications qui suivent ce nouvel usage.

En ce qui concerne l'étude de l'écologie, « ville durable » peut constituer un terme désignant un concept dans ce domaine professionnel ou technique spécifique. Mais dans les discours quotidiens, principalement dans les productions médiatiques et politiques, on trouvera nombre de ces *nominations* sans définition précise, ne pouvant prétendre au statut de terme qui lui confère les discours spécialisés. L'emploi d'expressions aux contours assez flous tels que *la France* ou l*'honneur de la France*, *patron* ou *camarade*, *partage* ou *solidarité,* est normal et fréquent (Siblot, 2001). Il semble nécessaire dans le discours politique, car ces nominations vont transmettre au public bon nombre d'informations comme la prise de position, les points de vue de l'orateur sur cette entité. En même temps, ces nominations jouent un rôle dialogal et énonciatif, et invitent le public à participer à la discussion. Dans les usages récents de l'adjectif *durable*, nombre de nouvelles nominations sont apparues, puis se sont stabilisées, non seulement dans leur forme, mais aussi au plan sémantique. Elles sont le fruit d'un processus dynamique et interactionnel, dans lequel les interdiscours sont énoncés, et non seulement évoqués (Calabrese, 2015).

En effet, selon Jackiewicz et Pengam :

> « Des enjeux sociaux et politiques importants entourent ces formes de catégorisations, en ce qu'elles attestent de visions des choses, de prises de positions, d'engagements idéologiques et d'enjeux identitaires particuliers » (Jackiewicz, Pengam, 2020 : 2).

Ces enjeux sont à la source de l'intérêt pour l'étude du phénomène de nomination, notamment en analysant des discours politiques.

En outre, le sens de ces expressions peut s'éloigner du sens des mots en surface – ce sens doit être découvert dans la praxis. Dans ce cas, c'est la nomination qui fournit le lien entre le langage et le réel (Siblot, 2001). C'est pourquoi les modalités de la production de nomination et la relation entre les nominations et les dénominations (ces dernières étant stabilisées dans la langue), intéressent de nombreux chercheurs en linguistique.

En TAL, le repérage des nominations émergentes est une question plus intéressante pour nous. Fruit d'un processus en cours, dans un état encore intermédiaire, les nominations émergentes sont plus stables que les initiatives singulières d'un seul locuteur. Elles peuvent également être considérées comme un indicateur de nouvelles tendances dans les discussions. Vu le rôle des nominations émergentes dans l'élaboration de nouveaux concepts ou points de vue, il peut être utile de les détecter. D'autant que les nominations émergentes représentent les attitudes des locuteurs majoritaires. Leur détection permet d'identifier des thématiques sociales émergentes, et nous pouvons analyser finement la mécanique discursive à travers ces nominations détectées.

Sur le plan linguistique, les nominations émergentes peuvent être traitées comme une forme de pré-néologie, permettant différentes approches. D'une part, on peut identifier précocement les futures dénominations ou néologies dans une perspective terminologique. D'autre part, selon Jackiewicz et Pengam, les nominations émergentes indiquent le point de vue et la prise de position du locuteur (Jackiewicz, Pengam, 2020 : 2), et leur détection aide à situer l'analyse au cœur des controverses politiques. Enfin, comme elles se manifestent souvent par un glissement de sens au cours de l'élaboration d'une nouvelle pensée, il est intéressant de pointer ces glissements de sens à travers leur repérage afin de décoder et de mettre à jour les processus à l'œuvre dans la création de nouveaux mots et de nouveaux concepts.

Le problème de la détection des nominations est qu'elle engage un processus interprétatif complexe, qui amène l'interprète à dépasser le seul niveau du texte : il faut pouvoir juger qu'une expression introduit un sens nouveau, et que ce même sens est commun à différentes occurrences, à travers des textes et des contextes variés. Une nomination émergente ne peut être qu'un hapax : c'est sa répétition et sa circulation à travers différents discours qui lui confère son statut émergent. Face à cette démarche interprétative, nécessairement manuelle, se pose la question des processus réalisés par les technologies informatiques. Si certaines étapes peuvent être automatiquement effectuées par la machine, c'est au chercheur, *in fine*, qu'il incombe de contrôler le processus. Autrement dit la détection, pour une notion aussi délicate, ne pourra être exécutée de manière totalement automatique.

La question est donc de sélectionner au mieux des candidats pour aider l'analyse manuelle. Avec le développement de l'apprentissage automatique et profond, avec l'élaboration des réseaux de neurones artificiels et leurs capacités étonnantes à encoder la

sémantique des phrases et des textes, on peut espérer entrainer les machines à résoudre une partie du problème. Basés sur de grands volumes de données textuels, des modèles de réseaux de neurones telles que les *transformers* (Vaswani, et al., 2017 permettent d'apprendre automatiquement de nombreuses relations liées aux contextes des mots. En s'appuyant sur des indices pertinents, de tels systèmes sont peut-être en mesure d'identifier ce qui caractérise les nominations émergentes au sein de leur contexte.

Dans la perspective d'un apprentissage supervisé, à part un bon algorithme, il faut constituer un jeu de données suffisant pour l'apprentissage, afin de permettre au système d'identifier en contexte des traits utiles pour la classification.

Plusieurs indices caractéristiques des nominations ont été relevés dans les études existantes : selon l'approche de l'ajustement qui est « un réseau de relateurs pour dessiner les contours des catégories et leurs configurations » (Jackiewicz, Pengam, 2020 : 10), les nominations peuvent être repérées par les autres entités en explorant les relations entre eux. Par conséquent, les contextes (les mots, expressions ou ponctuations) qui indiquent les relations peuvent constituer des marqueurs pertinents. Par exemple, dans la phrase suivante :

« pour faciliter le développement des mobilités douces [vélo, trottinettes...] »,

Les crochets sont les indices d'explicitation, et le contenu entre crochets expliquent la nomination « mobilités douces » en listant des exemples. Cet effort d'explicitation produit par le rédacteur est une marque dialogale de la prise en compte du lecteur, dont on suppose qu'il n'est pas encore familier de la nomination. En multipliant les observations, on constate qu'il est fréquent de trouver des indices intéressants dans le contexte.

Par ailleurs, la forme même de la nomination émergente peut constituer un indice. En observant les nominations en diachronie, à travers des corpus, on constate qu'elles correspondent souvent à des combinaisons nouvelles. La plupart du temps, elles se manifestent par des combinaisons lexicales impliquant des mots existants (ex. *ville durable*, *ville verte*, *économie punitive*), mais qui ne cooccurrent que dans l'usage de cette nouvelle nomination. C'est pourquoi nous proposons d'attacher de l'importance aux méthodes textométriques, qui permettront également d'identifier des candidats dans une démarche de comparaison en diachronie.

En outre, comme les nominations émergentes ont pour fonction de nommer une nouvelle réalité, et qu'elles réalisent ainsi un glissement au plan sémantique, il est

vraisemblable que ces combinaisons soient difficiles à prédire pour des systèmes entrainés sur des textes ou ces nominations sont absentes.

Pour répondre aux idées précédentes, quelques pistes de recherche nous semblent prometteuses. Concernant les indices en contexte, nous nous intéresserons notamment aux contextes riches en connaissance ; pour caractériser les combinaisons lexicales nouvelles, les pistes textométriques seront nécessaires ; et autour du glissement de sens et de la prédictibilité, les approches distributionnelles nous guideront.

Nous nous proposons d'aborder le sujet selon deux perspectives. Sur le plan linguistique, d'abord, nous proposons d'étudier les caractéristiques des nominations émergentes en vue de trouver des indices permettant afin de les identifier ; et du côté informatique, nous proposons d'employer des modèles novateurs susceptibles d'appréhender la sémantique globale des énoncés, en recourant aux techniques de l’apprentissage profond.

Dans une première partie, nous présenterons le contexte du stage et la problématique de notre recherche, après avoir défini certaines notions autour du concept de nomination.

Dans une deuxième partie, on développera l'état de l'art en lien avec notre sujet de recherche. Cette partie sera consacrée aux méthodologies et techniques qui nous semblent plus prometteuses et inspirantes, et nous la terminerons par l'énoncé de nos hypothèses de travail.

Dans une troisième partie, nous présenterons notre expérimentation basée sur les modèles de réseau de neurones. Celle-ci se déroule en quatre étapes : la construction du corpus, des jeux de données, la construction et l'entrainement du classifieur et les tests sur modèle MLM ("Masked language model") de BERT. Nous détaillerons l’objectif de chaque partie et les démarches associées, ainsi que les résultats de chaque expérience.

Enfin, nous analyserons les résultats produits par le classifieur et MLM sur un corpus plus étendu, afin d'évaluer de manière critique les limites de la méthode. Nous discuterons également des perspectives et des limitations liées à nos démarches expérimentales.

Partie 1

-

Contexte et Problématique

Chapitre 1. Contexte et problématique

1. Contexte

Mon mémoire s'inscrit dans le cadre du stage du laboratoire de linguistique et didactique des langues étrangères et maternelles (LIDILEM). Il est financé par le budget MIAI géré par le laboratoire d'informatique de Grenoble (LIG). Ce stage est principalement encadré par Oliver Kraif du laboratoire LIDILEM à l'Université Grenoble Alpes, et co-encadré par Agata Jackiewicz, Professeur de linguistique à l'Université Paul Valéry - Montpellier III.

1.1. Laboratoire LIDILEM

Le Laboratoire de Linguistique et Didactique des Langues Etrangères et Maternelles (LIDILEM) a été fondé en 1987, s'est constitué par cinq centres de recherche. Actuellement une soixantaine de membres permanents et environ 70 doctorants y sont rassemblés et travaillent ensemble dans quatre axes principaux de recherche :

- Description et modélisation linguistiques, corpus, TAL
- Didactique des langues : analyse et évaluation des processus d'enseignement / apprentissage
- Acquisition du langage : multimodalité, variabilité et contexte
- Sociolinguistique : identités, cultures, interactions, usages

Dans ce laboratoire, mon stage a été encadré par M. Olivier Kraif, et la thématique de mon mémoire se situe dans le premier axe de recherche, et plus précisément, il étudie les nominations émergentes à travers des méthodes et outils informatiques. Le but de ce travail est d'étudier la détection automatique des nominations émergentes à travers l'apprentissage profond.

1.2. MIAI

L'Institut MIAI Grenoble Alpes (Multidisciplinary Institute in Artificial Intelligence) est un établissement éducatif fondé en 2019. Il vise à conduire des recherches au plus haut niveau dans le domaine de l'intelligence artificielle, principalement dans deux aspects sur sept axes : IA du futur et IA pour l'humain et l'environnement.

MIAI propose des formations de haute qualité pour les étudiants et les professionnels de tous les niveaux, il encourage également les innovations dans les entreprises. Le principe est de soutenir le développement de tous les aspects de l'IA et de le généraliser et l'apporter dans la vie de tous les citoyens.

2. Problématique

La détection des nominations émergentes présente un double intérêt pour le TAL : pour l'identification et le traitement des néologismes d'une part, et pour aider à l'analyse du discours d'autre part.

Quant à l'étude de néologie futures et d'analyse de discours, le repérage des nominations émergente a l'intérêt en TAL. Avant de commencer la détection, nous devons définir ici les notions de nomination émergente et de sa famille.

2.1. Les notions de base autour de la nomination

A partir des années 80, le débat sur la dénomination se développe, et la notion de nomination est introduite comme une paire de concepts. Jusqu'aux années 90, l'étude de la nomination est prise en considération par les chercheurs dans le cadre de l'analyse du discours.

La nomination est toujours mentionnée en accompagnement de la notion de dénomination, car les nominations peuvent se transformer en dénominations si elles reçoivent la sanction de l'usage et finissent par être lexicalisées (Frath, 2015). Ainsi, les deux concepts sont étroitement liés.

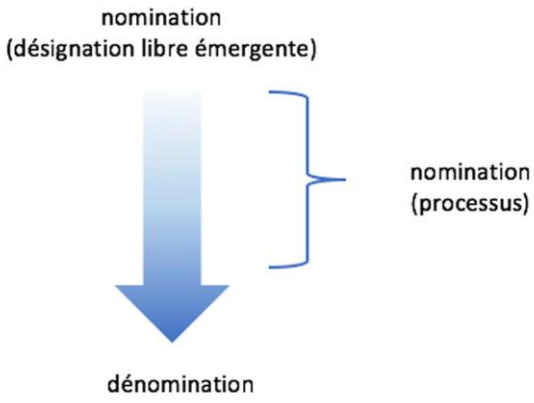


Figure 1. Transformation en dénomination (Pengam, Jackiewicz, 2019)

2.1.1. Dénomination, nomination et nomination émergente

Pour bien distinguer les deux concepts, voici la définition de la dénomination selon Detrie, Siblot et Vérine :

> « Une dénomination est, de façon stricte, la désignation d'une chose ou d'une personne par un nom, mais l'usage a étendu le terme aux catégorisations adjectivales et verbales. La dénomination est de la sorte du côté de la langue entendue comme une nomenclature d'étiquettes, celle dont les dictionnaires dressent l'inventaire et recensent les sens véhiculés par les discours. Elle s'oppose au processus de nomination, acte d'un sujet qui tout à la fois nomme et catégorise dans l'actualisation discursive. » (Detrie, Siblot & Vérine, 2001 : 76).

Partons de la définition suivante de Siblot : les nominations sont des « outils du langage produisant leur sens, de l'indication d'une praxis culturelle et religieuse qui, étrangère et d'apparition récente en France, se trouve n'être pas encore catégorisée en langue » (Siblot, 1992 : 8-9). Il s'agit donc d'un acte consistant à créer un sens nouveau. « La nomination est bien un acte de langage, l'acte premier de toute production de sens. » (Detrie, Siblot & Vérine, 2001 : 205). Notons que dans cette définition, la nomination est précisée comme un acte qui est lié au réel, donc au référent, avec une forme qui n'est pas encore stabilisée dans la langue.

Selon une autre définition, la nomination consiste à « cerner et construire en discours de nouveaux objets qui apparaissent dans [leur] expérience collective », elle « est une tentative pour donner un nom à un nouvel objet de notre expérience collective. » (Frath, 2015 : 36, 43). Ici, l'expérience collective est un paramètre essentiel.

Pour résumer, on peut dire que :

- La nomination est un acte dynamique et désignatif par lequel un objet nouveau est nommé en discours ;
- La nomination est contextuelle, et doit être interprétée en fonction d'un environnement culturel précis ;
- La nomination n'est pas stabilisée dans la langue, car elle est construite dans le discours.

Une fois posée cette notion de nomination, il faut préciser cette notion d'"émergence". L'idée est qu'il s'agit d'une forme intermédiaire entre nomination et dénomination, précisément, elle prise est dans un processus de stabilisation (cf. le schéma de la fig.1). Vu que la nomination d'innovations n'aboutit pas nécessairement à la dénomination et à la néologie quand elle reste purement discursive (Sablayrolles, 2007), par rapport à celle-là,

la nomination émergente est plus stable que l'initiative individuelle car elle est reprise et partagée par différents discours.

2.1.2. Définition et néologie

À part la dénomination, deux concepts qui se rapprochent de la nomination doivent en être distingués : la définition et la néologie.

D'abord, la définition est le fait d'expliciter le sens d'un mot. Les points communs avec la nomination sont qu'elles sont toutes deux issues d'un acte dynamique d'appréhension de nouveaux objets par le locuteur, et qu'elles désignent les nouveautés. Cependant la différence est aussi très claire, par rapport à la nomination aboutissant à un nom de manière relativement arbitraire et synthétique, la définition assume une fonction explicitante : elle doit donner une description logique, objective et pertinente. En plus, à travers la définition, nous ne pouvons plus déduire la prise de position ou les points de vue du locuteur. La définition est surtout liée au produit d'un effort d'explicitation et d'objectivation, tandis que la nomination, généralement, travaille sur des catégories implicites.

Ensuite, la néologie désigne la création des nouveaux mots dans la langue. Elle s'intéresse aux différents procédés de formation (morphologiques, syntaxiques, sémantiques) des nouveaux mots ou expressions. Comme le note Sablayrolles (2007), certaines nominations aboutissent à des néologismes mais pas toutes, et par ailleurs, bon nombre de néologismes générés ont d'autres raisons que la nomination de nouveaux concepts. Il convient donc de distinguer les deux notions.

Concernant la néologie, celle-ci désigne le processus de création des nouveaux mots dans la langue. En rapprochant les deux concepts, nous constatons que la néologie est correspondue à un état stabilisé et un candidat inclus dans le dictionnaire. Et une caractéristique importante pour se différencier, c'est que le néologisme signifie obligatoirement une nouvelle réalité et il n'est pas toujours nominatif. Pourtant les nominations n'ont pas toujours un nouveau sens. En effet, certaines nominations aboutissent à des néologies, néanmoins, bon nombre de néologismes générés ont d'autres raisons que la nomination des nouveaux concepts, de même, la nomination d'innovations ne se transforme pas nécessairement en néologie (Sablayrolles, 2007).

2.1.3. Néologie et émergence

On ne peut pas dire que toute nomination est néologique car il existe des nominations sans nouveau signifiant, et réciproquement il existe des néologismes non nominatifs (Sablayrolles, 2007). Néanmoins, ils s'agissent là de cas spéciaux et peu fréquents pour distinguer la néologie et la nomination. En général, les nominations de nouvelles réalités fonctionnent comme les néologismes.

Une autre différence qui empêche d'assimiler nomination et néologisme, c'est que la nomination est instable et individuelle. Pour représenter un nouveau concept, il y aura un bon nombre d'initiatives de nominations. Mais au cours des échanges entre locuteurs, les candidats nominations sont filtrés, et sont soit partagés par la communauté, soit remplacés par une autre nomination. La nomination émergente est justement le fruit de ce processus de communication, et désigne un état intermédiaire, une transition entre la nomination et le néologisme. Etant donné que par rapport aux nominations, les nominations émergentes ont déjà passé la sélection de la majorité, elles sont plus stables et mûres que les expressions individuelles, autrement dit, elles sont plus proches du néologisme, qui désigne la fin de ce processus de création.

En tout cas, en tant que forme en cours de stabilisation, la nomination émergente est liée étroitement au néologisme (qui représente l'état final). C'est pourquoi, du côté de la néologie, il semble que la nomination émergente joue un rôle de pré-néologie.

2.2. Problématique

Dans cette étude, l'objectif est de proposer des méthodes de manière à assister à la détection des nominations émergentes et à fournir de nouveaux outils. Nous pouvons traiter ce problème en combinant les connaissances de deux domaines : linguistique et informatique.

Les difficultés liées à cette problématique sont nombreuses :

- d'une part, la nomination est un acte dynamique. Son sens et emploi sont changeants en fonction de l'environnement actuel du contexte. Cela veut dire que la nomination s'explique et se caractérise par le contexte et la situation d'énonciation où s'est accompli cet acte, et selon lesquels nous pouvons faire des hypothèses interprétatives. Néanmoins, la nomination est construite dans le discours, elle est de nature discursive. Elle est donc complexe à appréhender puisqu'elle demande d'intégrer à la fois la dimension textuelle et

les composantes pragmatiques avec le contexte socio-culturel. Même si la nomination se compose des mots existants et familiers, il existe encore un écart entre le sens compositionnel et le sens émergent visé par le discours. Ce glissement de sens rend plus difficile la compréhension. Autrement dit, pour ne rien perdre, nous devons analyser le contexte et la praxis à la fois.

- d'autre part, la nomination est instable et variable, il y aurait bon nombre de nominations diverses désignant une même nouveauté.

Globalement, les questions qui se posent sont les suivantes : comment peut-on identifier automatiquement les nominations émergentes ? Quelles composantes du contexte peut-on analyser pour les identifier ? Au plan linguistique, de quelle façon caractérise-t-on le contexte ? Quelles sont les formes linguistiques qui apparaissent autour de la nomination ? L'analyse doit-elle porter sur les niveaux morphosyntaxique, sémantique ou pragmatique ?

Au plan morphosyntaxique, les questions sont nombreuses : y a-t-il des marques de surface facilement exploitables qui permettent de caractériser les nominations (comme les guillemets, les expressions de dissimilation) ? Existe-t-il des modes de formation productifs aboutissant à des nominations repérables dans leur forme, comme dans *vaccino-hésitant, climato-sceptique, ...* ? Est-ce que les nominations émergentes se manifestent plus souvent par des combinaisons lexicales impliquant des mots existants ? Par exemple, des patterns plus productifs, tel que N+ADJ (*ville durabl*e, *ville verte, économie punitive*) ? Quelles sont les parties plus stables dans ces patterns ?

Aux plans sémantiques et pragmatiques, le sujet soulève également des questions complexes : comment peut-on identifier la création d'un nouveau sens, qui ne serait pas encore enregistré dans le dictionnaire ? Peut-on utiliser les méthodes de détection de néologie ? s'appliquent-elles à la néologie sémantique afin de détecter les glissements de sens ? Ces glissements de sens s'appliquent-ils à toute la nomination ou juste à un de ses composants ? Comment peut-on saisir la nouveauté nommée par des nominations, à travers le contexte ou le praxème ? Est-ce qu'on peut trouver une autre entité considérée comme une référence dans le contexte ? Quelle relation ont-elles entre les deux ?

Au plan informatique, enfin, quels algorithmes peut-on mettre en œuvre pour la tâche de l'identification ? Peut-on entrainer un système à reconnaitre automatiquement des traits contextuels des nominations émergentes ? Dans le cas d'un glissement sémantique, ce

glissement peut-il être identifié par des phénomènes distributionnels, et donc par des modèles vectoriels de *word embedding* ou de *transformers* pré-entrainés tels que BERT ? Et les modèles prédictifs peuvent-ils permettent de mesurer le degré de nouveauté d'une association entre deux mots (comme *ville* + durable*)* ?

On le voit les questions sont nombreuses et complexes.

Pour arriver à y répondre sur un plan expérimental, des questions d'ordre méthodologiques devront être abordées : pour construire un corpus, quelle structure et quels outils sont plus pratiques ? Comment assure-t-on la fiabilité de l'annotation ?

Plus précisément, pour réaliser l'identification automatique, quelle tâche de TAL se rapproche le plus rapproche de notre recherche pour que nous nous en inspirions ? Quel réseau de neurones est plus adapté à nos besoins ? Si on utilise des modèles préentrainés d'apprentissage profond, quelle couche de neurones permettra d'obtenir une meilleure performance ? Lors de l'entraînement, comment initialiser et adapter les paramètres ? Comment éviter le sur-apprentissage et le sous-apprentissage ?

Dans la section suivante, nous effectuons un tour d'horizon de l'état de l'art, afin de d'examiner différentes méthodes dédiées au repérage des nominations et de dégager les pistes les plus prometteuse qui nous ont inspiré dans nos travaux. Nous terminons par l'établissement de nos hypothèses de travail.

Partie 2

-

État de l'art

Chapitre 2. Modélisation

Comme expliqué précédemment, ce travail vise à proposer des méthodes pour aider à détecter de façon automatique les nominations émergentes. Pour bien concevoir un tel outil, il faut voir dans un premier temps ce que dit l'état des travaux actuels autour de la détection de la nomination. En analysant des diverses approches TAL, nous pouvons en tirer des pistes de recherche prometteuses.

D'après nos observations, les nominations émergentes sont le plus souvent formées à partir de combinaisons de mots. Nous détaillerons donc, dans un second temps, ce que nous nommons la piste textométrique.

Ensuite, il peut être intéressant d'aborder la littérature sur les approches dites distributionnelles, afin d'en dégager des pistes inspirantes pour l'identification des glissements sémantiques.

Pour terminer, nous clôturerons cette partie par un résumé des pistes principales guidant nos propres travaux, et nous formulerons nos hypothèses de travail.

1. Approches pour le repérage des nominations

Le repérage des nominations émergentes est un champ de recherche intéressant de nos jours dans l'étude de nomination. Il permet d'identifier précocement les futures dénominations ou néologies par exemple dans une perspective terminologique, mais aussi il favorise la mise à jour des processus à l'œuvre dans la création de nouveaux mots et de nouveaux concepts. En outre, nous pouvons analyser finement la mécanique discursive à travers les nominations.

En 2020, dans le cadre du projet ANR TALAD (Traitement Automatique des Langues et Analyse du Discours), Longhi et al. (2020) ont mené des approches sur le repérage de nomination pour fournir une aide exploratoire au chercheur en analyse du discours (AD). Ils proposent les différentes approches suivantes :

1.1. L'approche statistique

Cette approche textométrique de la nomination a recours à des outils de textométrie, tel qu'IraMuTeq[2], afin d'explorer le corpus (Longhi, et al., 2020). Elle permet de rendre compte de caractéristiques significatives des données. Les chercheurs constituent des sacs de mots à partir des contextes, puis rapprochent ces contextes en fonction de leur similarité. Ils en tirent différents thématiques associés au mot pôle.

Le corpus est un ensemble de documents qui se compose de l'ensemble des données, y compris des métadonnées. Et la nature des textes rassemblés est présentée à la fois par un balisage ajouté, précédé de ses métadonnées. Le traitement est fait par le logiciel Iramuteq (2009).

Le premier traitement consiste à filtrer dans tout le corpus, l'ensemble des textes qui contiennent un terme ciblé. Ensuite, il s'agit de construire un sous-corpus focalisé sur ce terme avec l'ensemble des séquences qui contiennent ce terme. Ensuite on analyse ce sous-corpus en procédant à l'élaboration d'une classification hiérarchique descendante des contextes (CHD).

Le tableau 1 est un résumé des avantages et des inconvénients de la méthode textométrique.

Méthode	Avantage	Inconvénient
● « Détourner » la CHD ● Explorer les corpus à partir de formes lexicales isolées (listées dans Lexico ou à l'aide d'expressions régulières) et construire les sacs de mots	● Traiter un grand corpus ● Repérer les nominations ayant la structure similaire au niveau morphosyntaxique ● Comparer la situation de l'usage (le contexte)	● Manquer l'analyse textuelle, en particulier au niveau syntaxique et pragmatique ● Manquer l'exploitation de variantes de nomination

[2] IRaMuTeQ (pour « Interface de R pour les Analyses Multidimensionnelles de Textes et de Questionnaires ») est un logiciel libre et ouvert. Reposant sur le langage R et python, il supporte la langue française, anglaise, espagnole et portugaise. Il permet de faire des analyses statistiques à travers la méthode de classification de Max Reinert (classification hiérarchique descendante CHD sur un tableau croisant les formes pleines et des segments de texte).

Table 1. L'approche textométrique

1.2. Les approches à base de linguistique

1.2.1. Le traitement « manuel » des nominations

Cette méthodologie est issue de l'analyse du discours de tradition française (Longhi 2015). Premièrement il a besoin d'établir des corpus spécifiques en lien avec une problématique spécifique. Deuxièmement il consiste à chercher manuellement les nominations existantes sur un même sujet. A travers l'analyse syntaxique, sémantique et pragmatique, l'analyse d'indices contextuels tels que guillemets, commentaires métadiscursifs, reformulations, enclosures ou définitions est aussi prise en compte, les experts repèrent les nominations potentielles manuellement (Longhi, et al, 2020).

Nous voyons un résumé des avantages et des inconvénients de la méthode manuelle dans le tableau 2.

Méthode	Avantage	Inconvénient
● Une consultation manuelle des occurrences par le chercheur, y compris la densité de l'apparition des formes, et les procédés de mise en discours	● Analyser à la fois la forme repérée et son environnement immédiat ● Les analyses textuelles sont plus fines sur des séquences précises, la syntaxe et la pragmatique sont prises en compte ● Plus de possibilités sur les nominations émergentes dans un même corpus sont proposées	● Très coûteuse en temps et en ressources humaines. ● Le regard du chercheur est subjectif ● Ne permets pas de traiter des données de grands volumes

Table 2. L'approche du traitement « manuel »

1.2.2. Repérage des nominations par détection des chaînes de coréférence

Cette approche consiste en une analyse purement syntaxico-pragmatique, dont le cœur est la coréférence. Elle permet de repérer les variantes d'une même nomination. Pour

commencer, il s'agit d'identifier les chaînes de coréférence qui contiennent la nomination étudiée. Ces chaînes peuvent être construites de manière automatique (Grobol, 2019). Ensuite, elle consiste à étudier les reprises coréférentielles du terme qui seront souvent des reformulations de cette nomination (Longhi, et al, 2020).

Le tableau 3 condense des avantages et des inconvénients de la méthode de repérage par la détection des chaînes de coréférence.

Méthode	Avantage	Inconvénient
● Construire les chaînes de coréférence ● Détecter des mentions par le module neuronal ● Classifier les mentions par SVM ou des forêts d'arbres aléatoires ● L'analyse experte	● Explorer automatiquement des variations de nomination émergente ● Le processus de l'analyse est explicable et intelligible	● Les nominations étudiées sont contraintes par la relation de coréférence

Table 3. L'approche de repérage par la détection des chaînes de coréférence

2. Méthodologie de repérage et d'analyse de la nomination émergente

Jackiewicz & Pengam (2020) propose une méthodologie générale de repérage et d'analyse des nominations émergentes qui permet de conduire l'interprétation des descriptions référentielles en rendant compte d'une série d'indices contextuels dans les discours.

2.1. Repérage par relations statiques

Repéré à l'aide de relateurs sémantico-logiques intervenant dans les procédés d'ajustement référentiel, l'idée est d'identifier les relations statiques que les concepts entretiennent avec d'autres entités mieux connues ou « déjà-là ».

Le lien entre une entité repérée X et une entité repère Y peut se catégoriser en trois grands types : l'identification, la différenciation et la ruption (Jackiewicz, Pengam, 2020).

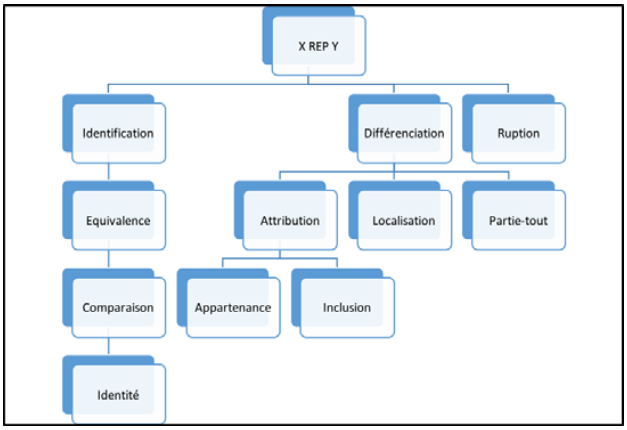


Figure 2. Réseau des relateurs de repérage (Jackiewicz, Pengam, 2020)

Précisément, l'identification découle de relations réflexives et symétrisables entre X et Y, X est identifié à l'aide de Y. Par exemple, « Organisation ultra-radicale sunnite, le groupe Daech ou État islamique (EI) a connu un essor fulgurant en 2014, proclamant un « califat », en Irak et en Syrie. » (L'Express, 5/12/2019). Nous trouvons ici deux entités connues sous des noms différents.

Quant à la relation de ruption, elle marque les contradictions entre deux entités, X est incompatible avec Y. Typiquement, X et Y sont deux classes disjointes issues d'une même classe. Par exemple, « Des « musulmans modérés » affrontent des « musulmans intégristes » » (LeMonde, le 30/12/1995), elles appartiennent à la classe de « musulman », mais elles sont en opposition au sein de la classe.

Nous résumons l'approche de repérage par relations statiques dans le tableau 4.

Méthode	Avantage	Inconvénient
● Déterminer la relation entre l'entité repérée et l'entité repère	● Le référent de nomination repérée est expliqué, notamment par les descriptions catégorielles ● Les échanges discursifs négocient, les formes d'expression et les fonctions grammaticales sont montrées ● Le type de l'élaboration est	● Ce sont plutôt les entités relatives qui sont repérées. ● Cela nécessite d'identifier en contexte une entité repère

	désigné par la relation	

Table 4. L'approche de repérage par relations statiques

Ce modèle de repérage attache de l'importance aux signes contextuels, comme le note des auteures :

« Une nomination émergente est repérable dans les discours par une série d'indices contextuels, souvent cumulés, tels que guillemets, commentaires métadiscursifs, reformulations, enclosures ou définitions » (Jackiewicz, Pengam, 2020 : 2).

Ces indices contextuels accompagnent les nominations émergentes et pointent simultanément les différentes formes d'élaboration (intralocutive, interlocutive et interdiscursive) (Jackiewicz, Pengam, 2020). Pour nous, l'analyse de ces contextes spéciaux sera une piste de recherche inspirante.

Dans notre étude, nous pouvons traiter quelques indices de surface (par exemple les guillemets, les crochets, etc.) comme les critères élémentaires de détection de nomination émergente. Et nous nommons ainsi les contextes qui contiennent ces indices « les contextes spécifiants ». Même si l'on ne peut pas dire que toutes ces indices accompagnent toutes les nominations émergentes, dans certains cas particuliers, ils sont nécessaires. Par exemple, les nouvelles expressions par rapport à une nouveauté (ex. « immigration choisie ») ou les contextes d'explication dans un discours où les crochets indiquent une insertion de l'explication (ex. « Des solutions de mobilité douce (trottinettes, vélos électriques, dans toutes les communes de France) »). Ces indices seront assez pratiques pour une ample de récolte des données de façon automatique dans un premier temps.

Pour bien traiter ces indices de nomination émergente, cette méthodologie repose sur l'observation systématique du contexte de ces expressions, tout comme les approches TAL s'intéressant aux contextes riches en connaissance. Nous pouvons voir ensuite la notion de « contextes riches en connaissances » afin de tirer d'éventuelles pistes de recherche pour nos travaux.

2.2. Contextes riches en connaissances

Le contexte représente des informations importantes pour analyser les termes. Il est l'environnement linguistique d'un terme dont le sens et le fonctionnement sont éclairés par le contexte (de Bessé, 1991).

Vu ce contexte, la notion de Contextes Riches en Connaissances (CRC) est introduite. Ce sont des contextes qui jouent un rôle déterminant pour comprendre le terme et signaler leur fonctionnement linguistique. « Il s'agit de portions de textes qui contiennent i) des termes d'un domaine spécialisé et ii) des marqueurs explicitant des relations entre ces termes. » (Hmida, 2014 : 113). Trois relations sont illustrées entre les termes par les contextes : l'hyperonymie, la méronymie et la cause. Par exemple, « Les cendres sont les principaux produits volcaniques émis par les volcans explosifs de la ceinture de feu du Pacifique. » (Hmida, al, 2015). Ici, « cendre » et « produit volcanique » sont deux termes et qui « sont les principaux » exprime une relation d'hyperonymie reliant ces deux termes.

Éventuellement, les CRC sont identifiés grâce à des marqueurs de relations qui sont souvent représentés par les patrons de connaissances. « Un patron de connaissances est une expression régulière, formée de mots, de catégories grammaticales ou sémantiques et de symboles, visant à identifier des fragments de texte explicitant des formes lexicales et des catégories grammaticales. » (Hmida, 2014 : 113). Selon la définition, dans l'exemple dessus, « sont les principaux » est lié à un certain patron de connaissances.

Dans cette méthodologie, le terme, le contexte et la relation sont bien définis et forment un triplet dont chacun est indispensable. Chaque élément se trouve éclairé par les deux autres. Et il est ainsi possible de déterminer un terme en identifiant un patron dans lequel il intervient.

Pour nos travaux, cette méthode fonctionne également en analysant le contexte de la nomination émergente ciblée et ses marqueurs de relation au niveau morphosyntaxique afin de les identifier. Précisément, la nomination émergente serait le terme intéressant et les indices seraient les partons de connaissances évoquant la relation. Dès qu'on détermine le contexte et les indices, nous pouvons localiser la nomination émergente correspondante et saisir sa signification et son fonctionnement à partir de ses contextes. Par exemple, dans « J'achète responsable et donc moins de produits non recyclables comme le plastique », « comme » est l'indice d'explication et « plastique » est le contexte qui explique la nomination « produits non recyclables » par un exemple.

3. Les méthodes textométriques

A part les indices liés au contexte, en observant les nominations émergentes, nous pouvons observer qu'en général elles se forment par composition libre de mots existants,

bien que d'autres procédés de formations soient également possibles, tels que les dérivations, mots-valises, acronymes, etc. (comme dans *redéconfinement* ou *vaccino-car*).

Cette combinaison de mots existants, par la cooccurrence qu'elle engendre, peut nous fournir des indices utiles, exploitables au plan fréquentiel et textométrique. De ce point de vue, les nouvelles nominations se manifestent par des cooccurrences contiguës dont les fréquences évoluent dans le temps. Ce caractère émergent peut être mesuré par des comparaisons de corpus diachronique. Ainsi, nous pouvons à travers des calculs statistiques, sélectionner les combinaisons de mots dont la cooccurrence augmente de façon significative entre deux instants donnés.

3.1. Lexicoscope 2.0

Pour identifier ces écarts fréquentiels dans les cooccurrences, nous avons utilisé le Lexicoscope 2.0. Il s'agit d'une plate-forme accessible et gratuite proposée par Oliver Kraif, initialement développée en collaboration avec Sascha Diwersy dans le cadre des projets Emolex[3] et Phraseorom[4](Kraif et Diwersy, 2012, 2014). Comme l'indique le titre de son article « Le lexicoscope : un outil d'extraction des séquences phraséologiques basé sur des corpus arborés » (Kraif, 2016), à travers cet outil les corpus peuvent être explorés finement en fonction des relations de dépendances syntaxiques, qui capturent les cooccurrences les plus intéressantes. Cette plate-forme permet en outre à l'utilisateur de récupérer les contextes des expressions cibles avec les analyses statistiques détaillées.

Sur cette plate-forme, jusqu'à présent, vingt corpus sont disponibles pour les quatre langues : le français, l'anglais, l'allemand et le français du Moyen Âge. L'utilisateur peut sélectionner les corpus prédéfinis ou créer librement son corpus en important ses propres textes.

Concernant les analyses statistiques, nous pouvons explorer en détail chaque corpus selon trois dimensions : vocabulaire, collocations et métadonnées. Il est aussi possible de rechercher les occurrences d'une suite de mots ou d'une expression de recherche particulière (en utilisant la syntaxe TQL, pour *Tree Query Language*). Les résultats de cette requête seront présentés à travers cinq onglets : statistiques, concordances, cooccurrences, *wordsketch* et partionnement des contextes. Par ailleurs, si le corpus choisi contient des sous-corpus, toutes les analyses seront effectuées globalement, mais aussi

[3] http://phraseotext.u-grenoble3.fr/emolex/spip.php?article1&lang=fr
[4] https://phraseorom.univ-grenoble-alpes.fr/

partiellement pour chaque sous-corpus. Cette fonctionnalité répond au besoin de comparaison de corpus.

Cet outil se veut accessible aux utilisateurs linguistes sans formation préalable en informatique. Ainsi, deux formes de requêtes sont possibles : soit en tapant directement l'expression de surface que l'on recherche (p. ex. « vélo partagé »), soit en formulant une requête complexe via le formalisme du langage de requête TQL.

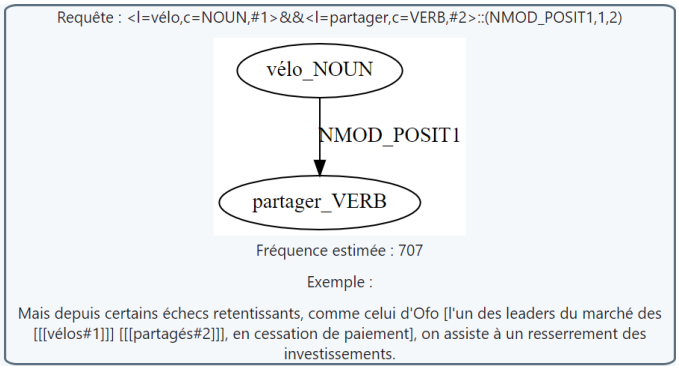


Figure 3. Requête TQL de "vélo partagé" :
<l=vélo,c=NOUN,#1>&&<l=partager,c=VERB,#2>::(NMOD_POSIT,1,2)

3.2. Méthode des ALR

Avant d'analyser statistiquement les nominations émergentes dans les discours en comparant des corpus, il convient de préciser ces expressions dans un premier temps. La méthode dite des ALR peut constituer un point d'entrée pour circonscrire le sujet et identifier les structures syntaxiques des expressions récurrentes spécifiques d'un sous-corpus.

La méthode des « Arbre lexico-syntaxiques récurrents » est inspirée par la notion de « cooccurrence syntaxique » (Evert, 2007) qui s'appuie sur la mesure statistique des cooccurrences de deux mots reliés par une relation de dépendance syntaxique, par exemple (*jouer* →OBJ→ *rôle*) (Kraif et Tutin, 2017).

Cette méthode s'appuie sur l'extraction d'une liste de cooccurrents significatifs ou lexicogrammes (Kraif et Diwersy 2012 ; 2014), aussi bien en partant d'un pivot (ou mot-pôle) que d'un pivot complexe ou sous-arbre (défini par une expression TQL).

La méthode des ALR, entièrement automatisée, fonctionne de la manière suivante (Kraif et Tutin, 2017 : 12) :

1. On part d'un pivot initial (mot simple ou arbre) ;

2. On en extrait le lexicogramme ;
3. Tous les collocatifs dépassant un certain seuil de cooccurrence et de mesure d'association (ici le loglike) sont rattachés au pivot, avec la relation concernée, pour former des arbres augmentés ;
4. On réitère l'étape 2 en reprenant ces nouveaux arbres comme pivot ; le processus est répété tant que l'on obtient, pour augmenter les arbres, des collocatifs dépassant les seuils de significativité, et que les arbres extraits n'ont pas dépassé une certaine longueur (paramétrable : dans la suite, la longueur sera fixée à 8 éléments).

Ainsi les arbres lexico-syntaxiques récurrents (ALR) sont obtenus. L'ALR de <proposer+dans+ce+article> est illustré par la figure 4 :

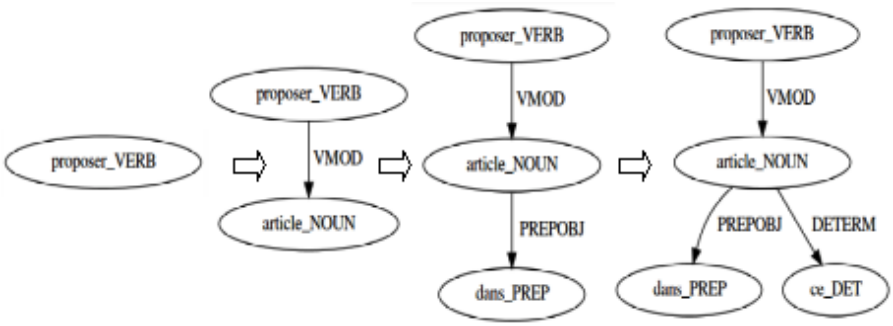


Figure 4. Extraction de l'ALR <proposer+dans+ce+article> (Kraif et Tutin, 2017)

Dans le cadre de nos travaux, vu que la plupart des nominations émergentes se réalisent dans des combinaisons de mots, nous pouvons lancer l'extraction des ALR autour d'une amorce lexicale. Avant tout, il faut préciser un pivot initial. Bien que l'amorce joue un rôle essentiellement thématique, nous proposons d'utiliser les nominations déjà trouvées pour en identifier de nouvelles. Ainsi, à partir d'une nomination comme *ville verte*, on pourra lancer les ALR à partir de *ville*, pour identifier d'éventuelle nomination en lien avec la thématique de la ville (*ville durable*, *ville comestible*, ...). Mais on pourra également partir de l'adjectif *vert*, qui réalise le glissement sémantique qui nous intéresse. On trouvera alors *croissance verte*, *tourisme vert*, etc. Avec les ALR, on pourra également trouver, le cas échéant, des nominations comportant plus que deux mots, comme *croissance réellement verte* ou *aménagement écologiquement durable*.

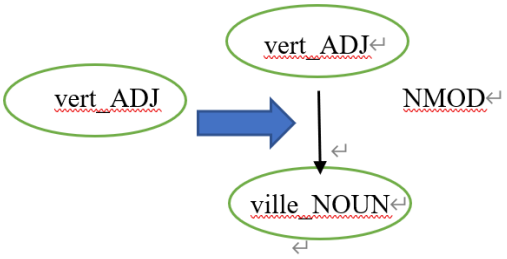


Figure 5. Extraction de l'ALR <ville verte>

Par ailleurs, en observant les combinaisons des mots qui nous intéressent, nous avons remarqué qu'elles se composent généralement d'une tête (un nom) et son modifieur (comme l'adjectif ou le complément qui fonctionne comme l'adjectif). Par conséquent, dans notre étude, et afin de restreindre le champ de nos explorations, nous nous concentrerons sur les patterns « N+ADJ » et « N+PREP+N ».

Nous proposons de préciser un pivot dans un premier temps et ensuite de chercher les expressions correspondantes aux patterns mentionnés. A travers l'expression en TQL, nous pouvons identifier facilement des expressions analogues, en précisant les relations de dépendances syntaxiques de type « nmod » ou « amod ». Par exemple, pour chercher des expressions analogues à la nomination émergente *ville verte*, on pourra obtenir l'ensemble des réalisations avec la requête suivante :

<l=vert,c=ADJ,#1>&&<c=NOUN,#2>::(NMOD_POSIT1,1,2)

4. Approches distributionnelles pour le glissement sémantique

De nombreux problèmes de TAL peuvent être reformulés comme des problèmes d'étiquetage de séquences. Les éléments qui composent le texte littéral (mots ou suites de caractères), pour faciliter les traitements en mathématique, sont remplacés par des représentations vectorielles dépendant de leur contexte. Cette conception intégrant des représentations distributionnelles peut être considérée comme une véritable révolution par rapport aux précédents modèles statistiques du langage, basés sur des séquences de n-grams. En effet, elle évite la croissance explosive des paramètres et prend en considération les liens sémantiques entre les mots, tout comme ce que les hypothèses distributionnelles proposent, les mots qui ont les contextes similaires sont proches sémantiquement (Harris, 1954) et un mot se caractérise par son contexte (Firth, 1957).

Elle répond au problème de la détection de nomination émergente, en comparant les contextes, et on peut supposer qu'elle permette de la sorte de saisir les évolutions sémantiques du mot cible.

4.1. Modèles distributionnels pour la nomination

Concernant la question de la variation de sens liée à la nomination, Manon Cassier (2020) a exploré des modèles distributionnels de repérage automatique en saisissant les glissements de sens.

Dans cette étude, les modèles destinés à repérer les candidats de nomination sont entraînés en utilisant les représentations de Word2Vec, avec l'architecture CBOW. Le principe est de substituer les mots par des vecteurs qui constituent une représentation condensée de l'ensemble de leurs contextes observés dans un corpus d'apprentissage. Cette représentation, appelée *word embedding*, est acquise en entrainant un réseau de neurone à prédire un mot en fonction de son contexte (ou réciproquement).

Afin de sélectionner les paramètres optimaux au cours de l'entraînement, deux modèles sont proposés et entraînés avec les paramètres par défaut de Word2Vec. Ils ne se différencient que par la taille de fenêtre (la taille du contexte considéré), 5 pour le premier modèle et 15 pour le second. L'adéquation de ces deux modèles est d'abord évaluée pour prédire chaque catégorie grammaticale en fonction du contexte. Selon les résultats, malgré le fait qu'un contexte plus restreint permette de mieux intégrer la syntaxe, il faut prendre en compte que la nomination est interprétée par un contexte du discours élargi, et non seulement par quelques mots autours du mot ciblé - un contexte plus large sera donc préférable au niveau sémantique. A travers le renouvellement de l'expérience, elle trouve que la fenêtre de 10 constitue un seuil à partir duquel les résultats ne varient plus.

Après avoir fixé le modèle, elle propose d'identifier les nominations émergentes en fonction des résultats de la prédiction :

> « Nous supposons que si le modèle propose dans sa liste de prédictions des mots très éloignés du mot à prédire, ce résultat implique que le mot en question est employé de manière inhabituelle (i.e. apparaît dans un contexte peu probable) et peut relever de la nomination » (Cassier, 2020:81)

Par conséquent elle suppose que si la prédiction de mot caché est très éloignée du mot à prédire, cela peut suggérer que le mot ciblé serait une nomination. Elle note cependant que ses premiers résultats ne sont guère exploitables du fait d'une mauvaise représentativité du corpus d'entrainement :

> « Néanmoins, ces résultats restent difficiles à exploiter puisque la majorité des prédictions comprennent des résultats très éparses, souvent très éloignés sans que l'on puisse déterminer s'ils sont la conséquence d'une nouvelle utilisation du mot ou d'une mauvaise représentativité des données dans le corpus d'entraînement. » (Ibid. :81)

Le tableau 5 est un résumé de l'approche distributionnelle pour la nomination.

Méthode	Avantage	Inconvénient
● Construire le corpus et annoter	● Un coût relativement faible en GPU	● Plusieurs facteurs restreignent le modèle

● Etablir un modèle et entraîner avec Word2Vec ● Sélectionner les paramètres optimaux par des expériences ● Prédire le mot pivot avec le modèle final et sélectionner les mauvaises prédictions	● Identifier automatiquement des candidats de nomination	entraîné, ex. propreté de données, corpus limité, surapprentissage, etc. ● Les résultats de prédiction ne sont pas fiables

Table 5. L'approche distributionnelle pour la nomination

Dans cette approche, deux idées nous semblent intéressantes. Tout d'abord, celle consistant à comparer les prédictions avec les mots à prédire. Comme une nomination émergente se caractérise aussi par la nouveauté, nous pouvons supposer qu'une mauvaise prédictibilité peut indiquer indirectement un glissement de sens, par lequel la nomination émergente serait détectée.

Un détail important dans ces expériences est la détermination de la taille de fenêtre du contexte. Un contexte étroit (il garde qu'une portion du contexte autour du mot lors de l'apprentissage du modèle) permet de mieux représenter les caractéristiques syntaxiques du mot ciblé, par contre un contexte plus grand est capable d'encoder plus en sémantique (rappelons qu'ici les contextes sont des sacs-de-mots). D'après ses tests, c'est la taille de fenêtre 10 qui équilibre au mieux la prédiction au plan syntaxique et la sémantique.

Inspiré par ces travaux, nous pourrons tenter l'expérience de la taille 10 de contexte lors de nos expérimentations de classification et de prédictibilité des nominations émergentes (cf. partie 3).

Ensuite, nous retenons l'idée d'utiliser des *word embeddings*, comme le font de nombreux modèles d'apprentissage. Les *embeddings* encodent tous les sens d'un mot en un vecteur en tenant compte des contextes à l'entour. Par rapport aux sens enregistrés dans le dictionnaire, il attache plus l'importance au sens actuel dans le discours en fonction des distributions observées en corpus. Si l'on veut établir un modèle de détection automatique de la nomination émergente à l'aide de l'apprentissage profond, la sélection de l'outil d'*embedding* est essentielle.

En effet, plusieurs modèles de représentation distributionnelle répondent à la question de la vectorisation, et il faut en faire une comparaison afin de déterminer une méthode préférable à notre modèle de détection.

4.2. Modèles du word embedding

Le *word embedding* est une méthode de représentation vectorielle de mots utilisée notamment dans le domaine TAL. Autrement dit, il permet de représenter chaque mot par un vecteur de nombres réels. S'appuyant sur l'hypothèse distributionnelle, cette nouvelle représentation offre la particularité que les mots apparaissant dans des contextes similaires possèdent des vecteurs aussi relativement proches dans l'espace vectoriel.

4.2.1. Embeddings statiques : compte vs prédiction

Pour produire un vecteur représentant un mot correspondant, deux méthodes statiques sont mises en pratique : la méthode à base de comptes et la méthode prédictive.

Basé sur les comptes, la méthode traditionnelle consiste à collecter d'abord les vecteurs de contexte enregistrant les fréquences d'occurrence des mots constituant le contexte, chaque dimension du vecteur étant attachée à un mot du vocabulaire. Vient ensuite une étape repondération et de redimensionnement de ces vecteurs en fonction de divers critères. La similarité sémantique peut ensuite être caractérisée par une mesure quantitative telle que le cosinus.

La méthode prédictive a bouleversé radicalement cette vision : les composantes des vecteurs sont directement définies comme des poids attachés à une couche de neurones permettant de prédire les mots qui ont tendance à apparaître en fonction des contextes.

Or les méthodes prédictives surclassent l'approche traditionnelle basée sur les comptes. En effet, Marco Baroni et al. (2014) ont montré que les modèles prédictifs obtiennent de meilleures performances que les méthodes traditionnelles à base de comptes en comparaissant systématiquement ces deux modèles (le modèle de compte se base sur l'outil de DISSECT[5] , et l'autre est entraîné avec le Word2Vec) dans une série de benchmarks variés.

[5] http://clic.cimec.unitn.it/composes/toolkit/

4.2.2. Word2Vec

L'outil Word2Vec [6] (Mikolov, et al., 2013) constitue une des premières implémentations de la méthode prédictive, et il est très répandu comme on l'a vu dans les travaux précédemment mentionnés.

En 2013, une équipe de recherche de Google, sous la direction de Tomas Mikolov, a présenté sous le nom de Word2Vec un groupe de modèles utilisés pour le *word embedding*. Son introduction a constitué une vraie révolution pour les modèles de langage. Deux modèles d'entraînement sont développés dans cet outil : CBOW et Skip-Gram.

Le modèle CBOW (Continuous Bag-of-Words Model), comme son nom l'indique, est basé sur un vecteur représentant le contexte en entrée sous forme de sac de mots. Ce vecteur est multiplié par une matrice pour obtenir un vecteur de plongement continu. L'objectif d'apprentissage consiste à masquer un mot et puis à le prédire en fonction de son contexte. Au cours de l'entraînement, le système apprend la représentation vectorielle visée sous la forme de l'ensemble des poids associés à un mot en sortie.

À l'inverse, il est également possible d'apprendre une représentation vectorielle à partir de la prédiction du contexte en fonction du mot ciblé. C'est le modèle Skip-gram qui permet de prédire les probabilités des mots du contexte en entrant un mot ciblé.

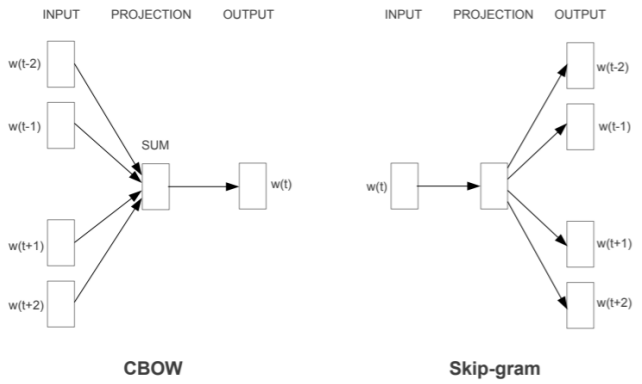


Figure 6. Modèles CBOW et Skip-gram (Mikolov, V. Le, Sutskever, 2013)

Même si les optimisations de Word2Vec rendent l'entraînement efficace et assez peu coûteux, le principal problème de cette représentation est son caractère statique (un mot égal un vecteur) qui n'intègre pas les phénomènes de polysémie lié au contexte. En effet, les mots polysémiques sont fréquents dans le langage naturel et ils permettent à la fois la flexibilité et l'efficacité dans la communication. Par exemple, le mot *mémoire* est ambigu

[6] https://code.google.com/p/word2vec/

et porte différents sens en fonction du genre : le féminin signifie « l'aptitude à se souvenir », ou encore « un dispositif informatique pour enregistrer des données » et le masculin peut correspondre à un « travail de recherche rédigé en vue de l'obtention d'un diplôme ». Un *embedding* de Word2Vec ne peut pas faire la distinction entre ces différents sens lors de l'encodage du mot *mémoire*, et son vecteur enregistrera pêle-mêle tous les contextes de cette forme graphique, quelle qu'en soit l'interprétation sémantique ou grammaticale. Cela va causer une forme de confusion, les contextes différents étant encodés dans un même espace vectoriel : il s'agit d'un défaut important des *embeddings* dits « statiques ».

4.2.3. Embeddings contextuels

Pour pallier ces défauts, de nouvelles méthodes d'*embeddings* contextuels, visant à représenter les mots en contexte, ont été développés. Le principe consiste à ajuster dynamiquement le *word embedding* d'après son contexte d'occurrence.

4.2.3.1. ELMO

ELMO (Embedding from Language Models) est un des premiers modèles qui répond à ce besoin (E. Peters, et al., 2018). L'idée essentielle d'ELMO est d'abord d'apprendre la représentation vectorielle de chaque mot en entrainant un modèle de langage bidirectionnel, combinant deux modèles de langage : le premier cherchant à prédire une forme en fonction de son contexte droit, le second en fonction de son contexte gauche.

Deux étapes principales composent le processus : le pré-entraînement par le modèle de langage bidirectionnel, et l'extrait de l'*embedding* du mot cible constitué par une combinaison des poids des couches intermédiaires et de la couche de sortie.

Le réseau de pré-entraînement est constitué par plusieurs couches de Bi-LSTM (Bi-directional long short term memory), qui enregistrent différents types d'information : la première couche permet d'encoder plus d'informations concernant la syntaxe et les POS ; les couches supérieures proches de la sortie contribuent à encoder des informations sémantiques liées au contexte, utiles par exemple pour la désambiguïsation.

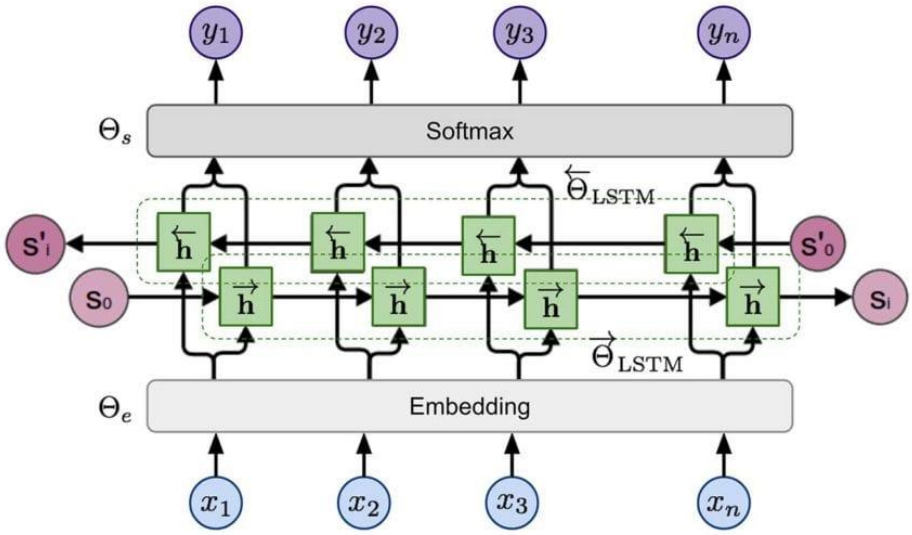


Figure 7. ELMO : deux couches de Bi-LSTM (cf. https://www.topbots.com/generalized-language-models-cove-elmo/, 2019)

Les vecteurs contextuels d'ELMO ont permis d'améliorer les résultats pour de nombreuses tâches réputées ardues en TAL, allant de l'analyse de sentiment aux systèmes de question-réponse, en passant par le *textual entailment* (implication textuelle). Néanmoins, au fur et à mesure de l'avance des réseaux de neurones, en 2018, Google a introduit une architecture radicalement différente, où les effets contextuels ne sont plus représentés par une architecture récurrente, mais par la généralisation des modèles d'attention. Il s'agit du modèle *Transformer* décrit dans l'article *Attention Is All You Need* (Vaswani, et al., 2017). Il est aujourd'hui assez largement reconnu que la capacité des *transformers* d'extraire des traits abstraits (sémantiques, grammaticaux, ...) en fonction du contexte est meilleure que chez les LSTM. De ce point de vue, le modèle BERT proposé aussi par Google est réputé plus performant pour de nombreuses tâches en TAL.

4.2.3.2. BERT

Le modèle BERT (Bidirectional Encoder Representation from Transformers) proposé par Google en 2018 est un modèle pré-entrainé basé sur l'architecture *transformer* (Devlin, al, 2018). Il permet deux types d'utilisation : l'utilisation des vecteurs pré-entraînés en fonction de deux tâches (Masked LM, le modèle de langage permettant de prédire des mots masqués et Next Sentence Prediction ou NSP, un classifieur permettant d'indiquer si la deuxième phrase en entrée est bien consécutive de la première) et le *fine-tuning* pour différentes tâches de classification appliquées en sortie. Son architecture se base sur le

modèle *transformer*, avec six couches comportant des modèles d'attention multi-têtes permettant d'encoder différents traits contextuels pour chaque mot de l'entrée.

Concernant les *embeddings* de l'entrée, ils sont la somme de 3 *embeddings* : l'*embedding* de token (des mots ou fragments de mots obtenus par *byte-pair encoding*), l'*embedding* de segment et l'*embedding* de position. L'*embedding* de token consiste à transformer chaque mot en un vecteur à dimension fixe (il s'agit donc d'un embedding statique de type Word2Vec). Dans BERT, chaque mot sera une représentation vectorielle à 768 dimensions. Ensuite, l'*embedding* de segment est utilisé pour distinguer deux phrases, au cours de la tâche de NSP. Chaque token de phrase première est représenté par 0 et 1 pour la phrase suivante. Enfin, à la différence du modèle *transformer* standard, l'*embedding* de position de BERT est obtenu par entraînement, ce qui lui permet d'analyser 512 tokens au maximum.

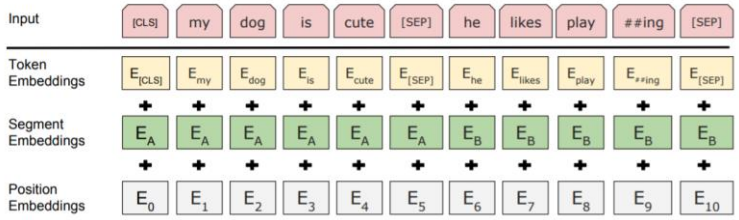


Figure 2: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

Figure 8. Trois embeddings (Devlin, Chang, Lee, Toutanova, 2018)

Le pré-entraînement est réalisé autour de deux tâches. La première correspond au modèle de *Mask Language*. Au cours de l'entrainement de ce modèle, 15% des mots dans chaque séquence du texte sont marqués par « Mask », et la machine doit prédire le mot en fonction des contextes gauche et droit. L'autre modèle consiste à prédire la phrase suivante : lors de l'entrainement, on présente 50% de paires de phrases adjacentes et 50% de paires de phrases sélectionnées aléatoirement. L'objectif consiste à effectuer une classification binaire, pour prédire si les deux phrases sont contiguës ou non.

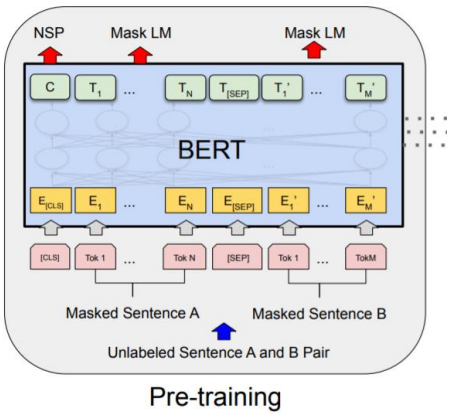


Figure 9. Modèles de pré-entraînement (Devlin, Chang, Lee, Toutanova, 2018)

Ces deux pré-entraînements permettent de saisir les représentations au niveau des mots et des phrases. Le modèle pré-entrainé profite des avantages du *transformer*, car il intègre les informations globales du contexte. En outre, pour des tâches populaires en TAL telles que l'analyse des sentiments, les systèmes de question-réponse comme SQuAD, la reconnaissance d'entités nommées, etc., il existe des modèles correspondants déjà pré-entrainés pour l'anglais. De ce fait, le modèle BERT a représenté une petite révolution dans le domaine du TAL, en obtenant des scores excellents pour presque toutes les tâches dans différentes campagnes.

Après avoir vu les outils intéressants en lien avec les *embeddings*, nous voyons se dessiner des pistes de recherche pour saisir les glissements de sens propre aux nominations émergentes : le sens d'un mot étant en partie déterminé par son contexte, un contexte inhabituel pourrait se traduire par une difficulté à prédire le mot dans son contexte. Nous notons que, à l'aide du modèle MLM de BERT, nous pouvons facilement tester la prédictibilité d'un composant d'une nomination émergente potentielle en comparant la liste des formes prédites avec le mot masqué.

En 2019, FlauBERT[7] (French Language Understanding via Bidirectional Encoder Representations from Transformers) est conçu dans l'objectif de fournir des modèles entraînés avec un immense corpus du français (Le et al., 2019). Au total 71 Go de données sont recueillies et elles se composent de 24 sous-corpus provenant de différentes sources et couvrant divers sujets. On a principalement trois sources : (1) les données unilingues pour le français fournies dans les tâches partagées de WMT19[8] (Li et al., 2019, 4 sous-

[7] https://huggingface.co/transformers/model_doc/flaubert.html
[8] European Parliament Proceedings Parallel Corpus 1996-2011. http://www.statmt.org/europarl/

corpus) ; (2) les corpus de textes en français offerts dans la collection OPUS[9] (Tiedemann, 2012, 8 sous-corpus) ; (3) des ensembles de données disponibles dans les projets Wikimedia (Meta, 2019, 8 sous-corpus).

Pour tester nos hypothèses, BERT, sous la forme de son modèle pré-entrainé pour le français FlauBERT, semble être le meilleur choix, non seulement pour les *embeddings* contextuels qu'il permet d'extraire, mais aussi par ses possibilités de *fine-tuning* pour s'entrainer à une tâche telle que la détection de nomination émergente. Autrement dit, nous pourront l'entrainer à déterminer si une phrase contient ou non une nomination émergente. Nous pouvons ainsi élaborer un classifieur en nous appuyant sur un modèle existant et pré-entraîné.

En guise de conclusion, nous allons résumer les principales pistes de recherche et formuler nos hypothèses de travail.

Deux pistes principales seront suivies dans nos travaux :

A/ Identifier les contextes caractérisant les nominations émergentes à l'aide d'un classifieur :

- Pour l'étude, sélectionner manuellement une liste de nominations émergentes.
- Extraire les contextes de ces nominations et annoter manuellement les contextes dits "spécifiants", c'est-à-dire comportant divers indices contextuels d'explicitation, d'ajustement ou de démarcation - des marques permettant à l'auteur de signaler au lecteur qu'il emploie une forme dans un sens inhabituel.
- Entraîner un classifieur qui permet d'identifier les contextes spécifiants.
- Calculer le taux de contextes spécifiants entourant les nominations cibles.

L'idée est donc ici de chercher à identifier des marques contextuelles susceptibles d'accompagner une partie des occurrences des nominations émergentes (tout comme l' « acte de baptême » d'une nomination, mentionné par Siblot 2001).

[9] OPUS - an open source parallel corpus. https://opus.nlpl.eu/

B/ Identifier les glissements sémantiques à l'œuvre dans les nominations émergentes, à l'aide du modèle MLM :

- Identifier un éventuel changement sémantique avec des *embeddings* contextuels : en mesurant pour un contexte donné, la prédictibilité du mot cible.
- On calculera la moyenne du rang du mot à prédire dans la liste des prédictions effectuées.

Nos expérimentations suivantes se basent sur 4 hypothèses :

1. Les nominations émergentes sont souvent caractérisées par des indices contextuels précis, indiquant qu'elles ne sont pas employées dans le sens usuel. Ces indices sont liés au besoin de définir, d'expliciter, d'opposer, etc. Nous formulons l'hypothèse que les nominations émergentes sont caractérisées par un taux élevé de contextes spécifiants - cette hypothèse devra être vérifiée en comparant ce taux avec des expressions "normales" (non émergentes) de même nature morphosyntaxique.
2. Ces indices contextuels sont de nature variable, au plan formel : explicitation entre parenthèses, crochets, ou après « : », utilisation de guillemets pour indiquer la citation ou la prise en charge énonciative, etc.
3. Un classifieur neuronal s'appuyant sur des *embeddings* contextuels est peut-être capable d'apprendre à identifier ce qui caractérise ces contextes, en combinant des indices de surface avec des indices sémantiques.
4. Les glissements de sens propres aux nominations émergentes peuvent être caractérisés par une moindre prédictibilité dans un modèle comme celui de BERT.

Dans la partie suivante, nous détaillons nos expérimentations basées sur l'apprentissage profond. Elle se divise en quatre chapitres. Après la présentation du corpus et des jeux de données, nous expliquons la construction et l'entraînement de notre classifieur, puis les expériences effectuées avec le modèle MLM de FlauBERT.

Partie 3

-

Expérimentation basée sur l'apprentissage profond

Chapitre 3. Construction du corpus

Nous avons cherché à construire un corpus susceptible de nous offrir de nombreux exemples de nomination émergente, en contexte. Ce corpus doit nous permettre une observation manuelle des phénomènes qui nous intéressent, mais aussi une extraction d'un jeu de données pour l'entrainement d'un classifieur.

Avant de commencer la collection, il faut définir en détail ce corpus écrit. Évidemment, une taille assez grande est préférable aux traitements statistiques, ainsi que nous essayons à recueillir complètement dans la mesure du possible. Vu que notre étude se réalise en français, le corpus ne couvrira qu'un seul langage, le français.

Eu égard aux thématiques émergentes, nous proposons de recueillir des articles de presse, des rapports et des débats citoyens de ces dernières années. A travers différents genres textuels (article de presse, contributions individuelles à un débat, rapports et notes de synthèse issus d'organisations gouvernementales et d'associations) nous cherchons à rassembler des thématiques variées et proches des préoccupations de la population - les sujets qui alimentent ce que l'on nomme le « débat public ».

Le mouvement des Gilets jaunes a ainsi permis d'aboutir à des productions discursives riches et intéressantes. A la suite de ces manifestations nationales, le Grand Débat national organisé par le gouvernement et le Vrai Débat organisé en contrepoint par des collectifs citoyens nous offrent un vaste corpus de contributions[10]. A la suite de ces initiatives, et pour répondre aux préoccupations concernant le changement climatique, de plus en plus central dans le débat public, une assemblée de citoyens français a été constituée pour former « La Convention citoyenne pour le climat », et aboutir à une série de propositions que le Président de la république s'est engagé à mettre en œuvre dans son ensemble. Pour préparer les travaux de cette convention, une série de rapports et d'études a été fournie en amont par des acteurs associatifs et institutionnels[11].

Enfin, la pandémie de Covid qui s'est déclarée en 2020 a également suscité un grand nombre de débats et de nominations émergentes concernant plusieurs aspects : économie, tourisme, politique, etc. En particulier, les mesures de confinement, par leur caractère

[10] http://phraseotext.univ-grenoble-alpes.fr/lexicoscope_2.0/analytics
[11] https://www.conventioncitoyennepourleclimat.fr/

exceptionnel, ont suscité de très riches discussions. Nous chercherons à suivre l'évolution de ces débats à travers des articles du quotidien *Le Monde*.

Concernant l'aspect diachronique du corpus, nous avons construit le corpus en composant des sous-corpus qui sont divisés non seulement en fonction des sources, mais aussi des périodes. Cette démarche permet de faciliter les comparaisons afin d'observer l'évolution de la nomination au fil du temps.

Le corpus est divisé en sous-corpus de la manière suivante :

- 2018-2019 : Le Monde 2018-2019 : des articles du Monde allant du lancement du Grand débat national au lancement de le Convention citoyenne sur le climat.
- 2019 : Le Grand Débat : les contributions du Grand Débat de 2019.
- 2019 : Le Vrai Débat : les contributions du Vrai Débat.
- 2019 : Convention citoyenne sur le climat : l'ensemble des ressources documentaires mis à la disposition de la Convention.
- 2020 : Le Monde 2020 : des articles du Monde allant du début du confinement de mars à la remise des travaux de la Convention en juin 2020.

1. Récolte des données

Ayant ainsi précisé la définition du corpus, il a ensuite fallu récolter les données. Comme les corpus de Grand Débat et Vrai Débat étaient déjà construits et disponibles sur la plate-forme Lexicoscope 2.0, il ne restait qu'à collecter les articles du Monde et les documents de la Convention.

Toutes les ressources documentaires pour la Convention climatique, soit 11 fichiers PDF, se trouvent sur le site officiel ; et concernant les articles du Monde, nous pouvons les consulter en version numérique sur le bouquet Europresse[12] qui est une base de données d'informations accessible aux bibliothèques universitaires abonnées pour la recherche et l'enseignement. Cette plate-forme facilite le téléchargement grâce à son formulaire de recherche permettant de définir des paramètres comme la source, la période, d'éventuels mots clés, etc.. Tous les résultats sont présentés de façon centralisée sous la forme d'une liste d'hyperliens.

[12] https://nouveau.europresse.com

Ça aurait été un travail immense et irréalisable que de télécharger et d'enregistrer les articles au bon format à la main. Pour automatiser cette tâche, la technique de *web crawling* grâce à la librairie Selenium a répondu parfaitement à nos besoins.

1.1. Robot d'indexation (web crawler)

Un robot d'indexation est un robot Internet qui permet d'explorer automatiquement et systématiquement le Web. On peut le considérer comme un outil qui fonctionne en simulant un comportement humain de navigation sur des sites web divers, permettant par exemple de cliquer sur des boutons, de consulter des données ou d'enregistrer les informations qu'il a rencontrées.

Le robot d'indexation peut vérifier les hyperliens et le code HTML. Il est employé également pour le *web scraping*, consistant à aspirer des données afin de construire un corpus.

1.2. Selenium

Selenium[13] est un outil gratuit (open source) dédié à l'automatisation de tests (pour le développement web) qui s'exécute directement sur un navigateur, tout comme un utilisateur réel le ferait fonctionner. Les navigateurs pris en charge incluent IE (7, 8, 9, 10, 11), Mozilla Firefox, Safari, Google Chrome, Opera, Edge, etc. et il existe des librairies pour plusieurs langages de programmation tels que Java, C#, Python, etc.

L'outil Selenium n'est pas un outil unique, mais une suite de logiciels et chaque élément répond à des besoins différents. Voici la liste des outils : Selenium IDE, permettant un enregistrement et une lecture simples des interactions avec le navigateur ; Selenium Grid, facilitant des tests sur plusieurs machines et la gestion de plusieurs environnements à partir d'un point central; et WebDriver qui s'emploie dans nos travaux effectuant des tests d'automatisation par un *driver* (ou pilote) du navigateur. Initialement, Selenium est conçu pour effectuer des tests : qu'il s'agisse de tester la compatibilité avec les navigateurs différents ou de mettre en œuvre des tests de régression pour identifier des bogues, vérifier les fonctions d'une application et la conformité avec les exigences des utilisateurs. Par contre, dans notre optique de recherche, il s'applique couramment pour naviguer automatiquement sur le Web et faire du *web scraping*.

[13] https://www.selenium.dev/fr/

Concernant l'utilisation concrète de Selenium, nous l'avons réalisé sur le navigateur Google Chrome en exécutant des scripts en python[14]. Voici les étapes principales effectuées pour automatiser la navigation sur Europresse :

1. Accéder au site de l'Europresse à travers la bibliothèque universitaire et se connecter avec notre compte étudiant
2. Choisir la fonctionnalité « Recherche avancée » : indiquer « Le Monde » dans le champ 'nom de la source' et préciser la période en choisissant les dates de début et de fin, et puis cliquer « rechercher »
3. Trier les résultats par ordre chronologique du plus ancien au plus récent et afficher tous les articles en actionnant la barre de défilement ;
4. Ouvrir un article en cliquant sur le titre, sélectionner et copier le titre, la date, l'auteur et le contenu du texte ;
5. Créer un fichier de XML, et enregistrer toutes les informations au format XML-TEI après une vérification de l'encodage XML (cf. figure 10) ;
6. Retourner sur la page précédente et répéter les étapes 4 et 5.

Grâce au langage HTML, les éléments d'un document sont bien formés avec une structure stable et ils s'identifient par des balises. Par conséquent, lors de l'exploration automatique d'Europresse, nous pouvons situer les informations qui nous intéressent avec des requêtes *xpath* en précisant les descriptions de leurs balises et de leur contexte. Par ailleurs, il faut faire attention ici à l'enregistrement. Etant donné certains caractères spéciaux (comme `&` qui doit être remplacés par `&`) liés à l'encodage XML, il est nécessaire de les repérer et de les remplacer en avance par les expressions correspondantes afin d'éviter des erreurs d'encodage XML.

Au total, 39 910 articles du Monde sont téléchargés automatiquement à partir d'Europresse. En raison de la durée limitée des sessions de connexion, ce test d'automatisation s'arrête après environs 4 heures de connexion. Finalement, le travail d'enregistrement nous a pris environ deux semaines, du fait de ces interruptions dans la connexion et de la nécessité de relancer le script manuellement à chaque fois.

Enfin, même si le téléchargement des 11 documents de la convention climatique a été trivial, comme il s'agit de fichiers PDF, il nous reste à les convertir en XML-TEI. Pour

[14] Le script est accessible ici : https://github.com/yumengding/nomination-mergente

automatiser cette opération, l'outil PDFMiner[15] a constitué une bonne solution : c'est un outil pratique pour extraire rapidement les informations d'un fichier PDF en analysant sa structure. Dans nos travaux, nous l'employons dans l'environnement Python pour extraire le contenu de chaque PDF et l'enregistrer dans un nouveau fichier XML[16].

2. Corpus final

Il ne restait plus qu'à regrouper, classifier en fonction de la source et la période, et à structurer les données collectées. Nous pouvons résumer ainsi globalement la composition de notre corpus, que nous avons baptisé « Le Monde d'Après » :

Corpus final	Sous-corpus
Le Monde d'Après	Le Monde 2018-2019
	Le Monde 2020
	Convention citoyenne sur le climat
	Le Grand Débat
	Le Vrai Débat

Table 6. Composants du corpus

Nom	**Le Monde d'Après**	Le Monde 2018-2019	Le Monde 2020	Convention citoyenne sur le climat	Le Grand Débat	Le Vrai Débat
Période	**12/2018 – 07/2020**	12/2018 – 10/2019	03/2020– 07/2020	2019	2019	2019
Langue	**Français**	Français	Français	Français	Français	Français
Nombre de documents	**100 473**	27 841	9 350	11	19 996	69 837
Nombre de phrases	**1 685 096**	784 483	330 925	11 047	471 700	86 941
Nombre de tokens	**35 715 532**	18 368 072	7 646 987	221 588	7 310 794	6 504 273

Table 7. Chiffres du corpus

Notons que le corpus intégral du Grand Débat étant trop volumineux (environ 180 millions de mots), nous avons décidé de n'en prendre qu'un échantillon d'environ 7 millions de mots, pour des raisons d'équilibrage entre les différents sous-corpus : de la sorte nous avons environ 10 millions de mots pour les corpus de débats, 25 millions de mots pour les corpus de presse - et le corpus de la Convention ne constitue qu'un petit sous-corpus de 221 000 mots peu comparable aux autres en termes de fréquence.

[15] https://pypi.org/project/pdfminer/
[16] Le script est disponible ici : https://github.com/yumengding/nomination-mergente

Chaque document du corpus est enregistré en format XML, le nom du fichier étant défini par son titre et sa date de publication, pour distinguer les articles ayant le même titre. Toutes les informations se structurent en forme TEI qui se manifeste principalement par deux parties : un en-tête et le corps du document. Généralement, six éléments essentiels sont pris en compte au niveau des métadonnées : titre, auteur, éditeur, date, source et texte.



Figure 10. Structure d'un fichier XML

3. Annotation du corpus

Un corpus ne se compose pas seulement de textes et de métadonnées, mais aussi d'annotations diverses. D'un point de vue général, l'annotation est le produit d'un acte interprétatif ou d'un calcul, et elle consiste à associer des informations telles qu'une catégorie morphosyntaxique, une relation syntaxique, des propriétés, etc., à certaines parties du texte (paragraphe, phrase, mot, etc.), de façon manuelle ou avec des outils automatiques.

En général, l'annotation d'un corpus écrit se concentre sur des aspects variés : part-of-speech (POS), lemmatisation, dépendances syntaxiques, traits sémantiques, entités nommées, coréférences, etc. Dans nos travaux, du fait de l'utilisation du Lexicoscope, nous nous intéressons plus particulièrement à l'étiquetage et aux relations de dépendance syntaxique.

Avec l'objectif de disposer d'un corpus annoté en syntaxe de taille importante avec une qualité suffisante, une annotation à la main n'est pas envisageable, et nous recourrons donc à des outils automatiques. Finalement, dans l'environnement de Python, nous avons choisi le package Stanza, qui par la suite a été intégré comme analyseur par défaut dans le Lexicoscope.

3.1. Stanza

Le package Stanza[17] est construit par le groupe de recherche NLP de Stanford (Qi et al., 2020), il fournit des outils pratiques et efficaces qui supportent 66 langues pour l'analyse linguistique. Employé à travers un pipeline, il est capable de transformer le texte en une série de phrases et de tokens, et d'y associer des lemmes, des informations morphologiques et des étiquettes POS, puis d'analyser des relations de dépendances et de reconnaître des entités nommées. Par ailleurs, fondé sur des réseaux de neurones, il permet aussi aux utilisateurs d'entraîner des modèles avec leurs propres données annotées.

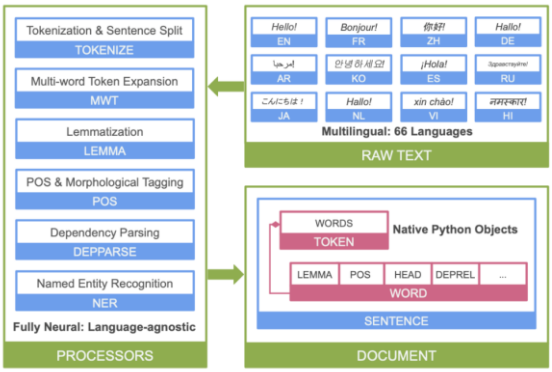


Figure 11. Pipeline de Stanza (Qi et al., 2020)

Pour l'annotation de notre corpus, c'est la fonctionnalité de l'analyse syntaxique qui est centrale. Ce module va extraire un arbre syntaxique pour chaque phrase d'entrée, par lequel les relations de dépendance entre les mots sont clairement explicitées. Et comme cette analyse s'appuie sur le formalisme de « Universal Dependencies », nous allons expliquer brièvement ce projet.

[17] https://stanfordnlp.github.io/stanza/

3.2. Universal Dependencies

Le projet « Universal Dependencies »[18] désigné fréquemment par le sigle « UD », est un projet de collaboration internationale pour créer des *treebanks* standardisées, accessibles et disponibles pour de nombreuses langues. L'annotation UD précise notamment un jeu d'étiquettes de relations et de parties du discours suffisamment générique et abstrait pour être commun à de nombreuses langues. Actuellement, des *treebanks* en plus de 70 langues sont accessibles avec ce formalisme.

Dans ce projet, l'analyse syntaxique s'effectue en précisant les relations de dépendance entre des mots par ailleurs étiquetés avec une catégorie grammaticale, et chaque relation se caractérise par sa fonction syntaxique et est marquée par un label unique. Le dictionnaire des labels de catégorie et de relation est disponible sur le site officiel, accompagné de bon nombre d'exemples concrets, pris dans des langues variées afin d'illustrer les explications.

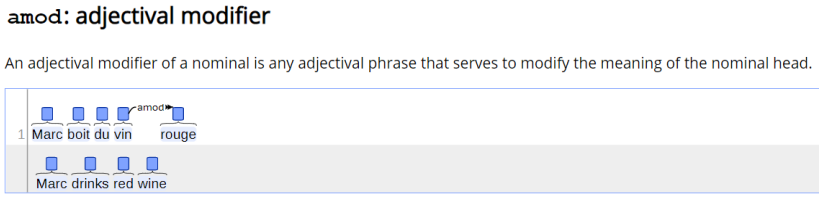


Figure 12. Exemple UD (https://universaldependencies.org/fr/dep/amod.html)

3.3. Evaluation des modèles

L'annotation automatique est plus simple et rapide que l'annotation manuelle, et plusieurs outils informatiques sont disponibles, comme Spacy, UDPipe, CoreNLP ou Stanza. Le choix de Stanza s'explique essentiellement par sa simplicité d'utilisation et ses performances excellentes en comparaison avec d'autres outils populaires en TAL.

Dans l'article de Peng Qi et al. (2020), les auteurs détaillent l'architecture et les fonctionnalités de Stanza, et ils comparent globalement ses caractéristiques avec CoreNLP, FLAIR, spaCy et UDPipe sous 6 aspects. Nous notons qu'il est le modèle qui supporte le plus grand nombre de langues et se fonde totalement sur des réseaux de neurones.

[18] https://universaldependencies.org

System	# Human Languages	Programming Language	Raw Text Processing	Fully Neural	Pretrained Models	State-of-the-art Performance
CoreNLP	6	Java	✓		✓	
FLAIR	12	Python		✓	✓	✓
spaCy	10	Python	✓		✓	
UDPipe	61	C++	✓		✓	✓
Stanza	66	Python	✓	✓	✓	✓

Table 1: Feature comparisons of Stanza against other popular natural language processing toolkits.

Figure 13. Comparaison les spécialités avec les outils populaires (Qi et al., 2003)

Concernant l'évaluation, il est entraîné et évalué sur 112 jeux de données au total pour 9 tâches, en comparaison avec UDPipe et spaCy. Les évaluations concernent 5 langues à grande diffusion (arabe, chinois, anglais, français et espagnol). En analysant les résultats, nous observons que Stanza obtient le meilleur score dans presque tous les tests. Particulièrement pour les tâches en français, on observe une amélioration nette par rapport aux performances des autres outils.

Treebank	System	Tokens	Sents.	Words	UPOS	XPOS	UFeats	Lemmas	UAS	LAS
Overall (100 treebanks)	Stanza	99.09	86.05	98.63	92.49	91.80	89.93	92.78	80.45	75.68
Arabic-PADT	Stanza	**99.98**	80.43	**97.88**	**94.89**	**91.75**	**91.86**	**93.27**	**83.27**	**79.33**
	UDPipe	**99.98**	**82.09**	94.58	90.36	84.00	84.16	88.46	72.67	68.14
Chinese-GSD	Stanza	**92.83**	98.80	**92.83**	**89.12**	**88.93**	**92.11**	**92.83**	**72.88**	**69.82**
	UDPipe	90.27	**99.10**	90.27	84.13	84.04	89.05	90.26	61.60	57.81
English-EWT	Stanza	**99.01**	**81.13**	**99.01**	**95.40**	**95.12**	**96.11**	**97.21**	**86.22**	**83.59**
	UDPipe	98.90	77.40	98.90	93.26	92.75	94.23	95.45	80.22	77.03
	spaCy	97.30	61.19	97.30	86.72	90.83	–	87.05	–	–
French-GSD	Stanza	**99.68**	**94.92**	**99.48**	**97.30**	–	**96.72**	**97.64**	**91.38**	**89.05**
	UDPipe	**99.68**	93.59	98.81	95.85	–	95.55	96.61	87.14	84.26
	spaCy	98.34	77.30	94.15	86.82	–	–	87.29	67.46	60.60
Spanish-AnCora	Stanza	**99.98**	**99.07**	**99.98**	**98.78**	**98.67**	**98.59**	**99.19**	**92.21**	**90.01**
	UDPipe	99.97	98.32	99.95	98.32	98.13	98.13	98.48	88.22	85.10
	spaCy	99.47	97.59	98.95	94.04	–	–	79.63	86.63	84.13

Figure 14. Evaluation des modèles (Qi et al., 2020)

Chapitre 4. Jeux de données

Dans la perspective de conduire des expériences de classification et de tester le modèle MLM, il nous a d'abord fallu construire les jeux de données correspondants.

1. Jeux de données pour les classifieurs

Pour entraîner le classifieur qui se base sur un modèle d'apprentissage, notre jeu de données doit contenir 3 éléments essentiels : des candidats de nomination émergente, des contextes tirés du corpus et l'annotation indiquant s'ils sont spécifiants ou non. Ces annotations humaines serviront de références pour construire nos modèles lors de l'apprentissage, elles doivent donc être établies de façon rigoureuse et fidèle.

Comme nous l'avons expliqué dans le chapitre 2, nous avons sélectionné des nominations émergentes potentielles en partant de certains pivots lexicaux, et nous nous sommes limités aux patterns « N+ADJ » et « N+PREP+N » en syntaxe. Le principe de la construction consiste, dans un premier temps à sélectionner des pivots et des expressions intéressantes et à recueillir des phrases contenant éventuellement des contextes spécifiants, puis dans un deuxième temps, d'annoter manuellement le caractère spécifiant ou non spécifiant des contextes retenus.

Ce processus a été exécuté en 5 étapes :

1. Extraire une série de collocations intéressantes, autour de deux thématiques : l'écologie, et le covid.
2. En fonction des collocations trouvées, sélectionner les noms ou les adjectifs qui formes des pivots intéressants, ainsi que les expressions associées à partir des patterns retenus.
3. Extraire un corpus de contextes autour des expressions retenues.
4. Effectuer une annotation double du caractère spécifiant (ou non) des contextes, et calculer l'accord Inter-Annotateur (AIA) pour valider l'annotation au plan intersubjectif.
5. Sélectionner les annotations consensuelles pour constituer notre jeu de données.

Comme nous travaillons sur les thèmes du climat et du covid, nous avons extrait une série de collocations intéressantes sémantiquement et correspondant aux patterns retenus

concernant ces deux domaines (p. ex. *ville verte*, *ville durable*, *monde d'après*, etc.) sur le Lexicoscope 2.0. Cette première phase exploratoire a été effectuée manuellement, les collocations trouvées étant celles que le Lexicoscope identifie comme saillante dans les lexicogrammes. Ensuite, nous avons sélectionné des pivots pertinents du point de vue des nominations, jouant le rôle de modifieur (adjectival ou nominal) dans ces expressions, et nous les avons classés par la polarité positive et négative (en vue de travaux ultérieur sur la polarité, que nous n'avons pas eu le temps de mener ici).

Champ thématique de l'écologie									
Polarité positive								Polarité négative	
1	durable	soutenable	recyclable	ajusté	? sobre	zéro		1	punitif
2	commun	partagé	solidaire	inclusif	coopératif	participatif	citoyen	2	contraignant
3	vertueux	responsable	doux	heureux				3	délirant
4	naturel	vert	sauvage	fertile					
5	d'après	post	de demain	en transition	désirable	utopique			
Champ thématique du Covid									
Polarité négative									
1	barrière								
2	distanciation								
3	confinement	déconfinement							
4	monde d'après	monde d'avant							

Figure 15. Listes de pivots

polarité	pivot	expression	marqueur	phrase entière
négative	barrière	geste barrière		Pris de court, les exploitants n'ont pas encore validé le casse-tête des jauges et des gestes barrières : faudra-t-il disperser les spe
			« »	Chez les CP, cinq élèves sur onze n'avaient pas entendu parler des « gestes barrières .
				L'opération, complexe, implique un réaménagement de l'ensemble des sites, une redéfinition des pratiques et procédures intégra
			être	Ainsi, pour la présidente du parti nationaliste, fermer les frontières aurait tout simplement dû être le premier geste barrière .
			« »	Et qui oserait minimiser l'importance des fameux gestes « barrière » ?
		mesure barrière	" "	La multiplication des clusters montre que le système de « dépistage-traçage-mise en quarantaine » est « dépassé », juge-t-il, app
			" "	Rappelant qu'il s'agit d' « une liberté fondamentale », le juge des référés a considéré que, « sauf circonstances particulières », l'i
			visant à	A Pékin, les chancelleries occidentales sont pointées du doigt pour leur prétendue nonchalance vis-à-vis des mesures barrières
			« »	Car, « il est établi que des personnes en période d'incubation ou en état de portage asymptomatique excrètent le virus et entreti
				« Les mesures barrières les plus rigoureuses auraient pu être mises en oeuvre plus précocement : le filtrage, l'information des pa

Figure 16. Extrait du recueil d'expressions

Pour ces différentes expressions, afin de mener une première étude qualitative, nous avons constitué des listes de contexte spécifiant marquant le caractère émergent de la nomination. Ces contextes peuvent être marqués de différentes manières : l'explicitation (p. ex. *c'est-à-dire*) ; les marques de prise en charge énonciative, métadiscours, guillemets (p. ex. *agriculture "soutenable"*) ; et l'ajout d'adverbes évaluatifs qui signalent le caractère parfois abusif de certains usages, et une prise de recul du locuteur par rapport à l'expression (p. ex. *agriculture vraiment durable*).

Explicitation (ici grâce aux parenthèses ou à l'énumération)		
	Marqueur	Exemple
1	( )	locomotions douces (train, vélo, trottinette)
2	[ ]	locomotions douces [train, vélo, trottinette]
3	:	locomotions douces : train, vélo, trottinette
4	c'est-à-dire	locomotions douces c'est-à-dire le train, le vélo, ...
5	Par exemple	locomotions douces par exemple le train, le vélo, ...
Marques de prise en charge énonciative, métadiscours, guillemets (discours rapporté)		
	Marqueur	Exemple
1	dite	croissance dite verte
2	soi-disant	croissance soi-disant verte
3	comme on dit aujourd'hui	croissance verte, comme on dit aujourd'hui
4	" "	agriculture "soutenable"

Figure 17. Exemples de marques définissant les contextes spécifiant

Après une sélection manuelle et une vérification à l'aide de mon tuteur Olivier Kraif, dans une première annotation, autour des nominations potentielles, 341 phrases ont été considérées comme comportant des contextes spécifiants. Pour équilibrer les données, afin que pour chaque classe le nombre de données soit égal, nous avons choisi par ailleurs 341 phrases "communes", avec des contextes ne contenant aucune marque, dans la première version du jeu de donnée. Plus précisément, pour chaque expression, nous avons ajouté le même nombre de contextes non spécifiants que de spécifiants, afin que le classifieur s'intéresse aux traits contextuels indépendamment de l'expression ciblée. 682 phrases de données au total ont été recueillies, chaque nomination étant entouré de balises <expr> et </expr> pour les identifier.

Ensuite, pour la qualité de l'annotation, nous avons mené une double annotation. Les listes de contextes, nettoyées de toute information additionnelle (à part les balises <expr> entourant les expressions cibles), ont été confiées aux deux encadrants du stage (A. Jackiewicz et O. Kraif) pour être annotées. Ce travail a été effectué complètement à la main, et consistait pour chaque phrase à ajouter un tag 1 ou 0 afin de décider si le contexte de l'expression est jugé spécifiant ou pas (0=pas spécifiant ou 1=spécifiant), par exemple[19] :

[19] Plus précisément la consigne était formulée de la manière suivante : « Les usages que l'on cible doivent indiquer à celui qui lit, de manière implicite : "attention, je (l'allocuteur) n'utilise pas cette expression dans le sens habituel - soit que je l'ai empruntée à d'autres (avec guillemets), soit qu'il s'agisse d'un néologisme, soit que je ne sois pas d'accord avec le terme, soit qu'il y ait besoin d'explicitation ou d'une définition.". Il s'agit donc de chercher des marques métadiscursives qui doivent orienter l'interprétation du récepteur (allocutaire) et attirer son attention sur le caractère émergent de la nomination. L'interprétation dépend beaucoup du

1 Consignes en verre et/ou en « <expr>plastiques recyclables</expr> » issus de l'agriculture (maïs par exemple) pour tous les produits.

0 Certaines regroupent le carton et le papier, certaines les séparent, certaines mettent les <expr>plastiques recyclables</expr> avec les autres éléments recyclables.

Il faut noter qu'il s'agit d'une annotation délicate, impliquant une interprétation fine du contexte, et des marques de surface telles que les guillemets ne suffisent pas à indiquer un contexte spécifiant. De fait, ces marques sont parfois ambiguës.

Considérons l'exemple suivant : « Nous avons tous notre "technologie zéro carbone préférée", toutefois il ne s'agit pas de booster notre ego mais bien de réduire les émissions de CO2. ». Dans cette phrase les guillemets peuvent marquer la nouveauté de l'expression « technologie zéro carbone », mais aussi elles peuvent porter sur l'expression « préférée » (pour indiquer une certaine familiarité).

Pour une annotation finale de qualité, nous procédons donc à une annotation multiple qui permet de voir les convergences et les divergences. Elle consiste tout d'abord à faire annoter par plusieurs annotateurs puis à observer les annotations qui divergent afin d'en discuter de manière intersubjective, afin de trouver un consensus et de déterminer s'il convient d'affiner les critères et de réannoter ces cas. Nous avons sélectionné toutes les données qui sont marquées par le tag différent et discuté en expliquant nos interprétations. Normalement, il aurait fallu refaire l'annotation et trouver un consensus pour toutes les données, néanmoins pour des raisons de temps, et dans la mesure où les divergences étaient minoritaires (une centaine de cas), nous avons décidé de nous limiter aux annotations convergentes pour former le jeu de données final.

Quant à l'évaluation de l'annotation humaine, elle consiste habituellement au calcul de l'accord Inter-Annotateur (AIA) pour mesurer la qualité et la fiabilité des annotations. Deux mesures s'appliquent fréquemment sur les résultats d'accords et de désaccords : la F-mesure et le Kappa de Cohen. Pour le calcul de la précision des contextes spécifiants, nous avons choisi l'annotation de Madame Jackiewicz comme référence, et calculé le Kappa de Cohen pour avoir un score (K = (Pa-Pe)/(1-Pe) ou Pa est la proportion d'accord obtenu, et Pe la probabilité d'obtenir un accord aléatoirement).

---

contexte (polysémie), du coup il ne suffit pas qu'il y ait un modifieur (p.ex. *plus durable*) ou des guillemets, ou un *c'est-à-dire*, ou des parenthèses pour que ça marche. »

Annotation_AJ / Annotation_OK&YD	0	1
0	305	36
1	65	276

Table 8. Accord des deux annotations sur le corpus de nomination

Précision (calculée sur l'annotation des 1 seulement)	Rappel (calculé sur l'annotation des 1 seulement)	Pa (accord observé)	Pe (Proba. accord aléatoire)	K (kappa de Cohen)
0.88	0.809	0.852	0.5	0.70

Table 9. Résultats de métriques de AIA sur le corpus de nomination

D'après l'échelle de Landis et Koch (1977), on se situe ici dans le cas d'un accord fort, même s'il n'est pas parfait : cela nous permet de valider cette première annotation.

Finalement, on en tire 581 phrases pour l'entraînement en retenant les annotations communes. Chaque ligne de donnée est ainsi construite :

```
tag + tabulation + pivot + tabulation + contexte
```

(le pivot étant remplacé par le marqueur `{tokenizer.mask_token}` en vue de l'expérience de prédiction avec MLM.

Par exemple :

```
0 --> durable --> Cette vague d'achats est le signe que la finance {tokenizer.mask_token} devient importante.
```

Puisque les classifieurs se fondent sur des modèles d'apprentissage, pour les entraîner, il y a besoin de trois sous jeux de données qui répondent aux trois étapes de l'entraînement : apprentissage, validation et test. Concernant ces sous jeux de données, nous proposons de répartir le jeu de données final en 80% TRAIN + 10% DEV + 10% TEST en effectuant une sélection aléatoire.

Il faut veiller à l'équilibre de données lors de l'apprentissage, c'est-à-dire qu'on s'assure que le nombre de chaque label est même dans chaque ensemble, afin que les modèles puissent apprendre équitablement les caractères de chaque classe. Plus précisément, nos expériences ne concernent que deux classes : spécifiant et non spécifiant. Ainsi nous avons formé le jeu de données TRAIN avec un nombre de tag 0 est égal au tag 1 (232 tags 0 et 232 tags 1).

En plus, en s'inspirant du seuil de fenêtre 10 dans l'approche distributionnelle de la nomination (Cassier, 2020), nous proposons d'ajouter une expérience en réduisant la taille des phrases en ne laissant qu'un contexte de -5 /+5 mots autour du mot pivot pour tous les modèles. Finalement, on aboutit aux jeux de données d'origine et aux nouvelles versions qui fixent la taille du contexte à 10.

	Jeu de données (contexte complet)				Jeu de données (taille de contexte10)			
	Global	TRAIN	DEV	TEST	Global	TRAIN	DEV	TEST
NB phrases	581	464	58	59	581	464	58	59
NB tokens	18102	14597	1701	1804	5589	4454	557	578

Table 10. Jeux de données pour les classifieurs

2. Jeux de données pour le modèle MLM

Dans cette partie de l'expérimentation, comme nous supposons que les nominations émergentes sont moins bien prédites que les expressions communes, il nous faut recueillir un ensemble d'expressions communes à titre de comparaison.

Vu les objectifs différents, les critères de construction sont naturellement dissemblables. Dans cette tâche, nous n'exigeons une annotation manuelle, mais il faut encore masquer le pivot des expressions à prédire. Chaque ligne de donnée est ainsi construite :

Pivot + tabulation + contexte (où le mot à prédire est masqué comme précédemment)

Par exemple :

```
Vertueux --> Récompenser les comportements {tokenizer.mask_token}.
```

Comme nous ne travaillons que sur les deux patterns de nomination « N+ADJ » et « N+PREP+N », et pour obtenir des contextes comparables à ceux que nous avons déjà extraits, nous avons extrait toutes les expressions correspondant à ces patterns. Ayant recours à la plate-forme Lexicoscope 2.0, nous avons trouvé 7900 expressions au total de la forme NOUN + ADJ, et 4123 pour les expressions NOUN + PREP + NOUN. En observant systématiquement les nominations émergentes sélectionnées, nous avons noté que, dans le cas de N+ADJ, le glissement sémantique a lieu le plus souvent sur l'adjectif, et dans le cas de N+PREP+N, ces phénomènes linguistiques sont souvent relatifs au deuxième nom, le complément. C'est pourquoi nous avons systématiquement défini l'adjectif et le complément du nom comme le pivot à masquer.

Concernant les données de nomination, nous reprenons les 682 phrases autour des nominations sélectionnées, qu'il s'agisse de contextes spécifiants ou non spécifiants. Pour mesurer l'influence de la taille de fenêtre 10 sur la prédiction, nous constituons également un jeu de données avec les versions correspondantes.

	Contexte complet			Taille de contexte 10		
	N+ADJ	N+PREP+N	Nominations	N+ADJ	N+PREP+N	Nominations
NB phrases	7900	4123	682	7900	4123	682
NB tokens	265559	143426	21460	76847	40930	6577

Table 11. Jeux de données pour le modèle MLM

Chapitre 5. Construction et entraînement du classifieur

Dans l'objectif d'assister à la détection des nominations émergentes de façon automatique, nous proposons de concevoir un classifieur qui vise à signaler si le contexte d'une nomination est jugé spécifiant ou non. Du côté de TAL, le problème peut être considéré comme un problème de classification de séquence, et s'appuyer sur des réseaux de neurones. Nous avons ainsi entrainé des classifieurs en nous appuyant sur des modèles réputés performants et populaires.

Deux modèles de réseaux de neurones nous intéressent : un FNN (Feed forward, Neural Network) et le modèle transformer de BERT. Vu que nous travaillons sur le français, nous employons finalement le modèle FlauBERT (Le et al., 2019). Cette architecture étant très gourmande en ressources [20] , le modèle flaubert/flaubert_base_uncased semble un choix pertinent - il est construit avec 12 couches, 768 dimensions pour les *embeddings*, 12 têtes comporte au final 138M de paramètres.

1. Jeux de données

Etant donné que nous comptons comparer les performances de deux classifieurs sur une même tâche, les jeux de données constituent un facteur sensible pour la suite des expériences. Pour garantir des conditions comparables pour ces deux classifieurs, les jeux de données correspondants aux différentes étapes pour chaque modèle doivent être identiques. Par exemple :

```
0 --> partagé --> Il faut mettre en place un site public du covoiturage et du transport {tokenizer.mask_token}.
```

Comme nous l'avons expliqué dans le chapitre 4, nous avons trois jeux de données qui sont une partition des 581 données annotées : TRAIN (80%, 464 lignes), DEV (10%, 58 lignes) et TEST (10%, 59 lignes). Nous résumons la construction de ces trois ensembles par ces trois étapes :

1. Lire chaque ligne de donnée et l'enregistrer dans une liste correspondante en fonction du tag 0 ou 1

[20] Pour mener ces expériences, nous avons pu bénéficier d'un serveur financé par la chaire d'excellence MIAI : un DELL Precision 7820 basé sur deux processeurs Intel Xeon à 6 coeurs et 32 Go de Ram, avec une carte GPU Nvidia Quadro RTX5000 dotée de 16 Go de RAM.

2. Pour le TRAIN, sélectionner aléatoirement 232 éléments dans la liste de tag 0 et 232 éléments avec le tag 1 ; Pour le DEV, parmi les éléments restants, sélectionner aléatoirement 29 éléments dans la liste de tag 0 et 29 éléments avec le tag 1 ; le reste constitue le TEST
3. Réordonner les données de chaque corpus, avec un ordre aléatoire

Concernant le corpus de contextes restreint à 10, on utilise le corpus TRAIN, DEV et TEST en limitant les contextes de 10 mots à l'entour.

2. Classifieur basé sur le FNN

Vu les performances des réseaux FNN sur la tâche de classification de textes (Glazkova, 2020), nous avons choisi d'établir un classifieur basé sur ce modèle. Dans cette partie, nous détaillons d'abord la construction[21], ensuite les expériences et leurs résultats.

2.1. Architecture

Avant d'expliquer l'architecture du classifieur, examinons brièvement ce réseau de neurones. Le réseau de neurones à propagation avant, en anglais *feed forward neural network* (FNN), est un réseau de neurones artificiels. Il se différencie de réseau de neurones récurrent par les connexions entre les nœuds qui ne forment pas une boucle, autrement dit le FNN est acyclique.

C'est le premier et le plus simple réseau de neurones artificiels, car c'est un perceptron multicouche. Cela veut dire que dans ce réseau, les informations ne se déplacent que dans une direction, vers l'avant, depuis les nœuds d'entrée, en traversant les nœuds cachés puis vers les nœuds de sortie. Ainsi, il n'y a pas de cycles ou de boucles dans ce réseau.

Un FNN peut être constitué par plusieurs couches cachées qui sont connectées chacune avec l'autre. Plus précisément, chaque nœud d'une couche est connecté à tous les autres nœuds de deux couches adjacentes. Dans chaque couche du réseau de neurones, à l'exception de la couche de sortie, la sortie de la couche actuelle ne peut pas directement être utilisée comme l'entrée de la couche suivante, il faut lui appliquer une fonction, qu'on appelle la fonction d'activation. Certaines fonctions d'activation sont fréquentes dans les FNN, suivant les sorties requises pour l'activation : sigmoïde (valeurs entre 0 et 1),

[21] https://github.com/yumengding/nomination-mergente

fonction tangente hyperbolique *tanh* (valeurs entre -1 et 1) et Fonction Unité Linéaire Rectifiée ReLU (permettant de ne faire passer que les valeurs positives).

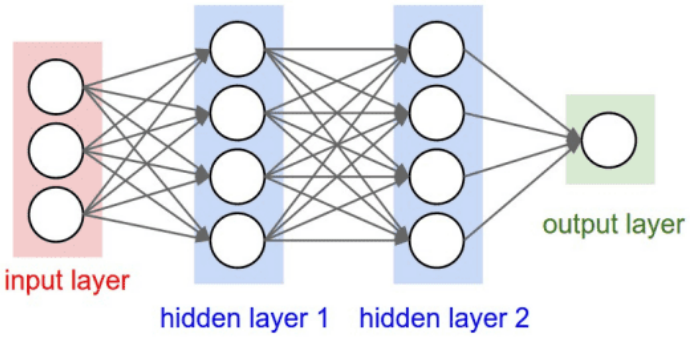


Figure 18. Architecture d'un FNN (Mukul Rathi, 2018)

Concernant notre classifieur, nous assemblons deux parties : d'abord la transformation du mot en vecteur, puis le modèle FNN. Nous choisissons le *word embedding* contextuel de FlauBERT grâce aux avantages expliqués dans l'état de l'art. Il consiste à encoder les données littérales en représentations vectorielles, après avoir obtenu une tokenisation de type *byte-pair encoding* à l'aide du FlaubertTokenizer[22]. Ici, nous prenons les vecteurs des mots cibles (avec 768 dimensions pour FlauBERT) comme sorties.

Nous établissons sur un modèle de FNN simple qui se compose de 100 couches cachées (comme le corpus d'entraînement n'est pas gros, il demande le réseau plus simple)[23], une fonction d'activation Sigmoïde est appliquée, elle fixe les sorties entre (0,1) pour effectuer une classification binaire. Ce modèle permet d'apprendre une donnée chaque fois et transférer vers l'avant pour la repondération jusqu'à la fin de données. Et ce processus va se répéter plusieurs fois en fonction du nombre d'itérations ou époques (de l'anglais *epochs*) au cours de l'entraînement.

2.2. Entraînement et expériences

Pour l'entraînement, on utilise la méthode la plus commune : pendant chaque époque, le système apprend avec toutes les données de TRAIN (par descente de gradient), puis on teste le réseau actuel avec le jeu de données DEV. Le principe est que la fonction de perte doit descendre continûment lors de l'apprentissage, mais quand la perte de DEV commence à augmenter, ce qui indique un *overfitting*, on peut fixer le nombre d'époques requis. On répète cette procédure jusqu'à une réduction de la fonction de perte, et une

[22] https://huggingface.co/transformers/model_doc/flaubert.html
[23] https://www.kaggle.com/rafjaa/dealing-with-very-small-datasets

descente presque invariable pour arrêter l'apprentissage. Enfin on enregistre le modèle entrainé comme modèle final afin d'utiliser les données de TEST pour évaluer.

Pour éviter le problème du surapprentissage qui est causé par une correspondance trop précise à la collection particulière de données, nous utilisons la méthode de l'abandon qui permet d'ignorer un pourcentage de nœuds lors de la transmission des valeurs (Hinton, et al., 2012) (*dropout* p=0.75), et l'arrêt précoce ou « early stopping » mis en pratique à l'aide de l'outil pytorchtools[24]. Nous posons ainsi que si la perte pour le corpus de validation DEV ne se réduit plus pendant 20 époques, l'entraînement se termine en avance, et on enregistre le modèle ayant la perte la plus basse lors du développement. Nous fixons les paramètres à l'aide d'une initialisation aléatoire .

Néanmoins, lors de l'entraînement, nous trouvons que le taux d'apprentissage (vitesse d'apprentissage qui est un paramètre important dans un algorithme d'optimisation) initial (*lr*=0.001) de l'optimisation Adam (algorithme d'optimisation ayant pour objectif de mettre à jour les poids du réseau et de se déplacer vers une fonction de perte minimum) ne fonctionne pas très bien, la perte de TRAIN ne descend pas beaucoup en peu de temps. Dans ce cas, on ajoute une expérience afin de fixer un meilleur taux d'apprentissage correspondant au modèle. Trois taux d'apprentissage fréquents sont testés : 0.001, 0.0001 et 0.00001.

A l'aide de l'outil Weights & Biases (wandb)[25], nous enregistrons le processus de l'entraînement, y compris l'évolution de la perte (en anglais *loss*) et d'exactitude pour TRAIN et DEV, et les représentons par les courbes ci-dessous.

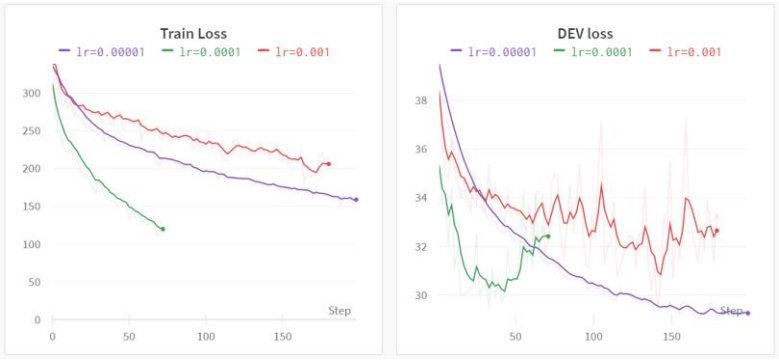


Figure 19. *Loss* de TRAIN et DEV avec des taux d'apprentissage différents

[24] https://pypi.org/project/pytorchtools/
[25] https://wandb.ai/site

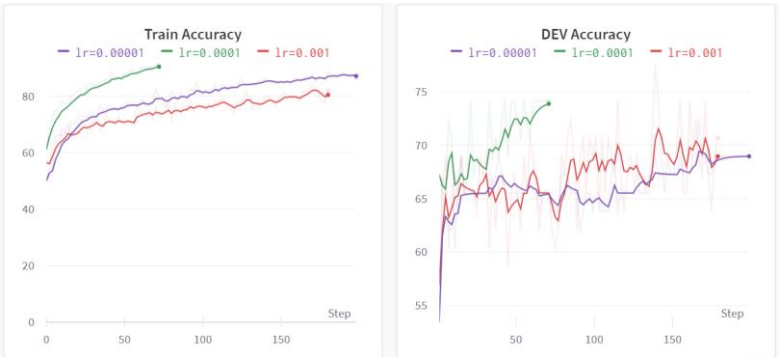


Figure 20. Exactitude de TRAIN et DEV avec des taux d'apprentissage différents

En observant les courbes, nous nous apercevons que globalement la perte descend plus beaucoup et plus vite quand on réduit le taux d'apprentissage de 0.001 à 0.0001, de même, le score de l'exactitude devient meilleur. Pourtant avec la réduction continue à 0.00001, une dégradation apparait pour les performances. Il semble donc qu'il existe un seuil à partir duquel les résultats ne s'améliorent plus. Cette observation nous pousse à fixer notre taux d'apprentissage à 0.0001 pour nos prochaines expérimentations.

Puisqu'une taille de contexte réduit à 10 semble représenter un seuil qui équilibre le mieux la syntaxe et la sémantique, nous avons répété l'expérience pour ces contextes restreints.

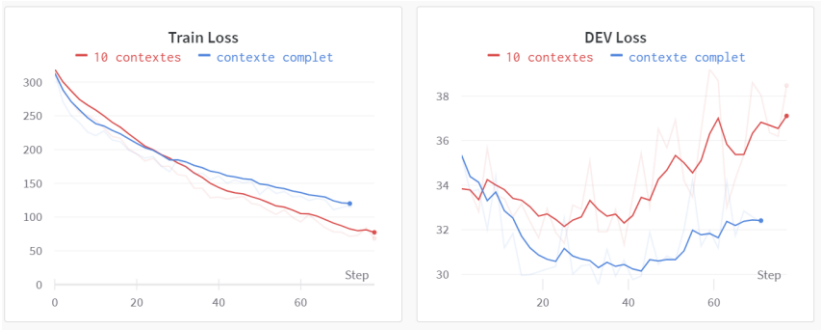


Figure 21. Contexte complet vs contexte de 10 : Loss de TRAIN et DEV

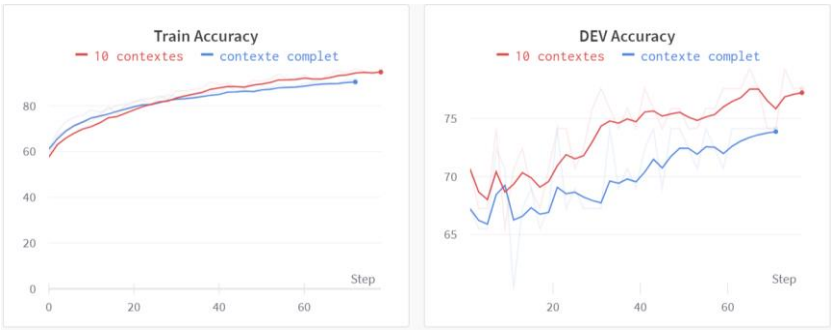


Figure 22. Contexte complet vs contexte de 10 : Exactitude de TRAIN et DEV

Par ces courbes, nous nous apercevons que du côté de la perte, les contextes restreints aboutissent à une descente rapide dans le TRAIN. Comme ils rencontrent le surapprentissage plus tôt selon DEV, il rend l'entraînement plus rapide. Du côté de l'exactitude, leur performance est similaire au cours de l'apprentissage, mais dans le DEV, les contextes de10 mots apportent une amélioration nette.

	Nb prédiction = tag (0 et 1)	Précision (calculée sur le tag 1 seulement)	Rappel (calculé sur le tag 1 seulement)	F1 %
Contexte complet	41	0,462	0,75	57,2
Contextes de 10	43	0,5	0,437	46,6

Table 12. Contexte complet vs 10 contextes : Scores du TEST

En général, sur la tâche de classification binaire et notamment quand nous nous intéressons aux classes qui ont moins de données (normalement les nominations sont moins fréquentes que les expressions communes), la précision, le rappel et le score F1 qui combine subtilement les deux sont les métriques plus fiables. Comme l'identification des contextes spécifiants nous intéresse mieux, ici nous nous concentrons sur les tags 1. Nous notons que selon les résultats, le contexte complet se conduit mieux pour la tâche de la classification des séquences. Nous avons ainsi fixé le contexte complet pour les expériences suivantes.

Lorsqu'on représente les mots ou les phrases par les vecteurs à l'aide du FlaubertTokenizer, les *embeddings* sont normalement le résultat de la dernière couche cachée, comme les *embeddings* qu'on a employés dans les expériences précédentes. Néanmoins, comme BERT comporte 12 ou 24 couches et que chaque couche produit en sortie des vecteurs de 768 ou 1024 dimensions, plusieurs formes de représentations de plongement de mots sont disponibles. Il est attesté que le choix du meilleur *embedding* contextuel dépend de la tâche, notamment à travers une expérience sur la reconnaissance d'entité nommée (NER) avec 6 embeddings de couches différentes (Devlin, al, 2018). Pour cette tâche, les résultats démontrent que la concaténation des quatre dernières couches cachées obtient une meilleure performance.

System	Dev F1	Test F1
ELMo (Peters et al., 2018a)	95.7	92.2
CVT (Clark et al., 2018)	-	92.6
CSE (Akbik et al., 2018)	-	**93.1**
Fine-tuning approach		
$BERT_{LARGE}$	96.6	92.8
$BERT_{BASE}$	96.4	92.4
Feature-based approach ($BERT_{BASE}$)		
Embeddings	91.0	-
Second-to-Last Hidden	95.6	-
Last Hidden	94.9	-
Weighted Sum Last Four Hidden	95.9	-
Concat Last Four Hidden	96.1	-
Weighted Sum All 12 Layers	95.5	-

Figure 23. Les combinaisons d'*embeddings* contextuels pour la tâche NER (Jacob Devlin et al., 2018)

Nous supposons ainsi que le choix de l'*embedding* contextuel influence la tâche de classification. Nous procédons à trois expériences à la suite, en fixant les paramètres initiaux et les données d'entrée de taille complète, en ne faisant varier que les *embeddings* : la dernière couche cachée, la somme des quatre dernières couches cachées et la concaténation des quatre dernières couches cachées, afin de comparer indépendamment leur influence sur la performance du classifieur.

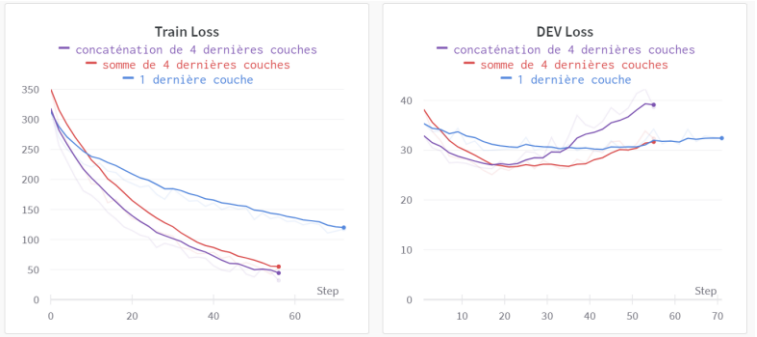


Figure 24. Loss de TRAIN et DEV sur les *embeddings* contextuels différents

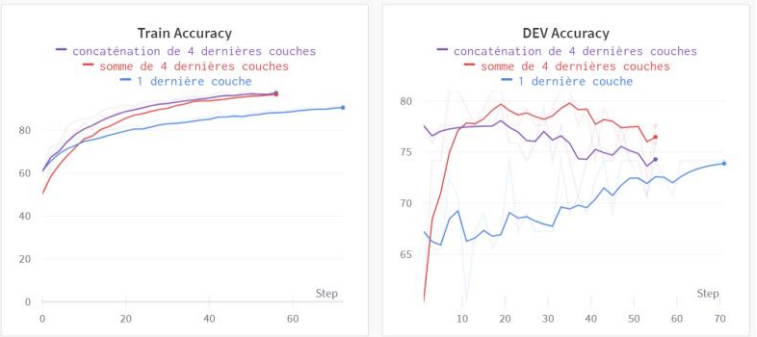


Figure 25. Exactitude de TRAIN et DEV sur les *embeddings* contextuels différents

En comparant les courbes dans chaque graphe, on s'aperçoit que, sur la perte au cours de l'apprentissage et du développement, la concaténation de quatre dernières couches

cachées permet une descente plus rapide, et la somme de 4 dernières couches cachées est moins performante, la dernière couche cachée étant la plus mauvaise. Pourtant, concernant l'exactitude de DEV, on trouve que la somme de quatre dernières couches cachées fonctionne mieux, et puis vient la concaténation, la dernière couche cachée étant toujours au troisième rang.

Par contre, cet ordre ne détermine pas définitivement leur performance sur l'identification finale, vu les résultats non satisfaisants sur le TEST en raison du jeu de donnée petit, nous faisons une évaluation croisée. On répète 10 fois le découpage en TRAIN 80% + DEV 10% + TEST 10%, et à chaque tirage TEST est bien distinct de TRAIN et DEV. On calcule la moyenne pour chaque mesure.

	MSE	RMSE	Précision (calculée sur le tag 1 seulement)	Rappel (calculée sur le tag 1 seulement)	F1 %
Dernière couche	0,256	0,502	0,747	0,706	72,4
Somme des 4 dernières	0,231	0,479	0,765	0,739	74,9
Concaténation des 4 dernières	0,242	0,492	0,756	0,723	73,7

Table 13. Scores du TEST des trois *embeddings* contextuels

En comparant ces chiffres, nous constatons que les performances de ces trois expériences présentent des nuances selon les scores de MSE (Mean Square Error)[26], RMSE (Root Mean Square Error), précision, rappel et F1. En effet, ils démontrent que le choix de l'*embedding* contextuel influence la performance du modèle. Avec une comparaison, la somme des 4 dernières couches cachées obtient le score MSE le plus bas et le F1 score le plus haut. Nous pouvons ainsi déterminer que pour l'identification des contextes spécifiants, l'embedding de la somme des 4 dernières couches cachées semble globalement mieux adaptée.

Pour résumer, le meilleur classifieur basé sur FNN utilise les paramètres suivants : un taux d'apprentissage de 0.0001, les données d'entrée des contextes complets et des *embeddings* issus de la somme des 4 dernières couches cachées.

[26] Métrique MSE mesurée à l'aide du module sklearn.metrics https://scikit-learn.org/stable/modules/generated/sklearn.metrics.mean_squared_error.html

3. Classifieur basé sur le FlauBERT

Ce classifieur se fonde sur le modèle FlauBERT, particulièrement sur le modèle FlaubertForSequenceClassification[27] qui vise à la tâche de classification des séquences, par *fine-tuning* du modèle pré-entrainé. Comme dans la partie précédente, nous expliquons d'abord la construction du classifieur[28], puis nous décrivons les expériences et analysons leurs résultats.

3.1. Architecture

Comme pour le classifieur FNN, ce classifieur comporte deux étapes : le prétraitement, qui consiste à transformer l'entrée en vecteur puis l'entraînement, consistant à entrer les vecteurs et les tags lors de la phase d'apprentissage. Ici, nous avons en entrée toute la phrase (et non pas le vecteur d'un seul mot), représentée à l'aide d'une suite d' « input_ids » (les identifiants des tokens dans le vocabulaire du modèle) et un tenseur pour « attention_mask » (se différencie de token masqué{tokenizer.mask_token}) qui évite d'être attentif aux tokens de remplissage (quand la longueur de l'entrée est moins que la longueur maximum, il ajoute les *paddings* avec la valeur 0). Avec le tag qu'on annoté, cela fait trois tenseurs qui constituent les données pour l'entraînement.

3.2. Entraînement et expériences

L'étape suivante est la même que pour le classifieur FNN : on répète le processus d'apprentissage et de développement pour plusieurs époques, puis on sauvegarde le modèle entraîné afin de procéder à une évaluation finale. Dans l'article de BERT (Devlin, al, 2018), les auteurs ont proposé 2, 3 ou 4 pour le nombre d'époques. Dans ce cas, on n'emploie plus la méthode d' « *early stopping* », mais on teste directement 5 époques et on examine les courbes de perte de TRAIN et DEV, afin de localiser le moment où les pertes de TRAIN et DEV sont les plus basses avant le surapprentissage.

Au cours de l'entraînement, à la différence du FNN, on soumet les données d'apprentissage par paquets, afin de paralléliser certains traitements. Il est conseillé de fixer la taille des paquets, ou *batch size*, à 16 ou 32. L'article de BERT recommande aussi les valeurs de 5e-5, 3e-5 et 2e-5 pour le taux d'apprentissage. Ces deux paramètres vont influencer la vitesse et la qualité d'apprentissage : le *batch size* affecte la capacité de

[27] https://huggingface.co/transformers/model_doc/flaubert.html
[28] https://github.com/yumengding/nomination-mergente

généralisation, et le taux d'apprentissage concerne la convergence du modèle. Il est donc nécessaire de choisir ces valeurs prudemment. Vu l'interdépendance de ces deux facteurs, nous pouvons tenter les expériences suivantes sur des combinaisons diverses. Dans cette suite d'expériences, nous prenons les jeux de données avec les contextes complets.

Au cours de l'entraînement, à la différence du FNN, on traite les données d'apprentissage par paquets, afin de paralléliser certains traitements : les *batch sizes* (nombre de données fournies par paquet) de 16 et 32 sont conseillés. L'article de BERT recommande les valeurs de 5e-5, 3e-5 et 2e-5 pour le taux d'apprentissage. Ces deux paramètres vont influencer la vitesse et la qualité d'apprentissage : en lien avec la repondération, le batch size affecte la capacité de généralisation et l'autre concerne la convergence du modèle. Il est donc nécessaire de choisir prudemment. Vu ces deux facteurs interdépendants, nous pouvons tenter les expériences suivantes sur les combinaisons diverses. Dans cette suite d'expériences, nous prenons les jeux de données de contexte complet.



Figure 26. *Loss* de TRAIN et DEV pour diverses combinaisons de *batch size* et de taux d'apprentissage

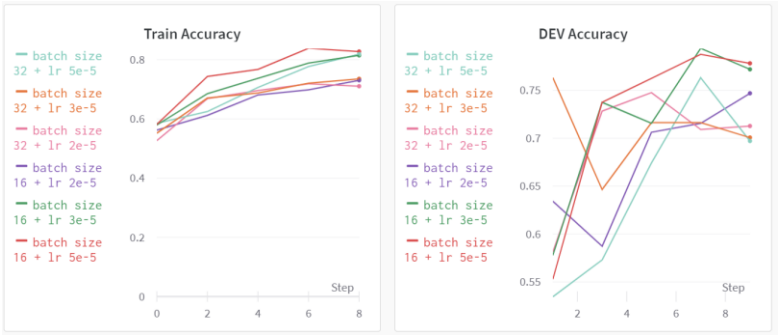


Figure 27. Exactitude de TRAIN et DEV pour des combinaisons diverses de *batch size* et de taux d'apprentissage

En observant les courbes de pour ces 6 combinaisons différentes, concernant la perte de TRAIN et DEV, il est évident que la courbe rouge descend au plus bas et plus vite

pendant 5 époques. En plus, du côté de l'exactitude, la rouge possède la meilleure performance pour le TRAIN. Même si pour le DEV, elle est dépassée par la courbe verte en un point, globalement, c'est encore la rouge qui est supérieure aux autres. Ainsi, nous fixerons le *batch size* à 16 et le taux d'apprentissage à 5e-5 pour les expériences suivantes.

Comme nous l'avons dit, la taille 10 de fenêtre du contexte nous intéresse. Une expérience de comparaison est ainsi mise en œuvre : classification par les contextes complets et par les contextes réduits à 10 tokens. On entraîne le modèle pendant 5 époques et enregistre le meilleur état en fonction de la descente de perte.



Figure 28. Contextes complets vs 10 contextes : Loss de TRAIN et DEV

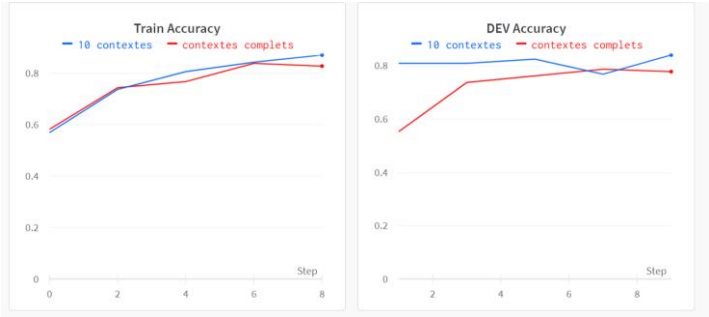


Figure 29. Contextes complets vs 10 contextes : Exactitude de TRAIN et DEV

Lorsqu'on concentre l'attention sur la courbe rouge, nous nous apercevons que, à la quatrième époque, la perte de DEV est la plus basse avant l'augmentation. En même temps, le modèle manifeste la meilleure exactitude. Nous choisissons ainsi d'enregistrer le modèle avec les contextes complets après l'entraînement de quatre époques. En suivant le même principe pour le modèle entraîné avec les contextes de taille 10, nous choisissons d'arrêter l'entrainement à la fin de l'époque 3. Concernant l'exactitude de ces deux expériences, globalement, les contextes réduits se comportent mieux, avec une courbe presque toujours au-dessus de celle des contextes complets.

Après avoir passé les données du jeu de TEST dans les deux modèles enregistrés, nous reformulons les résultats et les mesurons avec les métriques dessous.

	Nb prédiction = tag	Précision (calculée sur le tag1 seulement)	Rappel (calculée sur le tag1 seulement)	F1 %
Contextes complets	47	0,6	0,75	66,7
Contextes réduits à 10	50	0,667	0,875	75,7

Table 14. Contexte complet vs 10 contextes : Scores du TEST

En comparant les chiffres de ces deux modèles, si l'on considère les résultats des contextes complets comme la ligne de base, on remarque une augmentation de chaque chiffre. Par conséquent, nous gardons le classifieur entraîné avec des contextes réduits à 10 comme représentant des classifieurs basés sur BERT.

4. Classifieur final

Puisque nous avons deux classifieurs s'appuyant sur deux réseaux de neurones, et nous ne gardons qu'un pour l'évaluation terminale, nous comparons leurs résultats sur le TEST afin de sélectionner le classifieur final.

	Précision (calculée sur le tag1 seulement)	Rappel (calculée sur le tag1 seulement)	F1 %
Classifieur FNN	0,765	0,739	74,9
Classifieur BERT	0,667	0,875	75,7

Table 15. Classifieur FNN vs Classifieur BERT : Scores du TEST

On constate un écart modeste mais quand même significatif entre ces deux classifieurs, ce qui nous permet de conclure que le classifieur BERT est plus performant sur cette tâche d'identification. Finalement, nous retiendrons le classifieur BERT entraîné avec 10 contextes comme résultat de cette expérimentation, et nous l'utiliserons dans les suivantes.

En outre, concernant l'hypothèse de taille de fenêtre réduite, on observe qu'elle contribue effectivement à une amélioration des résultats pour le classifieur fondé sur BERT, contrairement au classifieur FNN. Enfin, nous pouvons conclure que l'influence de la taille de fenêtre 10 dépend des modèles de réseaux de neurones et des tâches, et qu'elle est préférable aux modèles BERT pour la tâche de classification des séquences spécifiantes ou non.

Chapitre 6. Modèle MLM de Bert

Étant donné qu'une caractéristique des nominations émergentes réside dans le glissement de sens engendré par l'émergence d'un nouveau concept dans un contexte particulier, nous supposons qu'une nomination émergente se manifeste par une faible prédictibilité dans un modèle entrainé sur des corpus antérieurs à l'émergence de ce concept. Avec pour objectif de repérer ce changement sémantique qui dénote un candidat potentiel, nous tentons des expériences de prédiction s'appuyant sur le modèle MLM de BERT.

Pour les mêmes raisons pratiques que dans l'expérimentation du classifieur, nous effectuons nos expériences avec le modèle FlaubertWithLMHeadModel[29], qui se fonde sur le modèle MLM de BERT, entrainé sur un corpus français. Le modèle déjà préétabli et pré-entraîné (les corpus utilisés sont introduits dans le chapitre 2) est disponible, nous pouvons donc l'utiliser directement[30].

Trois expériences sont principalement mises en œuvre : la prédiction des nominations sélectionnées, la prédiction des expressions suivant le patron « N + ADJ » et la prédiction des expressions suivant le patron « N + PREP + N ».

1. Expériences

A part les 682 données qui contiennent des nominations sélectionnées, 7900 données autour de « N + ADJ » et 4123 pour la forme « N + PREP + N » sont également testées avec ce modèle, avec pour objectif de préciser deux baselines. Pour chaque donnée, on réduit aussi le contexte à la taille 10 et on remplace le mot cible par l'identifiant `tokenizer.mask_token` afin de le prédire.

Pour mieux mesurer la prédictibilité des nominations sélectionnées et les deux patterns, nous proposons trois mesures : la moyenne du score de similarité avec la forme originale, la moyenne de la probabilité estimée de la forme originale, la moyenne du rang de la forme originale dans la liste des formes prédites.

Il nous demande ainsi d'enregistrer ces trois chiffres pour chaque donnée. Concernant les résultats retournés, deux éléments nous semblent intéressants : les scores de prédiction

[29] https://huggingface.co/transformers/model_doc/flaubert.html#flaubertwithlmheadmodel
[30] https://github.com/yumengding/nomination-mergente

logit qui sont directement calculés en sortie, pour l'ensemble du vocabulaire, et la liste des prédictions triée par logit décroissant. Quant à la probabilité du token original, elle consiste à normaliser les scores *logits* en probabilités à travers une fonction softmax[31]. A propos de cette dernière mesure, nous pouvons calculer le rang du mot à prédire dans une suite de prédictions en repérant la position du mot original.

2. Résultats

Afin de présenter clairement les résultats, nous les reformatons et les enregistrons dans un fichier d'Excel[32]. Trois feuilles de calcul correspondent aux trois jeux de données. Pour chacune, on a 7 types d'information : type de construction, rang du mot masqué, score, probabilité, expression complète, mot attendu et contexte de phrase. Pour les contextes déjà annotés, on ajoute les deux colonnes concernant le caractère spécifiant ou non des contextes.

annotation Agata	type de construction	rang de mot masqué	score 11.3477372457	probabilité 0.1321698272	expression intéressante	mot attendu	contexte de phrase
0	nom+adj	2	17.05054473876953	0.009500052779912949	développement durable	durable	Il devisait avec les puissa
1	nom+adj	1	19.06382179260254	0.7018222212791443	développement durable	durable	En 1987, le développeme
1	nom+adj	1	17.849231719970703	0.8467938899993896	développement durable	durable	Il faudrait d'abord que l'c

Figure 30. Extrait de résultats de nominations sélectionnés

En observant les résultats, nous notons que certains mots originaux n'apparaissent pas dans la liste de prédiction (les 50 000 premières prédictions) que le modèle retourne. Dans ce cas, nous enregistrons « pas prédit » pour le rang afin de signifier le mot n'est pas prédit. Ces résultats sont résumés dans le tableau dessous :

	Rang moyen	Non prédit	Prédit en 1ère position	Score	Probabilité
Nominations émergentes	399,193	11,73%	22,09 %	11,348	0,132
nom+adj quelconques	274,953	7,92 %	30,37 %	12,877	0,195
nom+prep+nom quelconques	360,821	5,31 %	39,22 %	13,871	0,268

Table 16. Résultats de la prédiction pour les nominations et pour les constructions quelconques

[31] https://fr.wikipedia.org/wiki/Fonction_softmax
[32] https://github.com/yumengding/nomination-mergente

Quand on comparer les résultats finals de ces trois groupes de données, on constate que les nominations émergentes contiennent les tokens originaux non prédits les plus nombreux. Et quant aux tokens prédits par le modèle, même s'ils apparaissent dans la suite de prédictions, leur faible prédictibilité s'explique par un rang moyen plus haut et par un score et une probabilité plus basse. On constate par ailleurs que le pourcentage de bonnes prédictions (quand le mot original arrive en tête) est sensiblement plus faible pour les nominations. Cela confirme donc notre quatrième hypothèse selon laquelle :

- Les nominations émergentes sont plus difficiles à prédire par rapport aux expressions communes.
- Une faible prédictibilité est une caractéristique de nomination émergente, qui peut se remarquer en mesurant le rang dans la liste de prédiction, le score et la probabilité à être prédit.
- Un changement de sens est sans doute à l'origine de cette plus faible prédictibilité.

Dans le but d'analyser les effets de la taille de contexte de 10 sur la prédiction du mot masqué, on effectue une expérience comparée avec le contexte complet sur les jeux de données de 682 nominations émergentes.

	Rang moyen	Non prédit	Prédit en 1ère position	Score	Probabilité
Contexte complet	399,193	11,73%	22,09 %	11,348	0,132
Contexte de 10	5657,137	12,32 %	3,01%	4,362	0,017

Table 17. Contexte complet vs contexte de 10 : résultats de prédiction

Nous notons que par rapport au contexte complet, un contexte partiel rend la nomination plus difficile à prédire, d'où une augmentation très sensible du rang moyen et du pourcentage de non prédit, ainsi qu'une diminution du score et de la probabilité.

Partie 4

-

Résultats et discussions

Chapitre 7. Elargissement des données

Maintenant que l'on dispose d'un classifieur pour détecter les contextes spécifiants, et que l'on a évalué le critère de prédictibilité, il peut être intéressant d'évaluer nos critères sur des données moins restreintes afin de consolider nos hypothèses. Nous effectuons ainsi une dernière tâche expérimentale avec pour objectif de :

1. Tester le classifieur sur d'autres données plus ouvertes et plus hétérogènes.
2. Effectuer une double annotation manuelle des nominations (notamment pour en évaluer l'accord inter annotateur).
3. Tester l'hypothèse de la plus grande fréquence des contextes spécifiants pour les unités considérées comme des nominations.
4. Évaluer le critère de prédictibilité sur cet ensemble élargi.

Avant de mettre en œuvre le classifieur, il faut définir un corpus assez grand dont les données contenant des expressions intéressantes sont recueillies aléatoirement, incluant des nominations et des expressions communes. Avec une annotation double, nous distinguerons manuellement les nominations potentielles et les usages quelconques. Ensuite, nous comparerons séparément les résultats du classifieur et les résultats du modèle MLM avec les résultats de l'identification à la main, afin de déterminer si les 2 critères retenus (fréquence élevée des contextes spécifiants et faible prédictibilité) caractérisent correctement les nominations identifiées.

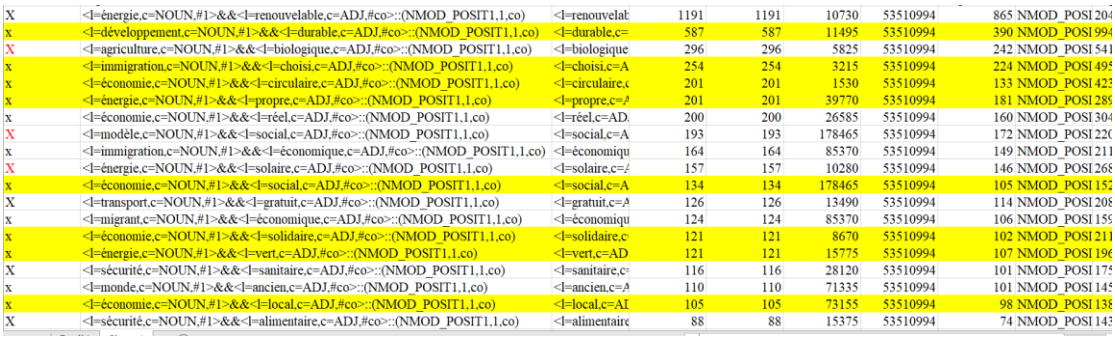
1. Construction du corpus élargi

Pour la sélection de données, nous avons suivi le même principe de repérage que pour les jeux de données précédents, en partant d'une amorce lexicale. Nous prenons dans un premier temps des noms « pivots » autour des thématiques du climat et du covid, d'autres sujets de société : *ville, énergie, écologie, société, monde, finance, transport, modèle, transformation, projet, gestion, agriculture, sécurité, économie, immigration, migrant, réfugié*

Ensuite, nous appliquons des critères textométriques : pour chaque nom, nous utilisons le lexicoscope pour extraire les 10 collocations adjectivales les plus significatives, ainsi que les contextes correspondants (les 200 premiers).

Au final, les collocations obtenues sont triées par fréquence décroissante et les 200 expressions les plus fréquentes sont retenues, accompagnées par 28678 phrases de contextes.

La tâche suivante consiste à annoter manuellement si les expressions correspondent, ou non, à une nomination émergente. A l'aide de mes encadrants, Olivier Kraif et Agata Jackiewicz, nous effectuons une annotation double. La consigne d'annotation était la suivante : « Vous semble-t-il que cette expression a été forgée, ou a pris un sens spécifique, récemment (au cours de la dernière décennie) afin d'introduire de nouvelles idées ? ».



X	<l=énergie,c=NOUN,#1>&&<l=renouvelable,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=renouvelal	1191	1191	10730	53510994	865	NMOD_POSI 204
x	<l=développement,c=NOUN,#1>&&<l=durable,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=durable,c=	587	587	11495	53510994	390	NMOD_POSI 994
X	<l=agriculture,c=NOUN,#1>&&<l=biologique,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=biologique	296	296	5825	53510994	242	NMOD_POSI 541
x	<l=immigration,c=NOUN,#1>&&<l=choisi,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=choisi,c=A	254	254	3215	53510994	224	NMOD_POSI 495
x	<l=économie,c=NOUN,#1>&&<l=circulaire,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=circulaire,c	201	201	1530	53510994	133	NMOD_POSI 423
x	<l=énergie,c=NOUN,#1>&&<l=propre,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=propre,c=A	201	201	39770	53510994	181	NMOD_POSI 285
x	<l=économie,c=NOUN,#1>&&<l=réel,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=réel,c=AD	200	200	26585	53510994	160	NMOD_POSI 304
X	<l=modèle,c=NOUN,#1>&&<l=social,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=social,c=A	193	193	178465	53510994	172	NMOD_POSI 220
x	<l=immigration,c=NOUN,#1>&&<l=économique,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=économiqu	164	164	85370	53510994	149	NMOD_POSI 211
X	<l=énergie,c=NOUN,#1>&&<l=solaire,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=solaire,c=A	157	157	10280	53510994	146	NMOD_POSI 268
x	<l=économie,c=NOUN,#1>&&<l=social,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=social,c=A	134	134	178465	53510994	105	NMOD_POSI 152
X	<l=transport,c=NOUN,#1>&&<l=gratuit,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=gratuit,c=A	126	126	13490	53510994	114	NMOD_POSI 208
x	<l=migrant,c=NOUN,#1>&&<l=économique,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=économiqu	124	124	85370	53510994	106	NMOD_POSI 155
x	<l=économie,c=NOUN,#1>&&<l=solidaire,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=solidaire,c	121	121	8670	53510994	102	NMOD_POSI 211
x	<l=énergie,c=NOUN,#1>&&<l=vert,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=vert,c=AD	121	121	15775	53510994	107	NMOD_POSI 196
X	<l=sécurité,c=NOUN,#1>&&<l=sanitaire,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=sanitaire,c	116	116	28120	53510994	101	NMOD_POSI 175
x	<l=monde,c=NOUN,#1>&&<l=ancien,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=ancien,c=A	110	110	71335	53510994	101	NMOD_POSI 145
x	<l=économie,c=NOUN,#1>&&<l=local,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=local,c=AI	105	105	73155	53510994	98	NMOD_POSI 138
X	<l=sécurité,c=NOUN,#1>&&<l=alimentaire,c=ADJ,#co>::(NMOD_POSIT1,1,co)	<l=alimentaire	88	88	15375	53510994	74	NMOD_POSI 143

Figure 31. Extrait d'une annotation double

Ici, les lignes en jaune marquent les convergences de ces deux annotations. Pour les lignes qui restent en blanc, les x en minuscule désignent les nominations identifiées par la première annotation, et les X en majuscule les nominations marquées par la deuxième.

La difficulté tient au fait que les nominations sont identifiées en dehors de tout contexte interprétatif. En outre, le caractère émergent des nominations réalise un vrai continuum : si certaine expression sont très anciennes, et d'autres très récentes, il existe un entre deux pour lequel il est difficile de trancher. Ici, en fonction de ses connaissances et de ses centres d'intérêt, chaque locuteur aura une perception très personnelle de la nouveauté langagière présentée par une nomination : il n'est donc pas étonnant que l'on ait des divergences importantes.

Pour quantifier ces différences, nous calculons à nouveau l'accord inter-annotateur (AIA) entre ces deux annotations.

Annotation_A / Annotation_OK-YD	Nomination	Expression commune
Nomination	17	7

Expression commune	21	155

Table 18. Corpus de l'évaluation : Accord Inter-Annotateur des deux annotations

Précision (calculée sur l'annotation des nominations seulement)	Rappel (calculée sur l'annotation des nominations seulement)	P_a (d'accord)	P_e (accord espéré aléatoire)	K (kappa)
0,447	0,708	0,86	0,7356	0,47

Table 19. Corpus de l'évaluation : métriques de AIA

Comme nous pouvions nous y attendre, nous observons un accord « modéré » (selon l'échelle de Landis & Koch, 1977) entre les deux annotations, qui peut s'expliquer par la fréquence des cas litigieux situés dans la zone grise entre nos deux catégories (une classification ternaire aurait peut-être été plus adaptée). En sélectionnant les annotations communes, nous pouvons estimer que le caractère émergent ou non émergent des nominations est néanmoins clairement établi.

2. Résultat de l'évaluation

Deux expériences sont mises en œuvre : la classification par le classifieur et la prédiction à l'aide de modèle MLM. Vu que ces deux modèles fonctionnent avec des contextes de taille différente, il faut dans un premier temps constituer les jeux de données sous la forme correspondante. Cela consiste à extraire les expressions cibles et à masquer le « pivot » dans les contextes, après quoi, soit on garde le contexte complet, soit on en réduit la taille à 10 tokens. Dans les jeux de données, chaque ligne se compose de :

Expression + tabulation + mot masqué + tabulation + contexte

Par exemple :

```
agriculture biologique --> biologique --> les moyens de l'agriculture {tokenizer.mask_token} .
```

Après avoir passé l'ensemble des données dans ces deux modèles, on obtient les résultats de la classification représentés par les tags 0 ou 1, les rangs de mot à prédire dans la liste de prédiction, le score et la probabilité du mot masqué dans la prédiction[33].

Dans un premier temps, nous avons évalué le résultat de l'identification des contextes spécifiants sur ces nouvelles données. Une évaluation exhaustive étant un travail immense

[33] https://github.com/yumengding/nomination-mergente

(pour des milliers de contextes), cette évaluation est menée sur un échantillon de 150 contextes (soit environ 1% du corpus élargi) sélectionnés de manière aléatoire.

	Précision (calculée sur l'annotation des 1 seulement)	Rappel (calculée sur l'annotation des 1 seulement)	F1%
Échantillon de 150	0,361	0,867	50,975

Table 20. Evaluation d'échantillon

Nous observons une exactitude et un rappel assez bons, mais une précision assez basse sur l'évaluation. Ce résultat peut s'expliquer par deux raisons. Comme les tags 0 sont plus nombreux que les tags 1, la proportion des tags est très différente de celle apprise lors de l'entrainement. Effectivement, selon la vérification manuelle, 15 contextes spécifiants et 135 non spécifiants sont sélectionnés aléatoirement. L'autre raison est peut-être lié à la surdétection de nomination à travers les indices formels : les marques contextuelles surfaciques comme les guillemets ou les parenthèses prennent peut-être un poids trop important dans la décision. Notre classifieur n'est pas encore capable de distinguer les différents usages de ces signes, et se contente de les identifier superficiellement.

Annotation	Prédiction	Phrase
1	1	À noter…développement d'une offre d'***énergie décarbonée*** compétitive **(**éolien, méthanisation, efficacité énergétique, etc.).
0	1	Le pouvoir voulait mettre en avant une ***société civile*** aux multiples talents, **(**...) son vivier ...

Table 21. Exemples des indices en fonctions différentes

Dans les deux phrases du tableau ci-dessus, le classifieur considère vraisemblablement les contextes comme spécifiant en fonction des parenthèses qui suivent. Effectivement, parfois les parenthèses ont une fonction d'explication en listant des exemples afin de détailler le sens, comme dans la première phrase. Mais elles sont ambiguës, car elles permettent également d'introduire des digressions ou bien, comme dans le deuxième exemple, marquer une omission. Ainsi la difficulté à distinguer les fonctions ou les sens de ces marques formelles aboutit à une surdétection des contextes spécifiants.

Du côté quantitatif, nous classifions les expressions en deux groupes et calculons les moyennes de cinq mesures pour chaque groupe.

Afin d'éviter un biais lié à la fréquence des expressions (les expressions les plus fréquentes étant à priori plus facile à prédire), nous avons ignoré les deux expressions les

plus fréquentes du groupe « expression commune » : *sécurité sociale* (2043) et *grande ville* (1391). De la sorte chaque groupe obtient une fréquence moyenne comparable pour les expressions retenues (respectivement 122 et 124 pour les expressions communes et les nominations, en valeurs arrondies). Pour ces deux groupes, on peut ensuite calculer les différentes valeurs moyennes (chaque occurrence recevant une pondération égale - et non chaque expression, certaines étant plus fréquentes que d'autres).

	% de contextes spécifiants	Rang moyen	Proportion de mots non prédits	Score	Probabilité
200 expressions	23,31 %	376,2	0,99 %	8,285	0,062
Nominations émergentes	24,89 %	686,1	2,3 %	6,903	0,051
Expressions communes	23,1 %	335,6	0,8 %	8,391	0,063

Table 22. Evaluation finale de nos critères

Quand on compare les pourcentages de contextes spécifiants, la classe des nominations émergentes obtient un pourcentage très légèrement supérieur à celui du groupe des expressions communes. Il est difficile de conclure face à une différence aussi ténue, d'autant que d'après nos observations, la détection des contextes spécifiants engendre beaucoup de bruit.

En revanche, nous nous apercevons que le groupe des nominations présente une nette augmentation du rang moyen et du pourcentage de mots non prédits. Cet écart, qui va du simple au double, est encore plus marqué que lors de notre première expérimentation.

Cela montre que le groupe des nominations correspond bien au critère de faible prédictibilité, qui découle d'éventuels glissements de sens.

Chapitre 8. Discussions

En analysant l'expérimentation totale et les résultats de l'évaluation final, nous voyons que concernant le classifieur, cette expérimentation permet d'attester nos trois premières hypothèses. Effectivement, ce classifieur neuronal arrive à identifier des indices de surface qui caractérisent le contexte. Il répond totalement à notre attente, à travers l'apprentissage à reconnaitre les indices formels comme les parenthèses, les guillemets et les deux points, etc., il est capable de prédire si l'expression ciblée est une nomination. Par ailleurs, vu une proportion plus haute de contexte spécifiant pour le groupe de nomination, il atteste que ces marques superficielles permettent de les caractériser et marquer dans certains cas.

Quant à l'expérience de l'approche distributionnelle basée sur le modèle MLM, nous trouvons qu'un écart important des résultats de prédiction entre les nominations et la ligne de base à partir des expressions quelconques, notamment sur le rang du mot à prédire dans la liste de prédiction. Vu que les glissements rendent la prédiction plus difficile, et les nominations sont moins prédictif s'expliquant par un rang haut et des score et probabilité bas à prédire, nous pouvons confirmer notre dernière hypothèse : une nomination émergente peut se caractériser par une faible prédictibilité indiquant un changement distributionnel duquel découle un possible glissement sémantique.

Bien que nous réussissions en partie à démontrer nos hypothèses à l'aide d'expériences s'appuyant sur les deux approches textométrique et distributionnelle, nous observons que les performances du classifieur ne sont pas satisfaisantes. Cette lacune s'explique sans doute par le fait qu'il s'appuierait surtout sur des marques superficielles, alors que la distinction spécifiant/non spécifiant engage une activité interprétative approfondie.

Pour avancer sur ce point, il faudrait pouvoir étendre et compléter notre jeu de données. Concernant la double annotation, il faudrait rechercher systématiquement le consensus, et définir des critères fiables, voire des tests, pour départager les cas litigieux. En analysant nos points de désaccords, ce qui permet de trancher entre « spécifiant » et « non spécifiant » est souvent lié à des indices interprétatifs. Ils nécessitent une compréhension globale du sens en fonction des connaissances du monde et des paramètres pragmatiques. Dans ce cas, l'annotation comporte une part d'interprétation subjective qui

est sujette à variation, et même si on a mis en pratique une annotation multiple et des critères rigoureux, on ne peut pas assurer un consensus total sur les décisions.

Ce travail d'explicitation des critères permettrait néanmoins de mieux cerner les propriétés discriminantes de ces contextes, afin de les intégrer dans un système opérationnel, qui pourrait même être symbolique plutôt que neuronal, de façon analogue aux patterns mis en œuvre pour la détection des contextes riches en connaissance (CRC).

Par ailleurs, malgré un prétraitement des données comme le nettoyage des mots collés ou des encodages spéciaux, une mauvaise performance est aussi possible du fait de problèmes d'orthographe ou de grammaire sur la forme à prédire ou dans son contexte immédiat (ces cas sont fréquents dans les corpus de débats).

Du côté du classifieur, même un *transformer* peut trouver ses limites, car il n'a accès qu'à des indices de surface, et traite les contextes comme spécifiant pourvu qu'ils contiennent ces indices. En outre, vu l'écart entre la précision et le rappel, il existe probablement un problème de surapprentissage qui affecte la capacité du classifieur. Nous avons arrêté l'entraînement du modèle en avance en nous appuyant sur l'observation de la courbe de perte, mais du fait de notre manque d'expérience, il se peut que le classifieur ne soit pas enregistré dans le meilleur état, ce qui peut causer et une faible capacité de généralisation.

Conclusion

Cette recherche a pour objectif de répondre à une problématique complexe : comment peut-on détecter automatiquement des nominations émergentes et sur quoi peut-on s'appuyer pour les identifier. Dans une approche préliminaire, nous avons expérimenté un classifieur dédié à l'analyse des indices formels présents dans le contexte des nominations, ainsi qu'un critère de détection s'appuyant sur la faible prédictibilité des nominations en utilisant un modèle de langage à l'état de l'art (celui de FlauBERT).

Globalement, ces deux approches ont été mises en pratique dans nos expérimentations. D'une part, étant donné que certains indices formels dans le voisinage des nominations permettent de les identifier, du fait de l'effort du locuteur d'en expliciter le sens ou d'en signaler la nouveauté, nous avons construit un corpus annoté de 682 occurrences de nominations dont la moitié correspond à des contextes « spécifiants », qui contiennent ces marques surfaciques, l'autre moitié correspondant à des « contextes normaux ou non spécifiants ». En nous basant sur des modèles de réseau de neurones performants, nous avons constitué un classifieur et l'avons entraîné avec les données du corpus afin de le rendre capable de classer les contextes entre spécifiants et non spécifiants. Nous avons aussi testé le classifieur sur un corpus élargi comportant à la fois des nominations émergentes et des expressions communes, mais vu le faible écart sur la proportion de contextes classés comme spécifiants observés entre ces deux groupes, nous penser qu'il est trop tôt, en fonction de nos connaissances actuelles, d'en faire un critère de repérage pour les nominations émergentes. Au préalable, il faudrait pouvoir nettoyer manuellement les résultats de la classification automatique, au moins sur un échantillon, pour se faire une idée de l'écart réel concernant le pourcentage de contextes spécifiants entre ces deux groupes. Nous indiquons cette tâche à titre de perspectives...

D'autre part, au plan sémantique, les nominations émergentes se traduisent par de légers glissements de sens, qui découlent de changements distributionnels. Nous avons ainsi mesuré la prédictibilité des mots cibles, au sein des nominations émergentes, en calculant la moyenne du rang, le score *logit* et la probabilité de prédiction en sortie du modèle préentraîné MLM de FlauBERT. En comparant la prédictibilité des nominations avec celle des expressions quelconques, de structure et de fréquence analogue, nous avons pu ainsi confirmer que les nominations sélectionnées connaissent une faible prédictibilité, indice d'une variation sémantique. La prédictibilité serait donc un critère utile pour la

détection de ces variations, qui constituent un véritable verrou, d'un point de vue plus général, pour la détection de la néologie sémantique.

Lors de la mise en œuvre de notre étude, nous avons rencontré principalement deux difficultés. Tout d'abord, celle de la fiabilité des annotations. Comme l'identification à la main s'appuie sur des connaissances du monde et des interprétations qui peuvent être diverses, il existera toujours des divergences sur ce type d'annotation. Pour garantir la fiabilité de l'annotation autant que possible, il faut sans doute se limiter à une catégorie de phénomène plus restreinte, pour laquelle les marques de surface sont moins ambiguës.

La deuxième difficulté concerne l'entraînement et l'amélioration du classifieur. Au début, cela implique de transformer les mots en représentations vectorielles à l'aide des *embeddings* contextuels. Comme la forme de l'entrée influence cette procédure d'encodage, en particulier la taille du contexte retenu, nous nous interrogeons ainsi sur le choix de l'entrée initiale.

En outre, nous trouvons que le jeu de données pour l'entraînement est vraisemblablement trop petit, ce qui ne permet pas de couvrir tous les indices pertinents. Cela peut causer des difficultés dans l'extraction des traits pertinents, et se traduire par un problème de surapprentissage.

Même si certains paramètres sont initialisés par le modèle, pour améliorer la performance du classifieur dans la situation actuelle, il faut les modifier au fil des expériences. Plusieurs choix sont disponibles pour chacun et les paramètres sont interdépendants pour le modèle. Pour nous, la sélection d'une bonne combinaison de paramètres s'est révélée difficile, car les combinaisons possibles étaient trop nombreuses.

Pour résumer, notre étude permet de mieux comprendre les difficultés liées à la détection de nominations émergentes en exploitant des modèles performants tels que FlauBERT. Elle combine l'approche textométrique, l'approche distributionnelle et les réseaux de neurones et permet de proposer un outil, le classifieur, encore perfectible, et des critères quantitatifs pour caractériser les nominations émergentes, dans la perspective d'aider à leur repérage.

Bibliographie

Baroni, M., Dinu, G. et Kruszewski, G. (2014). Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* https://aclanthology.org/P14-1023.pdf

Calabrese, L. (2015). Reformulation et non-reformulation du mot islamophobie. Une analyse des dynamiques de la nomination dans les commentaires des lecteurs. *Langue française*, 188.

Cassier, M. (2020). Exploitation de modèles distributionnels pour l'étude de la nomination dans un corpus d'interviews politiques. *Actes de JEP TALN 2020*, https://hal.archives-ouvertes.fr/hal-02786188v3

De Bessé, B. (1991). Le contexte terminographique. *Meta, 36 (1)*, 111–120.

Détrie, C., Siblot, P. et Vérine, B. (2001). *Termes et concepts pour l'analyse du discours.* Paris : Champion.

Devlin, J., Chang, M-W., Lee, K. et Toutanova, K. (2018). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*. https://arxiv.org/abs/1810.04805

Diwersy, S., Goossens, V., Grutschus, A., Kern, B., Kraif, O., Melnikova, E. et Novakova, I. (2014). Traitement des lexies d'émotion dans les corpus et les applications d'EmoBase. *Corpus (13)*, 269-293. https://journals.openedition.org/corpus/2537

E. Peters, M., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K. et Zettlemoyer, L. (2018). Deep contextualized word representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*. https://arxiv.org/pdf/1802.05365.pdf

Evert, S. (2007). *Corpora and collocations.* Extended Manuscript of Chapter 58 of Lüdeling A. et Kytö M., 2008, *Corpus Linguistics. An International Handbook*. Berlin: Mouton de Gruyter.

Frath, P. (2015). Dénomination référentielle, désignation, nomination. *Langue française, 188*, 33-46. DOI: 10.3917/lf.188.0033.

Glazkova, A. (2008). A Comparison of Synthetic Oversampling Methods for Multi-class Text Classification. https://arxiv.org/abs/2008.04636

Gonon, L., Goossens, V., Kraif, O., Novakova, I. et Sorba, J. (2018). Motifs textuels spécifiques au genre policier et à la littérature « blanche ». *Actes de CMLF 2018*, https://www.shs-conferences.org/articles/shsconf/pdf/2018/07/shsconf_cmlf2018_06007.pdf

Guillaume, B. et Perrier, G. (2017). Reflexion sur l'annotation de corpus ecrits du français en syntaxe et en semantique. *ACor4French – Les corpus annotés du français, Jun 2017, Orléans, France.* pp.1-8. https://hal.inria.fr/hal-01651753

Hmida, F. (2014). Identification de Contextes Riches en Connaissances en corpus comparable. *21ème Traitement Automatique des Langues Naturelles*, 112-123.

Hmida, F., Morin, E. et Daille, B. (2015). Extraction de Contextes Riches en *Connaissances en corpus spécialisés. 22ème Traitement Automatique des Langues Naturelles*, 109-115.

Jackiewicz, A et Pengam, M. (2020). Un modèle pour l'étude des nominations émergentes. Notion de repérage pour saisir les modalités d'ajustement sémantique et discursif, *CMLF 2020*. DOI : https://doi.org/10.1051/shsconf/20207812004

Kraif, O. (2008). Comment allier la puissance du TAL et la simplicité d'utilisation ? l'exemple du concordancier bilingue ConcQuest. *JADT 2008, PUL, 625-634, vol. 2.* https://hal.archives-ouvertes.fr/hal-01073703/document

Kraif, O. et Diwersy, S. (2012). Le Lexicoscope : un outil pour l'étude de profls combinatoires et l'extraction de constructions lexico-syntaxiques. *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2: TALN*, 399-406. https://aclanthology.org/F12-2033.pdf

Kraif, O., Tutin, A. et Diwersy, S. (2014). Extraction de pivots complexes pour l'exploration de la combinatoire du lexique : une étude dans le champ des noms d'affect. *Conference : 4e Congrès Mondial de Linguistique FrançaiseAt : BerlinVolume : 8*, 2663-2674. DOI : 10.1051/shsconf/20140801362

Kraif, O. et Tutin, A. (2017). Des motifs séquentiels aux motifs hiérarchiques : l'apport des arbres lexico-syntaxiques récurrents pour le repérage des routines discursives. *Bases, Corpus, Langage*. DOI : https://doi.org/10.4000/corpus.2889

Landis, J.R. et Koch, G.G. (1977). The Measurement of Observer Agreement for Categorical Data. *Vol. 33, No. 1 (Mar., 1977),* pp. 159-174. Biometric Society. https://www.jstor.org/stable/2529310?origin=crossref

Le, H., Vial, L., Frej, J., Segonne, V., Coavoux, M., Lecouteux, B., Allauzen, A., Crabbé, B., Besacier, L. et Schwab, D. (2019). FlauBERT: Unsupervised Language Model Pre-training for French. *Proceedings of the 12th Language Resources and Evaluation Conference*. https://arxiv.org/abs/1912.05372

Lefeuvre, L. et Condamines, A. (2017). MAR-REL : une base de marqueurs de relations conceptuelles pour la détection de Contextes Riches en Connaissances. *Actes des 24ème Conférence sur le Traitement Automatique des Langues Naturelles. Volume 2 - Articles courts.* https://aclanthology.org/2017.jeptalnrecital-court.23.pdf

Longhi, J (dir.) 2015. Stabilité et instabilité dans la production du sens : la nomination en discours. *Langue française*, 188. DOI : https://doi.org/10.3917/lf.188.0005

Longhi, J., Antoine, J-Y., Mirzapour, M., Jackiewicz, A. et Lefeuvre-Halftermeyer, A. (2020). Le repérage de nominations dans les corpus textuels : de l'exploitation de l'analyse des données textuelles à l'exploration des chaînes de coréférence par le TAL. *JADT 2020 : 15es Journées internationales d'Analyse statistique des Données Textuelles*. https://hal.archives-ouvertes.fr/hal-02906925

Mikolov, T., Chen, K., Corrado, G. et Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. *ICLR (Workshop Poster) 2013*. https://arxiv.org/pdf/1301.3781.pdf

Mikolov, T., V. Le, Q et Sutskever, I. (2013). Exploiting Similarities among Languages for Machine Translation. *CoRR abs/1309.4168 (2013)*. https://arxiv.org/pdf/1309.4168.pdf

Pengam, M. et Jackiewicz, A. (2019). Sens et emplois de l'expression « Musulmans Modérés » dans les discours médiatiques. *Open Library of Humanities*. DOI: https://doi.org/10.16995/olh.431

Picton, A. (2009). Marqueurs et contextes riches en connaissances pour observer l'évolution en diachronie courte : Éléments méthodologiques en corpus. *Conference: Language for Special Purposes (LSP 2009)*. http://docplayer.fr/159200487-Aurelie-picton-1.html

Qi, P., Zhang, Y., Zhang, Y., Bolton. J. et D. Manning. C. (2020). Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations.* https://arxiv.org/pdf/2003.07082.pdf

Sablayrolles, J-F. (2007). Nomination, dénomination et néologie : intersection et différences symétriques. *Neologica*, 1, 87-99.

Siblot, P. (2001). De la dénomination à la nomination. *Cahiers de praxématique*, 36. https://journals.openedition.org/praxematique/368

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A, N., Kaiser, L. et Polosukhin, I. (2017). Attention is all you need. *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing SystemsDecember 2017* Pages 6000–6010.
https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

Sitographie

FlauBERT. https://huggingface.co/transformers/model_doc/flaubert.html [Dernière consultation le 15/08/2021]

Lexicoscope2.0. http://phraseotext.univ-grenoble-alpes.fr/lexicoscope_2.0/ [Dernière consultation le 11/08/2021]

LIDILEM. https://lidilem.univ-grenoble-alpes.fr/ [Dernière consultation le 22/06/2021]

MIAI. https://miai.univ-grenoble-alpes.fr/ [Dernière consultation le 22/06/2021]

Pdfminer 20191125 – PyPI. https://pypi.org/project/pdfminer/ [Dernière consultation le 13/05/2021]

Pytorchtools-PYPI. https://pypi.org/project/pytorchtools/ [Dernière consultation le 21/07/2021]

Selenium with Python. https://selenium-python.readthedocs.io/ [Dernière consultation le 13/05/2021]

Stanza. https://stanfordnlp.github.io/stanza/ [Dernière consultation le 10/07/2021]

Glossaire

Apprentissage profond : Ensemble de méthodes d'apprentissage automatique tentant de modéliser avec un haut niveau d'abstraction des données grâce à des architectures articulées de différentes transformations non linéaires(https://fr.wikipedia.org/wiki/Apprentissage_profond)

Contexte : Le contexte d'un événement inclut les circonstances et conditions qui l'entourent (https://fr.wikipedia.org/wiki/Contexte)

Coréférence : Plusieurs syntagmes nominaux différents dans une phrase ou dans un discours désignent la même entité (https://fr.wikipedia.org/wiki/Cor%C3%A9f%C3%A9rence)

Corpus : Ensemble de documents, artistiques ou non (textes, images, vidéos, etc.), regroupés dans une optique précise (https://fr.wikipedia.org/wiki/Corpus)

Dépendance syntaxique : Le fait que la présence d'un mot et tous ses dépendants est légitimée par un autre mot (http://alpage.inria.fr/statgram/frdep/Publications/FTBGuideDepSurface.pdf)

Désambiguïsation lexicale : Détermination du sens d'un mot dans une phrase lorsque ce mot peut avoir plusieurs sens possibles(https://fr.wikipedia.org/wiki/D%C3%A9sambigu%C3%AFsation_lexicale)

Glissement de sens : Un mot ou une expression acquiert au fil du temps un sens différent de celui d'origine (https://fr.wikipedia.org/wiki/Glissement_s%C3%A9mantique)

Lemmatisation : Traitement lexical à appliquer aux occurrences des lexèmes sujets à flexion un codage renvoyant à leur entrée lexicale commune (https://fr.wikipedia.org/wiki/Lemmatisation)

Part of speech : La nature d'un mot en grammaire (https://en.wikipedia.org/wiki/Part_of_speech)

Sémantique : Etudier les signifiés, ce dont on parle, ce que l'on transmet par un énoncé(https://fr.wikipedia.org/wiki/S%C3%A9mantique)

Syntaxe : Etudier la manière dont les mots se combinent pour former des phrases ou des énoncés dans une langue (https://fr.wikipedia.org/wiki/Syntaxe)

Textométrie : Elle propose des procédures de tris et de calculs statistiques pour l'étude d'un corpus de textes numérisés (https://journals.openedition.org/corpus/2121)

Token : Mot anglais signifiant jeton (https://fr.wikipedia.org/wiki/Token)

Sigles et abréviations utilisés

AD : Analyse du discours

ALR : Arbre lexico-syntaxique récurrent

AIA : Accord Inter-Annotateur

BERT : Bidirectional Encoder Representation from Transformers

Bi-LSTM: Bi-directional long short term memory

CBOW: Continuous Bag-of-Words Model

CHD : Classification hiérarchique descendante des contexts

CRC : Contextes Riches en Connaissances

ELMO: Embedding from Language Models

FNN: Feedforward Neural Network

IraMuTeq : Interface de R pour les Analyses Multidimensionnelles de Textes et de Questionnaires

LIDILEM : Laboratoire de linguistique et didactique des langues étrangères et maternelles

LIG : Laboratoire d'informatique de Grenoble

MIAI : Multidisciplinary Institute in Artificial Intelligence

MLM: Mask Language Model

POS: Part of speech

SQuAD: Stanford Question Answering Dataset

TAL : Traitement Automatique des Langues

TALAD : Traitement Automatique des Langues et Analyse du Discours

TEI : Text Encoding Initiative

UD : Universal Dependencies

Wandb : Weights & Biases

XML : Extensible Markup Language

Table des illustrations

Figure 1. Transformation en dénomination (Pengam, Jackiewicz, 2019) 12
Figure 2. Réseau des relateurs de repérage (Jackiewicz, Pengam, 2020) 23
Figure 3. Requête TQL de "vélo partagé" : <l=vélo,c=NOUN,#1>&&<l=partager,c=VERB,#2>::(NMOD_POSIT,1,2)..... 27
Figure 4. Extraction de l'ALR <proposer+dans+ce+article> (Kraif et Tutin, 2017) .. 28
Figure 5. Extraction de l'ALR <ville verte> 28
Figure 6. Modèles CBOW et Skip-gram (Mikolov, V. Le, Sutskever, 2013)..... 33
Figure 7. ELMO : deux couches de Bi-LSTM (cf. https://www.topbots.com/generalized-language-models-cove-elmo/, 2019)..... 35
Figure 8. Trois embeddings (Devlin, Chang, Lee, Toutanova, 2018)..... 36
Figure 9. Modèles de pré-entraînement (Devlin, Chang, Lee, Toutanova, 2018)..... 37
Figure 10. Structure d'un fichier XML 46
Figure 11. Pipeline de Stanza (Qi et al., 2020)..... 47
Figure 12. Exemple UD (https://universaldependencies.org/fr/dep/amod.html) 48
Figure 13. Comparaison les spécialités avec les outils populaires (Qi et al., 2003).... 49
Figure 14. Evaluation des modèles (Qi et al., 2020) 49
Figure 15. Listes de pivots..... 51
Figure 16. Extrait du recueil d'expressions..... 51
Figure 17. Exemples de marques définissant les contextes spécifiant 52
Figure 18. Architecture d'un FNN (Mukul Rathi, 2018)..... 59
Figure 19. *Loss* de TRAIN et DEV avec des taux d'apprentissage différents 60
Figure 20. Exactitude de TRAIN et DEV avec des taux d'apprentissage différents... 61
Figure 21. Contexte complet vs contexte de 10 : Loss de TRAIN et DEV..... 61
Figure 22. Contexte complet vs contexte de 10 : Exactitude de TRAIN et DEV 61
Figure 23. Les combinaisons d'*embeddings* contextuels pour la tâche NER (Jacob Devlin et al., 2018) 63
Figure 24. Loss de TRAIN et DEV sur les *embeddings* contextuels différents 63
Figure 25. Exactitude de TRAIN et DEV sur les *embeddings* contextuels différents. 63
Figure 26. *Loss* de TRAIN et DEV pour diverses combinaisons de *batch size* et de taux d'apprentissage 66
Figure 27. Exactitude de TRAIN et DEV pour des combinaisons diverses de *batch size* et de taux d'apprentissage 66
Figure 28. Contextes complets vs 10 contextes : Loss de TRAIN et DEV..... 67
Figure 29. Contextes complets vs 10 contextes : Exactitude de TRAIN et DEV 67
Figure 30. Extrait de résultats de nominations sélectionnés..... 70
Figure 31. Extrait d'une annotation double..... 74

Table 1. L'approche textométrique 21
Table 2. L'approche du traitement « manuel »..... 21
Table 3. L'approche de repérage par la détection des chaînes de coréférence..... 22
Table 4. L'approche de repérage par relations statiques 24
Table 5. L'approche distributionnelle pour la nomination..... 31
Table 6. Composants du corpus..... 45
Table 7. Chiffres du corpus 45
Table 8. Accord des deux annotations sur le corpus de nomination 54
Table 9. Résultats de métriques de AIA sur le corpus de nomination 54

Table 10. Jeux de données pour les classifieurs 55
Table 11. Jeux de données pour le modèle MLM 56
Table 12. Contexte complet vs 10 contextes : Scores du TEST 62
Table 13. Scores du TEST des trois *embeddings* contextuels 64
Table 14. Contexte complet vs 10 contextes : Scores du TEST 68
Table 15. Classifieur FNN vs Classifieur BERT : Scores du TEST 68
Table 16. Résultats de la prédiction pour les nominations et pour les constructions quelconques 70
Table 17. Contexte complet vs contexte de 10 : résultats de prédiction 71
Table 18. Corpus de l'évaluation : Accord Inter-Annotateur des deux annotations ... 75
Table 19. Corpus de l'évaluation : métriques de AIA 75
Table 20. Evaluation d'échantillon 76
Table 21. Exemples des indices en fonctions différentes 76
Table 22. Evaluation finale de nos critères 77

Table des annexes

Annexe 1 Extrait des marqueurs de nominations 91
Annexe 2 Extrait des résultats de l'annotation double et de la prédiction 92
Annexe 3 Extrait des résultats de l'évaluation finale 93

Annexe 1
Extrait des marqueurs de nominations

Marqueur
il convient d'être clair
il faut être clair
je ne peux pas être plus clair
pour parler sommairement
soit dit en passant
à tous les sens du mot
à tous les sens du terme
au sens
au sens figuré
au sens propre
au sens propre comme au sens figuré
au sens strict
au sens strict du mot
au sens strict du terme
aux deux sens
aux deux sens de ce terme
aux deux sens du mot
ce qu'il faut appeler
ce qu'on peut appeler
ce qu'on pourrait appeler
c'est le cas de le dire
c'est le mot
c'est le mot !
c'est le mot exact
c'est le mot juste
c'est le mot qui convient
comme on dit au rugby
comme vous aimez à le dire
comme vous venez de le dire
comment vous dites déjà
dans les deux sens du mot
dans les deux sens du terme
dans tous les sens du mot
dans tous les sens du terme
devrais-je dire
dirais-je
entre guillemets
et je dis bien
il faudra dire les choses telles qu'elles sont

Annexe 2
Extrait des résultats de l'annotation double et de la prédiction

annotation Yumeng	annotation Agata	type de construction	rang de mot masqué	score 11.3477372457	probabilité 0.1321698272	expression intéressante	mot attendu	contexte de phrase
1	0	nom+adj	2	17.05054473876953	0.009500052779912949	développement durable	durable	Il devisait avec les puissa
1	1	nom+adj	1	19.06382179260254	0.7018222212791443	développement durable	durable	En 1987, le développem
1	1	nom+adj	1	17.849231719970703	0.8467938899993896	développement durable	durable	Il faudrait d'abord que l'
1	1	nom+adj	3564	2.447561740875244	4.976795935363043e-06	développement durable	durable	développement {tokeniz
1	1	nom+adj	1	20.48317527770996	0.9754701256752014	développement durable	durable	Le développement {toke
1	1	nom+adj	1	14.093353271484375	0.19185274839401245	développement durable	durable	Mais le développement
1	1	nom+adj	2	15.705249786376953	0.23718801140785217	développement durable	durable	On nous somme de réd
1	0	nom+adj	2	17.196304321289062	0.09697233140468597	développement durable	durable	Dans les programmes a
1	1	nom+adj	1	20.30354881286621	0.9685400724411011	développement durable	durable	Après trois ans d'études
0	0	nom+adj	1	18.895036697387695	0.7981733679771423	développement durable	durable	L'essentiel porte sur l'éd
0	0	nom+adj	1	20.719627380371094	0.9628145098686218	développement durable	durable	Les analyses et préconis
0	0	nom+adj	1	19.67886734008789	0.580017626285553	développement durable	durable	Celle d'une « création m
0	1	nom+adj	1	17.740825653076172	0.5839037299156189	développement durable	durable	Le parler vrai est de bier
0	0	nom+adj	1	24.651039123535156	0.9966649413108826	développement durable	durable	L'éducation a fait aussi l
0	0	nom+adj	1	21.606760025024414	0.8243820071220398	développement durable	durable	Mais les entreprises de c
0	0	nom+adj	1	23.981224060058594	0.996802568435669	développement durable	durable	Cette éducation à l'envir
0	0	nom+adj	1	20.51453399658203	0.8632386326789856	développement durable	durable	Le réchauffement climat
0	1	nom+adj	1	18.860057830810547	0.9839094281196594	développement durable	durable	inscrire les négociations
1	0	nom+adj	4	11.526856422424316	0.020402777940034866	économie « durables	durables	Le directeur de l'Agence

Annexe 3
Extrait des résultats de l'évaluation finale

Nomination	Expression	total (14935)	prédiction 0	prédiction 1	%contexte sp	rang moyen=72.860094	nb non prédit	scc
	<l=sécurité,c=NOUN,#1>&&<l=social,c=ADJ,#co>::(NMOD_POSIT1,1,co)	445	276	169	0.379775280	3.2112359550561798	0	7.8
	<l=ville,c=NOUN,#1>&&<l=grand,c=ADJ,#co>::(NMOD_POSIT1,1,co)	702	556	146	0.207977207	18.83048433048433	0	12.
	<l=énergie,c=NOUN,#1>&&<l=renouvelable,c=ADJ,#co>::(NMOD_POSIT1,1,co)	422	326	96	0.227488151	46.26066350710901	0	8.3
	<l=société,c=NOUN,#1>&&<l=civil,c=ADJ,#co>::(NMOD_POSIT1,1,co)	683	522	161	0.235724743	28.792093704245975	0	8.6
	<l=monde,c=NOUN,#1>&&<l=entier,c=ADJ,#co>::(NMOD_POSIT1,1,co)	845	710	135	0.159763313	6.491124260355029	0	9.8
	<l=énergie,c=NOUN,#1>&&<l=fossile,c=ADJ,#co>::(NMOD_POSIT1,1,co)	341	275	66	0.193548387	246.1524926686217	0	7.6
	<l=finance,c=NOUN,#1>&&<l=public,c=ADJ,#co>::(NMOD_POSIT1,1,co)	384	313	71	0.184895833	9.4765625	0	9.5
	<l=ville,c=NOUN,#1>&&<l=petit,c=ADJ,#co>::(NMOD_POSIT1,1,co)	405	280	125	0.308641975	2.246913580246914	0	11.
	<l=transport,c=NOUN,#1>&&<l=public,c=ADJ,#co>::(NMOD_POSIT1,1,co)	208	164	44	0.211538461	41.32211538461539	0	7.5
x	<l=développement,c=NOUN,#1>&&<l=durable,c=ADJ,#co>::(NMOD_POSIT1,1,co)	409	323	86	0.210268948	32.691931540342296	0	7.9
	<l=sécurité,c=NOUN,#1>&&<l=national,c=ADJ,#co>::(NMOD_POSIT1,1,co)	518	308	210	0.405405405	5.06949806949807	0	9.9
	<l=sécurité,c=NOUN,#1>&&<l=routier,c=ADJ,#co>::(NMOD_POSIT1,1,co)	71	53	18	0.253521126	1445.5633802816901	0	6.1
	<l=modèle,c=NOUN,#1>&&<l=économique,c=ADJ,#co>::(NMOD_POSIT1,1,co)	390	314	76	0.194871794	50.84615384615385	0	9.0
	<l=société,c=NOUN,#1>&&<l=français,c=ADJ,#co>::(NMOD_POSIT1,1,co)	268	209	59	0.220149253	66.32462686567165	0	7.7
	<l=développement,c=NOUN,#1>&&<l=économique,c=ADJ,#co>::(NMOD_POSIT1,1,co)	236	196	40	0.169491525	20.440677966101696	0	9.2
	<l=projet,c=NOUN,#1>&&<l=grand,c=ADJ,#co>::(NMOD_POSIT1,1,co)	186	137	49	0.263440860	4.241935483870968	0	10.
	<l=transport,c=NOUN,#1>&&<l=aérien,c=ADJ,#co>::(NMOD_POSIT1,1,co)	196	128	68	0.346938775	72.85204081632654	0	6.2
	<l=agriculture,c=NOUN,#1>&&<l=biologique,c=ADJ,#co>::(NMOD_POSIT1,1,co)	108	72	36	0.333333333	12.537037037037036	0	5.1
	<l=économie,c=NOUN,#1>&&<l=mondial,c=ADJ,#co>::(NMOD_POSIT1,1,co)	258	203	55	0.213178294	203.04263565891472	0	8.7

Table des matières

Remerciements 2

Sommaire 4

Introduction 6

Partie 1 - Contexte et Problématique 10

CHAPITRE 1. CONTEXTE ET PROBLEMATIQUE 11

1. CONTEXTE 11
2. PROBLEMATIQUE 12

Partie 2 - État de l'art 18

CHAPITRE 2. MODELISATION 19

1. APPROCHES POUR LE REPERAGE DES NOMINATIONS 19
2. METHODOLOGIE DE REPERAGE ET D'ANALYSE DE LA NOMINATION EMERGENTE . 22
3. LES METHODES TEXTOMETRIQUES 25
4. APPROCHES DISTRIBUTIONNELLES POUR LE GLISSEMENT SEMANTIQUE 29

Partie 3 - Expérimentation basée sur l'apprentissage profond 40

CHAPITRE 3. CONSTRUCTION DU CORPUS 41

1. RECOLTE DES DONNEES 42
2. CORPUS FINAL 45
3. ANNOTATION DU CORPUS 46

CHAPITRE 4. JEUX DE DONNEES 50

1. JEUX DE DONNEES POUR LES CLASSIFIEURS 50
2. JEUX DE DONNEES POUR LE MODELE MLM 55

CHAPITRE 5. CONSTRUCTION ET ENTRAINEMENT DU CLASSIFIEUR 57

1. JEUX DE DONNEES 57
2. CLASSIFIEUR BASE SUR LE FNN 58
3. CLASSIFIEUR BASE SUR LE FLAUBERT 65
4. CLASSIFIEUR FINAL 68

CHAPITRE 6. MODELE MLM DE BERT 69

1. EXPERIENCES 69
2. RESULTATS 70

Partie 4 - Résultats et discussions 72

CHAPITRE 7. ELARGISSEMENT DES DONNEES 73

1. CONSTRUCTION DU CORPUS ELARGI 73
2. RESULTAT DE L'EVALUATION 75
CHAPITRE 8. DISCUSSIONS 78
Conclusion 80
Bibliographie 82
Sitographie 85
Glossaire 86
Sigles et abréviations utilisés 87
Table des illustrations 88
Table des annexes 90
Table des matières 94

MOTS-CLÉS : nominations émergentes, contextes spécifiants, corpus, TAL, apprentissage profond

RÉSUMÉ

Les nominations émergentes permettent de désigner de nouveaux concepts à travers l'usage de nouveaux mots ou de mots dont le contenu sémantique a été légèrement modifié. Elles caractérisent un état précurseur de la néologie, dans la mesure où elles mettent en circulation des expressions non encore stabilisées par l'usage. Elles présentent un intérêt dans le domaine de l'analyse de discours, car elles permettent de véhiculer de manière implicite le point de vue et les valeurs du locuteur, notamment dans le domaine des sujets de société qui font débat (p.ex. avec des nominations telles que agriculture raisonnée, mobilité douce ou immigration choisie). En observant les nominations en corpus, on observe qu'elles se caractérisent parfois par des indices contextuels et formels, qui pourraient être exploités dans des outils d'aide à la détection. Poursuivant cette hypothèse, ce travail est centré sur l'exploitation de modèles d'apprentissage profond pour détecter de manière automatique des critères utiles à l'identification nominations émergentes.

Nous proposons une étude empirique basée sur un corpus que nous avons constitué pour la période 2018-2020, incluant des débats citoyens, des articles de presse, et des rapports dans le domaine de l'écologie. Dans un premier temps, ce mémoire propose d'étudier une approche basée sur la détection de contextes dits spécifiants, visant à marquer un certain positionnement métadiscursif du locuteur, pour caractériser les nominations émergentes. Dans un second temps, nous nous intéressons à un second critère visant à repérer les glissements de sens de certains mots en fonctions de la difficulté à les prédire selon un modèle MLM basé sur FlauBERT. Ce deuxième critère apparait au final comme le plus facile à mettre en œuvre - notre classifieur entrainé pour reconnaitre les contextes spécifiants générant à ce stade trop de bruit pour constituer un indice fiable.

KEYWORDS: emerging nominations, specific contexts, corpus, NLP, deep learning

ABSTRACT

Emerging nominations are used to express the new concepts via new words or the words with semantic changes. They are taken for the precursor of neology insofar as they are not yet stabilized for use. Interested in the field of discourse analysis, they can convey implicitly the speaker's view and values, particularly about the social issues (e.g. integrated farming, sustainable mobility or selective immigration). By noticing the nominations in corpus, we observe that they are sometimes described by the contextual and formal marks, which could be exploited in detection. Base on this hypothesis, this work focuses on exploiting the deep learning models to automatically detect the criteria useful for the recognition of emerging nominations.

We propose a research base on a corpus which collect debates, press articles, and reports in the field of ecology for the period of 2018-2020. Firstly, we propose a study of detecting the specific contexts in order to pinpoint the emerging nominations. Secondly, we are interested in the FlauBERT's MLM model for detecting the semantic changes via a difficulty predicting. In the end of work, the second criterion appears as the easiest to implement because our classifier produces too much noise.