Classification sémantique de sens nominaux en français avec des modèles de langues pré-entraînés: Enrichir une ressource lexicale

Nicolas Angleraud
Sous la direction de Lucie Barque et Marie Candito

Université Paris Cité
Laboratoire de Linguistique Formelle

2023 - 2024

Table des matières

1 Introduction

Ce mémoire a pour objet l'enrichissement sémantique d'une ressource lexicale du français. Le français ne dispose pas, au contraire de l'anglais, d'une ressource couvrante et hiérarchisée comme WordNet (Miller et al., 1990), qui permet d'effectuer des généralisations sémantiques sur le lexique. L'objectif de ce travail est donc de mettre à disposition de la communauté francophone une ressource permettant le même type de généralisations que celles que permet WordNet pour l'anglais, notamment grâce aux *Unique Beginners* qui structurent sémantiquement ses sous-hiérarchies.

Dans ce travail, nous nous concentrons uniquement sur l'enrichissement des descriptions de sens nominaux dans le Wiktionnaire, en attribuant une classe sémantique générale de type "supersens" (quelques dizaines de classes sémantiques, comme Person, Act, Food...) à chaque entrée décrivant un sens nominal dans cette ressource. Pour cela, nous entraînerons un classifieur supervisé à prédire un supersens et l'appliquerons à tous les sens du Wiktionnaire du français. Ce type d'enrichissement pourrait se révéler utile à la fois pour la recherche en linguistique, notamment pour des considérations de sémantique lexicale et de fréquence de type sémantique en lexique, et pour la recherche en traitement automatique des langues, notamment en fournissant plus de données sur les sens rares.

Dans la suite du mémoire, nous introduisons l'utilisation de classes sémantiques à grain épais, aussi appelés supersens, ainsi que la classification de texte en traitement automatique des langues et dressons un panorama de la littérature sur leur utilisation (section 2). Nous présentons ensuite, d'une part le lexique utilisé pour la recherche, le Wiktionnaire, ainsi que le corpus FrSemCor, et d'autre part le jeu de classes sémantiques utilisé pour la classification qui a pour origine les classes lexicographiques de WordNet et des remaniements de ce jeu effectués notamment pour l'annotation du corpus FrSemCor puis la présente étude (section 3). Nous expliquons après les architectures des classifieurs que nous avons testées afin d'entraîner le meilleur possible (section 4). Nous décrivons alors le protocole expérimental pour l'entraînement et l'évaluation des classifieurs d'une part, et d'autre part nous analysons les résultats obtenus lors des expériences (section 5). Une fois que nous avons décrit ce qui précède, nous présentons la ressource, de son extraction à son contenu avant puis après enrichissement sémantique avec notre meilleur classifieur de sens lexical, et discutons d'idées pour exploi-

ter une telle ressource en linguistique et traitement automatique des langues (section 6). Enfin, nous concluons sur le travail qui a été fait et sur des idées pour de futurs recherches qui pourraient s'inscrire dans la continuité (section 7).

2 Littérature

2.1 Aux origines des "supersens" : le "supersense tagging"

Les supersens ont été définis au départ comme classes lexicographiques à gros grain, permettant d'organiser le travail lexicographique de construction du thésaurus WordNet, pour l'anglais (Miller et al., 1990), très célèbre.

Enormément de travaux en TAL pour l'anglais ont utilisé cette ressource, ainsi que le corpus SemCor, un corpus d'environ 360 000 mots, en anglais, où toutes les occurrences couvertes par WordNet ont été manuellement annotées avec le sens WordNet.

(Ciaramita and Johnson, 2003) ont été ensuite les premiers à réutiliser ces classes, pour étiqueter des noms en corpus (travail étendu ensuite aux verbes (Ciaramita and Altun, 2006)). Cette tâche d'étiquetage en supersens (en anglais "supersense tagging") était considérée à l'époque comme préalable utile pour des tâches plus sophistiquées faisant intervenir la sémantique (recherche documentaire, résolution de coréférence, réponse à des questions ...). En effet, les supersens fournissent une désambiguisation partielle des noms en contexte, tout en étant plus faciles à obtenir qu'une annotation complète en sens. En outre, la granularité plus grossière des supersens est parfois présentée comme bénéfique (Ciaramita and Altun, 2006), en tous cas par rapport à la granularité à l'inverse trop fine des sens de WordNet.

2.2 Utilisation de supersens en TAL

La tâche d'étiquetage en supersens a ensuite suivi les évolutions du domaine. On peut citer par exemple (Basile, 2013), qui aborde la tâche en utilisant des machines à vecteur support ("support vector machines" en anglais), au lieu d'un perceptron moyenné.

L'évolution vers l'utilisation de représentations vectorielles de mots, au début des années 2010, a aussi amené à construire des représentations vectorielles de *sens de mots*, pour éviter de manipuler des représentations qui mélangent les différents sens d'un mot. On peut citer (Flekova and Gurevych, 2016), qui construisent un espace vectoriel sémantique, intégrant dans

le même espace des vecteurs de mots (des formes fléchies), de supersens, et de couples mot-supersens. Le tout est construit pour l'anglais, en réutilisant un algorithme de construction de vecteurs de mots (word2vec skip-gram (Mikolov et al., 2013)), simplement en utilisant la concaténation de trois corpus d'apprentissage : le corpus de départ contenant des formes fléchies, le même corpus où les formes sont suffixées par leur supersens prédit, et le même corpus où les formes sont remplacées par leur supersens. La prédiction préalable des supersens est faite par un classifieur appris sur le corpus SemCor.

Le TAL a ensuite évolué vers l'utilisation de vecteurs de mots en contexte, avec notamment l'arrivée du modèle BERT (Devlin et al., 2019). Il s'agit d'un modèle de langue pré-entraîné sur gros corpus, de manière auto-supervisée : c'est un apprentissage supervisé, mais en utilisant des exemples créés automatiquement et trivialement. Ici les exemples sont du type "phrase à trou", la tâche principale pour le pré-entraînement est de prédire un mot étant donné son contexte. Une fois le pré-entraînement fait, on peut réutiliser le modèle pour obtenir des vecteurs d'occurrences de mots. Ce cadre général a été adapté par Levine et al. (2020) pour définir SenseBERT, un modèle capable non seulement de fournir un vecteur de mot en contexte mais aussi un vecteur de supersens en contexte. SenseBERT est appris non seulement sur la tâche de prédiction d'un mot en contexte, mais aussi la tâche de prédire le supersens en contexte. Plus précisément, les auteurs n'utilisent pas un corpus annoté en supersens, mais seulement **un lexique fournissant les supersens possibles d'un mot**. Lors du pré-entraînement, la perte utilisée favorise l'ensemble des supersens possibles du mot courant, au détriment de tous les autres supersens. Plus précisément, la perte utilisée est moins la somme des log-probabilités de tous les supersens possibles du mot. Les auteurs montrent que SenseBERT permet d'améliorer en retour l'étiquetage en supersens, et aussi certaines des tâches du benchmark GLUE.

Aloui et al. (2020) remarquent que les vecteurs contextuels tels qu'issus de BERT sont difficilement interprétables. Ils proposent le modèle SLICE, qui construit un espace vectoriel de mots hors et en contexte, dont les dimensions sont les supersens, et donc sont interprétables.

2.3 Classification automatique de texte

Pour augmenter le Wiktionnaire avec un supersens pour chaque sens, techniquement nous utilisons un classifieur supervisé qui prend en entrée une

séquence de mots (une définition ou un exemple en contexte) et prédit une classe. On se place dont dans le cadre général de la classification supervisée de séquences de mots. Les modèles de langue pré-entraînés dont BERT cité supra ont permis des améliorations de performance subtantielles pour ce type de tâche. Par exemple, pour la tâche "d'analyse de sentiments" qui étant donné un verbatim, prédit une étiquette de sentiment positif, négatif ou neutre, le modèle BERT (Devlin et al., 2019) fournit le nouvel état de l'art. La technique utilisée est d'intégrer au classifieur de séquences de mots le modèle de langue pré-entraîné : la séquence est d'abord passée dans le modèle de langue pour récupérer un vecteur contextuel par mot (ou sous-mot), et ensuite servir d'entrée au classifieur pour une tâche donnée. Il y a deux modes à l'apprentissage. Soit les vecteurs contextuels sont pris comme entrée du classifieur aval, et le modèle de langue pré-entraîné n'est pas modifié. Soit à l'apprentissage pour la tâche aval, les paramètres du modèle de langue sont mis à jour, au même titre que les paramètres spécifiques au classifieur final visé. On parle de "fine-tuning" en anglais, ou réglage fin. C'est cette technique que nous allons utiliser.

L'autre grand type de modèle pré-entraîné sont les modèles "génératifs" (ou "auto-régressifs"), dont le représentant principal est le modèle GPT pour generative pre-trained transformer. Ces modèles sont utilisables pour peu qu'on transforme une tâche en "prédire un mot sachant un contexte gauche". Au départ moins performant que les modèles de type BERT, les modèles génératifs plus récents, grâce à leur grand nombre de paramètres (175 milliards de paramètres pour GPT-3 (Brown et al., 2020), et un nombre inconnu pour GPT-4) obtiennent de très bonnes performances.

Dans notre cas, nous avons une tâche de classification vers plusieurs dizaines de classes possibles (24 classes retenues). Cela reste transformable en une simple tâche de prédiction de mot sachant un contexte gauche. Mais nous n'avons pas eu le temps de tester des modèles génératifs à ce jour, et dans toute le reste du mémoire, nous utilisons des modèles bidirectionnels de type BERT.

2.4 Annotations sémantiques lexicographiques en français

Pour le français, il n'existe pas de lexique large augmenté d'informations sémantiques. Il y a eu plusieurs projets de construction d'un wordnet : le projet EuroWordnet (Vossen, 1998)mais qui n'est pas très couvrant ni très utilisé car il n'est pas utilisable librement ; et le WOLF (Sagot and Fišer, 2008), qui a été obtenu par projection automatique du Wordnet anglais. Il a une couverture et une qualité insuffisante pour des études lexicographiques sur le français.

C'est pourquoi dans notre travail, nous repartons d'un lexique couvrant (Wiktionary fr), et nous visons d'ajouter une annotation en supersens. C'est un compromis cf. d'une part on capitalise sur le travail collaboratif fait au sein de Wiktionary, et on espère obtenir une annotation en supersens de qualité suffisante pour des études lexicographiques.

Une telle ressource n'existe pas à notre connaissance pour le français. Des lexiques français avec classes sémantiques existent pour le français mais sont beaucoup moins couvrants (WOLF déjà cité, ou bien le RL-fr (Polguère, 2014)).

3 Données

Dans cette section, nous commencerons par présenter le Wiktionnaire et les différents élements de descriptions des sens lexicaux qui vont faire l'objet d'une classification sémantique (section 3.1). Nous présenterons ensuite le jeu d'étiquettes sémantiques, appelées *supersens*, utilisé pour classer les sens nominaux de cette ressource (section 3.2). Nous détaillerons enfin le processus d'annotation manuel effectué sur une partie des données (section 3.3). Celles-ci serviront de données d'entrainement et d'évaluation des classifieurs qui seront présentés dans la section suivante (section 4).

3.1 Le Wiktionnaire

3.1.1 Présentation du Wiktionnaire

Le *Wiktionnaire* est un projet de dictionnaire multilingue, collaboratif et libre, créé par la *Wikimedia Foundation*, la même organisation qui gère Wikipédia. Lancé en 2002, il vise à offrir des définitions, des traductions, des synonymes, des antonymes, des étymologies et des exemples d'utilisation pour les mots dans toutes les langues. Contrairement aux dictionnaires traditionnels, le contenu du Wiktionnaire est rédigé et modifié par des bénévoles du monde entier, permettant une mise à jour rapide et une couverture linguistique vaste et diversifiée. Chaque utilisateur peut contribuer en ajoutant ou en corrigeant des informations, rendant le Wiktionnaire un outil dynamique et en constante évolution. Les données sont publiées sous licence *Creative Commons*, garantissant leur libre accès et réutilisation.

Bien que développée par des non-spécialistes, nous avons choisi cette ressource pour son caractère libre, couvrant et pour sa qualité globale, jugée suffisante dans de précédentes études. Pour le français, la granularité et la cohérence de l'inventaire de sens ont notamment été jugées satisfaisantes, d'après l'accord inter-annotateurs dans une tâche de désambiguisation de verbes en contexte (Segonne et al., 2019).

3.1.2 Structuration des données du Wiktionnaire

Les données du Wiktionnaire sont organisées selon la hiérarchie suivante : des pages contenant des entrées lexicales lesquelles peuvent contenir plusieurs descriptions de sens lexicaux associés. La figure 1, qui reprend une partie de la

description proposée dans le Wiktionnaire pour le mot *crème*, nous permettra d'illustrer les différents éléments.

— Une **page** correspond généralement à un lemme ou une expression polylexicale. Dans notre exemple, la page décrit le mot CRÈME.

— Les pages sont découpées en **entrées lexicales** qui distinguent les différentes catégories grammaticales du mot et, au sein d'une même catégorie, les cas éventuels d'homonymie. Dans notre exemple, la page pour le mot *crème* est constituée de 3 entrées, deux nominales et une adjectivale. Deux entrées nominales sont distinguées, avec une première entrée qui décrit le mot de genre féminin, l'autre qui décrit le mot de genre masculin.

— Les entrées lexicales comportent une ou plusieurs descriptions de **sens lexicaux**, selon le caractère plus ou moins polysémique de l'unité décrite. Dans notre exemple, la première entrée décrit un nom féminin polysémique, dont 4 des sens lexicaux (cadre vert) sont représentés.

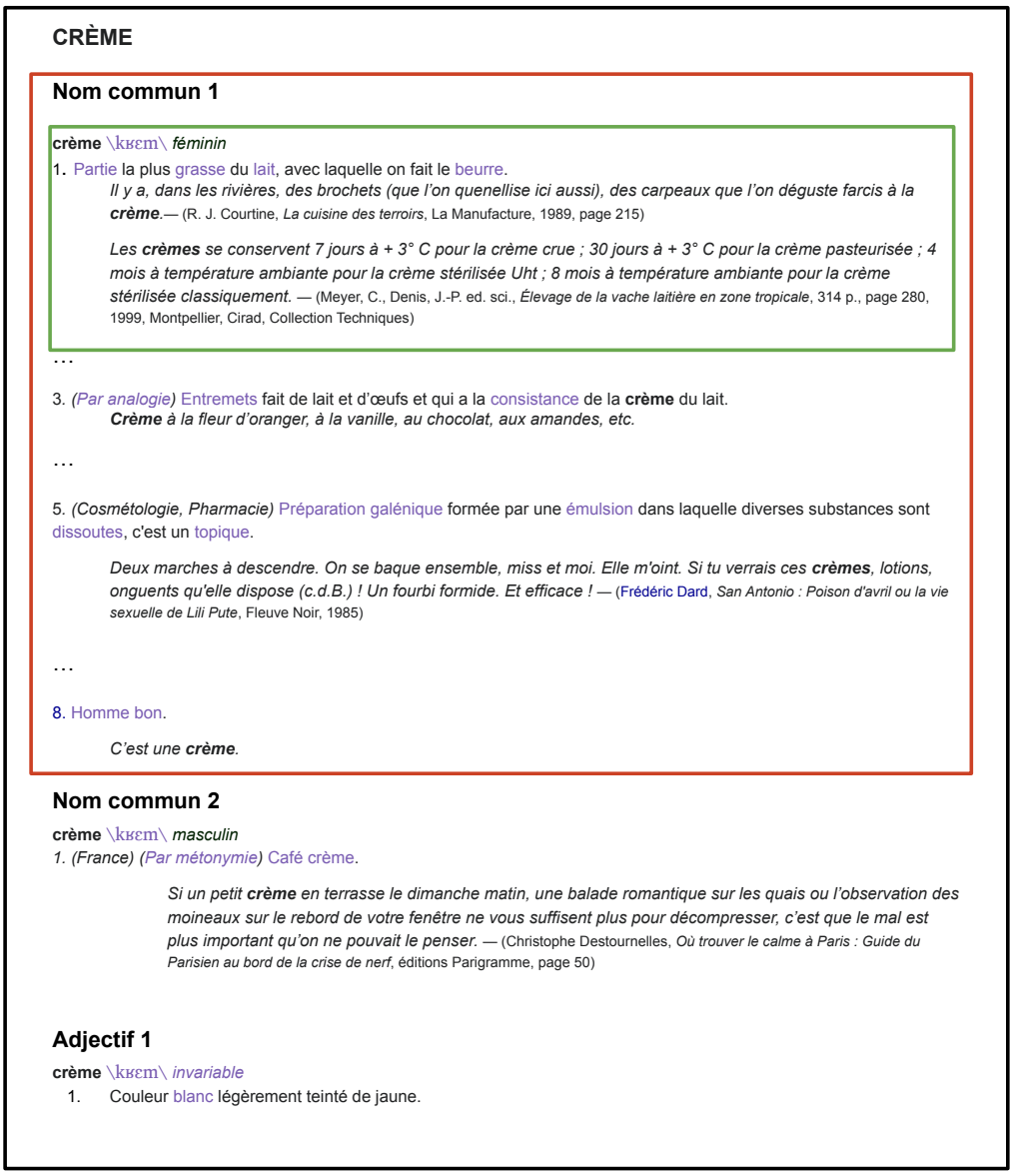

FIGURE 1 – Organisation d'une page du Wiktionnaire. La figure reprend une partie de la page (cadre noir) décrivant le mot *crème*, constituée de 3 **entrées** ("Nom commun 1", "Nom commun 2", et "Adjectif 1", la première entrée est encadrée en rouge), chacune d'elles constituée de **sens lexicaux** (le premier sens lexical de la première entrée est encadré en vert).

La classification sémantique effectuée dans notre travail se fait au niveau des sens lexicaux. Elle se concentre par ailleurs sur les sens lexicaux des noms. Ainsi, le travail va consister à attribuer une classe sémantique à chaque sens nominal (par ex la classe Person au sens n°8 de *crème*, cf Fig. 1) en s'appuyant sur les informations du Wiktionnaire associées aux sens. Celles-ci sont au nombre de quatre :

— Le **lemme** du sens décrit (qu'il s'agisse d'une unité monolexicale, comme *crème*, ou polylexicale, comme *fruit de mer*)

— Une **définition** lexicographique, qui décompose le sens défini

— Le cas échéant, des **étiquettes métalexicographiques** qui donnent des indications sur les relations entre sens (ex. par extension), sur les conditions d'utilisation du sens (ex. Cosmétologie), etc.

— Un ensemble, éventuellement vide, d'exemples d'emploi du sens lexical, ou **exemples lexicographiques**, tirés de corpus (ex. les deux exemples pour le sens 1 de l'entrée 1 de *crème*)

3.1.3 Extraction des données du Wiktionnaire

Pour extraire les données du Wiktionnaire avec lesquelles nous voulons travailler pour ce projet de recherche, nous avons utilisé un fichier Turtle de DBnary (Sérasset, 2014) de type ontolex qui représente les données au format RDF. Un triplet RDF est une structure de données composée de trois parties : sujet, prédicat et objet. Il représente une relation entre deux entités. Le fichier que nous utilisons met notamment à disposition les données de pages, entrées lexicales et sens lexicaux du Wiktionnaire au format RDF (figures 2, 3 et 4).

```
fra:crème  rdf:type       dbnary:Page ;
       dbnary:describes  fra:crème__nom__1 , fra:crème__nom__2 , fra:crème__adj__1 , fra:crème__adj__2 .
```

FIGURE 2 – Représentation au format RDF de la page du mot "crème" dans le fichier Turtle des données du Wiktionnaire.

```
fra:crème__nom__1  rdf:type    ontolex:Word , ontolex:LexicalEntry ;
        rdfs:label             "crème"@fr ;
        dbnary:hypernym        fra:laitage ;
        dbnary:hyponym         fra:chantilly ;
        dbnary:partOfSpeech    "-nom-" ;
        dcterms:language       lexvo:fra ;
        lexinfo:partOfSpeech   lexinfo:noun ;
        lime:language          "fr" ;
        ontolex:canonicalForm  fra:__cf_crème__nom__1 ;
        ontolex:sense          fra:__ws_1_crème__nom__1 , fra:__ws_2_crème__nom__1 , fra:__ws_3_crème__nom__1 ...
```

FIGURE 3 – Représentation au format RDF de la première entrée lexicale nominale du mot "crème" dans le fichier Turtle des données du Wiktionnaire.

```
fra:__ws_5_crème__nom__1
        rdf:type            ontolex:LexicalSense ;
        dbnary:senseNumber  "5" ;
        skos:definition     [ rdf:value  "Liqueur fine de certains types."@fr ] ;
        skos:example        [ rdf:value  "Crème de cassis."@fr ] .
```

FIGURE 4 – Représentation au format RDF du cinquième sens lexical nominal de la première entrée lexicale nominale du mot "crème" dans le fichier Turtle des données du Wiktionnaire.

Ainsi, en exploitant cette structure de données nous pouvons récupérer les informations pertinentes pour notre travail, en récupérant notamment les données des sens lexicaux tels que la définition et les exemples lexicographiques ainsi qu'en récupérant la hiérarchie entre les différents niveaux d'information, chaque entité étant identifiable par un ID.

Nous avons donc procédé, en utilisant la version du fichier la plus à jour en Mars 2023, à l'extraction des pages contenant au moins une entrée nominale elle-même contenant au moins un sens nominal. Nous avons décidé d'ignorer les sens dont la définition décrit une utilisation obsolète (ayant certaines étiquettes en début de définition telles que *obsolète* ou *ancien*) de ce nom dans le but d'obtenir une ressource plus représentative de l'état de la langue française actuellement.

3.2 Jeu d'étiquettes pour la classification : les supersens

3.2.1 Origine des supersens pour le français

Les **supersens** sont des catégories sémantiques à grain épais héritées des Unique Beginners de WordNet (Miller et al., 1990). Il s'agit de catégories sémantiques générales qui structurent différentes hiérarchies de WordNet en regroupant les synsets sous des concepts généraux et abstraits. Il y a 25 Unique Beginners pour les noms, 15 pour les verbes, et 2 pour les adjectifs. Parmi les noms, on trouve des catégories telles que *Act*, *Animal*, *Artifact*, *Body*, *Cognition*, *Event*, *Feeling*, *Food*, *Group*, *Location*, *Object*, *Person*, *State*, *Substance*, et *Time*. Ces classes ont été conçues à l'origine pour organiser le travail lexicographique, lors du développement de la ressource, et non pour faire de la classification sémantique. Cependant, ces classes ont été utilisées de cette manière, notamment depuis les premiers travaux sur le supersense tagging (Ciaramita and Johnson, 2003).

3.2.2 Définition des supersens pour la description du français

Les supersens originaux de WordNet ont été adaptées de plusieurs manières dans différents projets et notamment à l'occasion de la création du corpus FrSemCor (Barque et al., 2020). Ce corpus du français est le résultat d'une campagne d'annotation sémantique des noms du corpus Sequoia (Candito and Seddah, 2012), par ailleurs déjà annoté pour la morphologie et la syntaxe. L'annotation sémantique en supersens a été motivée par le manque de données de corpus annotées sémantiquement en français.

Les supersens issus de WordNet n'ayant pas été conçus spécifiquement pour de la classification sémantique, ils ont fait l'objet d'un travail de définition dans le cadre du développement du corpus FrSemCor. Ainsi, des classes ont été laissées de côté tandis que d'autres ont été créées pour l'intérêt de ce projet. En effet, l'objectif est d'avoir la classification la plus cohérente possible, que la couverture des classes soit la plus totale et que chaque sens tende à n'avoir qu'une seule classe possible et que les classes soient ainsi mutuellement exclusives. Nous pouvons notamment mentionner la création de supersens complexes, utilisant les opérateurs "*" et "x" pour combiner les supersens simples, "*" pour indiquer que les deux classes liées sont représentatives du sens décrit et "x" pour indiquer la composition des deux classes entre elles.

Parmi les supersens existants, pour notre tâche de classification des sens issus des noms du Wiktionnaire, nous avons retenu les vingt et un supersens simples et trois supersens complexes les plus fréquents dans les données annotées pendant le stage (voir section 3.3).

La création du corpus FrSemCor (Barque et al., 2020) introduit en outre des classes sémantiques encore plus générales qui peuvent regrouper plusieurs supersens : les **hypersens**. Ces classes très générales peuvent se révéler intéressantes, d'une part pour des considérations plus générales d'évaluation de la classification, et d'autre part pour fournir des informations sémantiques plus grossières sur les sens. Les classes retenues pour ce travail de recherche ainsi que leur hypersens associé sont illustrés dans la table ci-après (cf. table 1).

Supersens	**Hypersens**
Animal, Person	Animate entity
Artifact, Food, Body, Object, Plant, Substance	Inanimate entity
Act, Event, Phenomenon	Dynamic situation
Attribute, State, Feeling, Relation	Stative situation
Cognition, Communication	Informational object
Quantity	Quantification
Institution	Institution
Possession	Possession
Time	Time
Artifact*****Cognition	Inanimate entity*****Informational object
Act*****Cognition	Dynamic situation*****Informational object
Group**x**Person	Quantification**x**Animate entity

TABLE 1 – Liste des 24 supersens utilisés dans ce travail de recherche, avec leur hypersens correspondant

3.3 Classification manuelle de sens lexicaux

3.3.1 Sélection des sens à annoter manuellement

Notre objectif étant de proposer une classification automatique supervisée des sens nominaux du *Wiktionnaire*, il nous fallait des données d'entraînement pour nos classifieurs. Une annotation manuelle en supersens a donc été produite pour 16 092 sens nominaux du Wiktionnaire.

Ces sens lexicaux annotés manuellement sont des sens de 6 333 lemmes, avec une couverture des sens variable selon les lemmes (ex. dans certain cas, seul le premier sens du lemme a été annoté manuellement). Les 16 092 sens annotés manuellement proviennent d'une part d'études précédentes, en particulier (Aloui et al., 2020), dont on reprend 10 117 sens annotés, qui seront notre ensemble d'entraînement.

Deux sous-ensembles supplémentaires ont toutefois été sélectionnés et annotés spécifiquement pour les besoins de cette étude :

— **L'ensemble RANDOM** : il correspond à une sélection aléatoire de sens lexicaux dans l'ensemble du Wiktionnaire (en ignorant toutefois les lemmes déjà présents dans l'ensemble précédent). Ce sous-ensemble nous servira à évaluer les performances de notre classifieur sur un échantillon représentatif du Wiktionnaire.

— **L'ensemble FREQUENCY** : une autre partie des sens annotés manuellement correspond à une sélection parmi une liste de 10000 noms fréquents, non présents dans les ensembles précédents [1]. Cet ensemble nous sert à évaluer les performances sur un échantillon plus représentatif des conditions de classification en contexte. Pour chacun de ces noms fréquents, le sens apparaissant en premier dans le Wiktionnaire de manière systématique et parfois tous les sens de ce lemme ont été annotés manuellement.

1. Les fréquences ont été calculées dans un extrait de Wikipedia, Wikisource et Uncorpus (trois extraits respectivement de 53, 300 et 134 millions de tokens), dans la partie FR du corpus BigScience, disponible à https://huggingface.co/spaces/bigscience-data/bigscience-corpus.

3.3.2 Qualité de l'annotation

L'annotation a été réalisée par Lucie Barque. Pour évaluer la qualité de l'annotation, nous avons demandé à un second annotateur, Richard Huyghe, d'annoter en double aveugle les 204 sens lexicaux de 41 lemmes sélectionnés aléatoirement. L'accord obtenu sur ces données est de 0,76, avec un coefficient Kappa de Cohen de 0,74, ce qui correspond à un accord satisfaisant.

3.3.3 Résultats de l'annotation manuelle

Parmi les 16 092 sens des 6 533 lemmes annotés manuellement, on compte 53% de lemmes monosémiques. L'ambiguïté moyenne est de 2,5 sens par lemme. De plus, 2,5% de lemmes sont des expressions polylexicales. Chaque sens annoté est illustré par 1,8 exemple en moyenne, et 18% des sens sont dépourvus d'exemples.

Nous donnons plus d'informations quantitatives sur ces données dans la section partitionnement des données (section 5.1.1).

Cette tâche de classification sémantique manuelle a été l'occasion d'évaluer la qualité des descriptions du dictionnaire. Il ressort de cette évaluation que :

— Le Wiktionnaire contient un certain nombre de définitions et exemples problématiques, notamment pour la classification sémantique, comme par exemple des définitions disjonctives, des définitions et exemples que nous jugeons assez douteux, ou bien des méta-définitions et définitions circulaires qui décrivent de manière contournée ou trop générale des sens.

— Le Wiktionnaire comprend un certain nombre d'entrées dont les informations sont incomplètes voire manquantes, c'est-à-dire par exemple des définitions réduites à une étiquette lexicale qui spécifie un sens par rapport au sens précédent, des définitions manquantes et des sens n'ayant aucun exemple.

Toutefois, la qualité générale des descriptions est suffisamment bonne pour permettre leur classification sémantique dans la grande majorité des cas, comme l'illustre entre autre le bon accord inter-annotateur obtenu sur les données annotées en double-aveugle (cf section 3.3.2).

4 Architecture des classifieurs

Notre objectif a été d'obtenir un système prenant en entrée un sens lexical et renvoyant un supersens parmi les 24 supersens retenus.

Pour cela nous avons mis au point des classifieurs supervisés, bénéficiant d'un apprentissage par transfert grâce à des modèles de langue pré-entraînés.

Pour représenter un sens lexical en entrée, nous avons utilisé sa définition, parfois son lemme et parfois les exemples illustrant le sens. Or d'un point de vue linguistique, la définition d'un sens et un exemple illustrant un sens sont des objets très différents. La définition ne reprend en général pas le mot lui-même, et décrit le sens. L'exemple illustratif au contraire contient le mot, fléchi, en contexte. De ce fait, nous avons opté pour entraîner des classifieurs distincts : on parlera dans la suite de **classifieur de définition** versus **classifieur d'occurrence** selon que le texte en entrée est une définition lexicographique ou une occurrence de sens en contexte. À noter que les occurrences peuvent être des exemples lexicographiques (dans notre cas, issus du Wiktionnaire) ou bien des occurrences en corpus (dans notre cas, issus de FrSemCor).

Nos classifieurs varient également selon le type de modèle de langue pré-entraîné utilisé. Nous commencerons par détailler les classifieurs utilisant un modèle bidirectionnel (section 4.1). Les résultats des différentes expériences réalisées seront présentés dans la section 5.

4.1 Classifieur sur modèle contextuel bidirectionnel

4.1.1 Classifieur de définition

Pour ce classifieur, l'entrée est soit :

— la définition seule

— ou bien la contaténation du lemme et de la définition, séparés par " :".

On a donc une séquence de mots en entrée. L'architecture consiste ensuite à :

— tout d'abord tokeniser la séquence en utilisant le tokeniseur associé au modèle de langue pré-entraîné : la séquence de mots est découpée

en une séquence de **tokens**, un mot peut être conservé tel quel ou bien découpé en plusieurs tokens. En outre des tokens "spéciaux" sont ajoutés en début et fin de séquence.

— encoder cette séquence via un modèle de type BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2019) ;

— récupérer le vecteur contextuel du token spécial de début de séquence ([CLS]), destiné à fournir une représentation contextuelle de toute la séquence d'entrée ;

— et envoyer ce vecteur contextuel à un perceptron multicouche (MLP), fournissant en sortie une distribution de probabilités sur les supersens : en l'occurrence nous avons utilisé un MLP à une seule couche cachée.

Lors de l'entraînement, il y a une couche de "dropout" activée après la première couche linéaire du MLP, c'est-à-dire que certains neurones de la couche sont aléatoirement ignorés, ce qui rend le classifieur plus robuste. Les paramètres du MLP et du modèle de type BERT sont respectivement appris et fine-tunés.

L'architecture est illustrée à la figure 5. Dans les sections qui suivent, nous donnons quelques précisions supplémentaires sur le modèle de langue pré-entraîné utilisé (FlauBERT), et sur le perceptron multicouche. Les détails concernant le choix des hyperparamètres est donné dans le protocole expérimental (section 5.1).

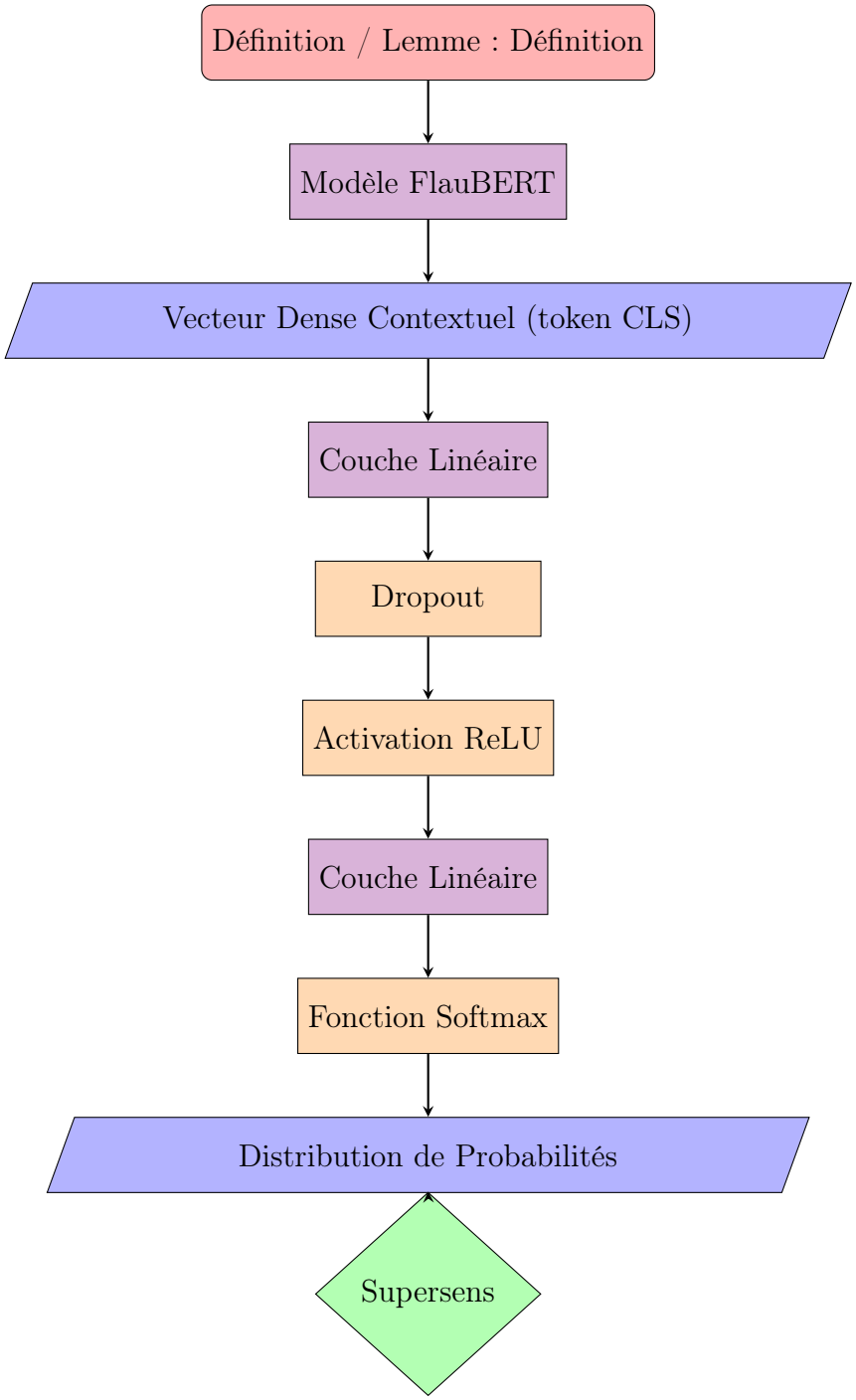

FIGURE 5 – Schéma de l'architecture du classifieur de définition avec modèle contextuel bidirectionnel. L'entrée est la définition, précédée ou pas du lemme.

4.1.1.1 Le modèle de langue pré-entraîné FlauBERT

FlauBERT (Le et al., 2020) est un modèle de langue bidirectionnel basé sur l'architecture Transformer (Vaswani et al., 2023), plus précisément sur BERT (cf. cours de TAL for NLP3). Le caractère bidirectionnel tient au fait que le modèle utilise une auto-attention sur la phrase entière, ce qui permet d'obtenir, pour chaque token de la séquence d'entrée, un vecteur dit "contextualisé", embarquant le contexte complet (à la fois à gauche et à droite). Cela permet au modèle de capturer les relations complexes entre les mots en apprenant à encoder des représentations contextuelles de haute qualité, optimisées pour les tâches de traitement du langage naturel. FlauBERT est pré-entraîné sur un large corpus de textes en français, ce qui lui permet de bien représenter les mots français dans leur contexte. Son tokeniseur est particulièrement adapté à la segmentation du texte en français et ses paramètres sont spécifiquement optimisés pour capturer les caractéristiques propres à cette langue.

4.1.1.2 Perceptron Multicouche

Le Perceptron Multicouche est un type de réseau de neurones artificiels utilisé en apprentissage automatique. Il s'agit notamment d'un modèle qui permet la résolution de problèmes non linéaires grâce à l'utilisation d'une ou plusieurs fonctions d'activation non linéaires. Un perceptron multicouche est composé de trois types de couches :

- **Couche d'entrée :** Les neurones de cette couche reçoivent les signaux d'entrée. Chaque neurone correspond à une caractéristique de l'entrée.
- **Couches cachées :** Entre la couche d'entrée et la couche de sortie se trouvent une ou plusieurs couches cachées. Le passage d'une couche à une autre se fait par combinaison linéaire (utilisant une matrice de paramètres, à apprendre) et l'application d'une fonction d'activation non linéaire.
- **Couche de sortie :** Les neurones de cette couche produisent le résultat final du réseau. Dans le cas qui nous intéresse, le vecteur à la couche de sortie est transformé (via la fonction log-softmax) en une distribution de log-probabilités sur les différentes classes de sortie (pour nous des supersens).

Pour un vecteur d'entrée, le vecteur des log-probabilités en sortie est obtenu à travers un processus appelé propagation avant : le vecteur d'entrée passe à travers les combinaisons linéaires et fonctions d'activation de chaque

couche. Les résultats successifs de cette transformation sont transmis à la couche suivante jusqu'à la couche de sortie.

L'apprentissage se fait par minimisation d'une fonction de perte, dans notre cas la log-vraisemblance négative : elle vaut l'opposé de la log-probabilité de la vraie classe. On a bien que minimiser cette perte correspond à maximiser la probabilité de la bonne classe (le bon supersens pour nous), et mécaniquement diminuer la probabilité des autres classes. L'algorithme de minimisation est un algorithme itératif de type descente de gradient, le gradient de la perte par rapport aux paramètres à apprendre est calculé automatiquement grâce à la librairie logicielle pytorch.

Comme dit supra, à l'apprentissage, les paramètres du modèle de langue pré-entraîné sont "fine-tunés", c'est-à-dire qu'ils sont mis à jour au même titre que les paramètres spécifiques du MLP. On est donc dans le cadre d'un apprentissage par transfert : les paramètres appris sur la tâche générique de prédiction de mot sachant un contexte vont être mis à profit et spécialisés pour la tâche de classification en supersens.

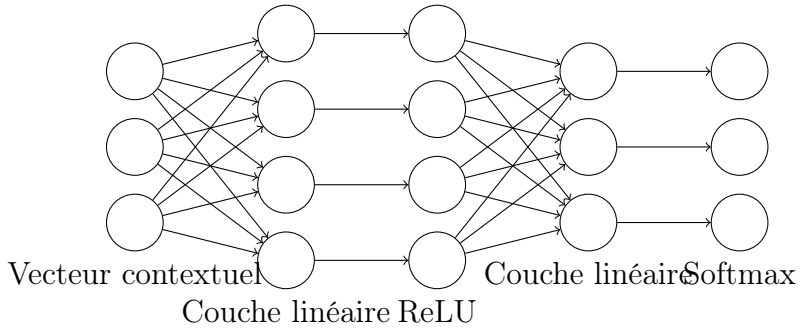

FIGURE 6 – Schéma d'un Perceptron Multicouche pour de la classification prenant un vecteur dense contextuel en entrée, avec deux couches linéaires, une couche d'activation ReLU (ou dropout+ReLU) et en sortie la distribution de probabilités des classes

4.1.2 Classifieur d'occurrence de mot

L'architecture pour le classifieur d'occurrence est la même, la différence résidant dans le choix du vecteur contextuel fourni en entrée du MLP : pour le classifieur de définition, nous venons de voir que c'est le vecteur contextuel

du token spécial initial ([CLS]) qui était donné. Pour prédire le sens d'un mot en contexte, on va aider le classifieur en fournissant au MLP le vecteur contextuel correspondant à l'occurrence du mot à classer.

Plus précisément, au sein d'un exemple lexicographique, on repère où se situe l'occurrence du mot illustré. On utilise pour cela une lemmatisation ou bien une recherche simplifiée, sur formes fléchies, possible étant donné qu'on ne travaille que sur des occurrences de nom, dont la flexion n'est pas trop compliquée à gérer [2]. Dans l'exemple illustrant un sens lexical, on choisit la première occurrence de forme fléchie qui concorde avec le lemme du sens lexical. On utilise ensuite le vecteur contextuel du premier token de cette occurrence.

Par exemple pour le premier sens de l'unique entrée du lemme *souscription*, dont la définition est *Signature qu'on met au-dessous d'un acte pour l'approuver.*, un des exemples lexicographiques est *Ils ont approuvé cet acte par leur souscription, par leurs souscriptions.* Dans cet exemple nous utilisons le vecteur dense associé au huitième token *souscription</w>* (figure 7) correspondant à l'ID *16395* (figure 8) dans le vocabulaire du FlauBERT utilisé.

['Ils</w>', 'ont</w>', 'approuvé</w>', 'cet</w>', 'acte</w>',
'par</w>', 'leur</w>', 'souscription</w>', ',</w>', 'par</w>',
'leurs</w>', 'sous', 'criptions</w>', '.</w>']

FIGURE 7 – Tokens de l'exemple donné pour le nom "souscription" résultant du tokeniseur de FlauBERT.

[262, 62, 6602, 192, 1690, 38, 81, 16395, 14, 38, 121, 10967, 21246, 16]

FIGURE 8 – ID des tokens de l'exemple donné pour le nom "souscription" résultant du tokeniseur de FlauBERT.

Pour prédire un sens s avec seulement les exemples lexicographiques, nous utilisons la moyenne des scores obtenus à partir de chaque exemple individuel.

2. En utilisant notre méthode nous avons pu localiser plus de 95% des occurrences des noms dans les exemples lexicographiques du Wiktionnaire.

Notons ex_i les n exemples lexicographiques pour $1 < i < n$, et $\mathbf{score}_{ex_i}$ le vecteur de log-probabilités de chaque supersens, pour l'exemple ex_i. Nous noterons le vecteur résultant de cette opération $\mathbf{score}_{ex}(s)$. Nous avons ainsi :

$$\mathbf{score}_{ex}(s) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{score}_{ex_i}$$

Une fois le vecteur $\mathbf{score}_{ex}(s)$ obtenu, le supersens prédit $\hat{c}$ est simplement celui de plus fort score :

$$\hat{c} = \text{argmax}_{c \in C} score_{ex}(s)_c$$

L'architecture est illustrée à la figure 9. Comme pour le classifieur de définition, les paramètres du perceptron multicouche et du modèle BERT sont respectivement appris et finetunés.

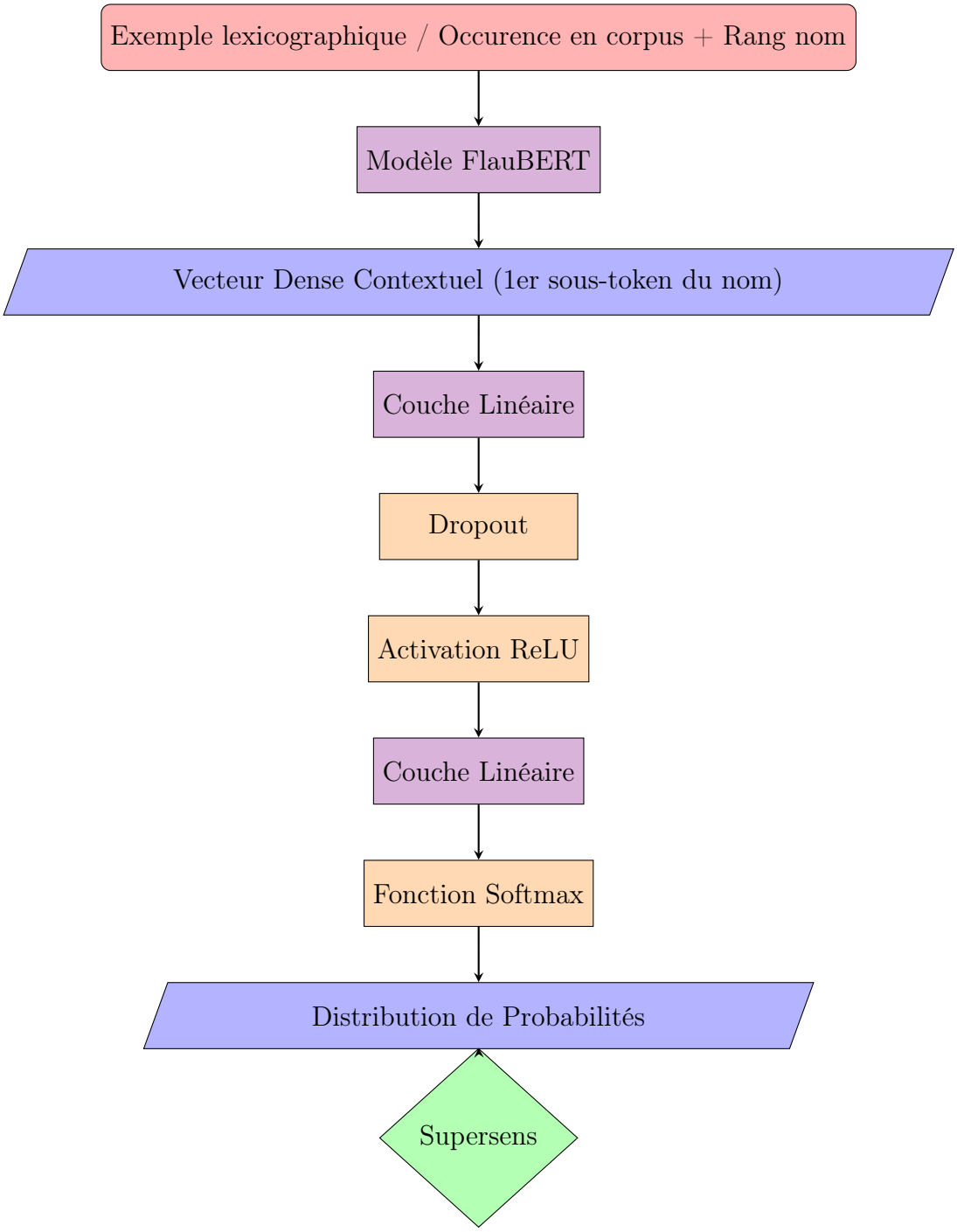

FIGURE 9 – Schéma de l'architecture du classifieur d'exemple lexicographique ou d'occurence en contexte avec modèle contextuel bidirectionnel

Il faut noter que si les deux architectures des classifieurs de définition et d'occurrence sont très semblables techniquement, pour le classifieur de définition, le rang du token dont on prend le vecteur contextuel est constant (rang 0), alors qu'il est variable dans le cas du classifieur d'occurrences. Dans notre implémentation nous utilisons des opérations matricielles pour sélectionner les bons vecteurs contextuels pour tout un "batch" de données d'entrée.

4.1.3 Combinaison des classifieurs de définition et d'occurrence

Nous avons vu jusqu'à présent comment classifier en supersens une définition et un exemple lexicographique. Nous allons maintenant voir comment nous classifions un sens lexical s en utilisant à la fois la définition et les exemples de chaque sens.

La première étape consiste à obtenir la distribution de probabilités des supersens en utilisant le classifieur de définition. Plus précisément, nous utilisons le vecteur de log-probabilités, que nous noterons $\mathbf{score}_{def}(s)$.

Ensuite, si le sens lexical s a au moins un exemple, pour chaque supersens, nous allons utiliser la moyenne des scores obtenus à partir de chaque exemple individuel. Notons ex_i les n exemples lexicographiques pour $1 < i < n$, et $\mathbf{score}_{ex_i}$ le vecteur de log-probabilités de chaque supersens, pour l'exemple ex_i. Nous noterons le vecteur résultant de cette opération $\mathbf{score}_{ex}(s)$. Nous avons ainsi :

$$\mathbf{score}_{ex}(s) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{score}_{ex_i}$$

Pour obtenir le vecteur score final, noté $\mathbf{score}(s)$, contenant toujours un score par supersens, nous faisons la moyenne pondérée de $\mathbf{score}_{ex}(s)$ et $\mathbf{score}_{def}(s)$ (vecteur scores via définition et scores via exemples lexicographiques). Les poids dans cette moyenne pondérée sont les exactitudes moyennes des classifieurs correspondants [3] :

$$\mathbf{score}(s) = \alpha_{def} * \mathbf{score}_{def}(s) + \alpha_{ex} * \mathbf{score}_{ex}(s),$$

3. La moyenne des exactitudes sur les ensembles rand-dev et freq-dev.

$$\alpha_{def} = \begin{cases} exactitude(clf\_def) & \text{si le sens a une définition} \\ 0 & \text{sinon} \end{cases}$$

$$\alpha_{ex} = \begin{cases} exactitude(clf\_ex) & \text{si le sens a un exemple ou plus} \\ 0 & \text{sinon} \end{cases}$$

Une fois le vecteur **score**(s) obtenu, le supersens prédit $\hat{c}$ est simplement celui de plus fort score :

$$\hat{c} = \text{argmax}_{c \in C} score(s)_c$$

5 Expériences

Dans cette partie, nous allons introduire le protocole expérimental et expliquer comment nous avons partitionné les données, quelles procédures d'entraînement nous avons implémentées et nous allons expliciter les métriques d'évaluation que nous avons utilisées (section 5.1). Ensuite, nous allons présenter les résultats des différentes expériences réalisées avec des classifieurs sur modèle contextuel bidirectionnel (section 5.2).

5.1 Protocole expérimental

5.1.1 Partitionnement des données

Nous disposons de 16 092 annotés manuellement en supersens. Nous les avons découpés en différents jeux de données :

- Données d'entraînement : Les données qui sont utilisées pendant l'entraînement qui permettent d'optimiser les paramètres du modèle afin qu'il s'améliore sur la tâche de classification.
- Données de validation : Les données qui sont utilisées pour l'évaluation des performances du classifieur lors de l'entraînement sur des données non utilisées lors de celui-ci afin d'ajuster les hyperparamètres et d'éviter que le modèle se spécialise trop sur les données d'entraînement et ne soit pas capable de généraliser. Nous utilisons deux ensemble de validation **freq-dev** et **rand-dev**, contenant respectivement les sens venant de la moitié des lemmes présents dans les données FREQUENCY, et les sens venant de la moitié des lemmes présents dans les données RANDOM.
- Données d'évaluation : Les données restantes de FREQUENCY et RANDOM constituent les ensembles **freq-test** et **rand-test**. Ces données ne sont jamais utilisées lors de l'entraînement et la recherche d'hyperparamètres, et permettent d'évaluer les performances du classifieur final sur des données du même type mais qui n'ont jamais été vues.

Il est important de noter que nous avons fait en sorte qu'il n'y ait aucune intersection de lemmes dans des sens de jeux de données différents. En effet, étant donné que nous utilisons le lemme dans la chaîne de caractères donnée au modèle pour la classification d'un sens lexical, et de manière générale car

nous souhaitons conserver des considérations sur les lemmes, il semble pertinent de séparer ainsi les sens lexicaux.

Nous pouvons observer ci-dessous la répartition des sens dans les différents jeux de données d'une part (table 2), et d'autre part les distributions des classes sémantiques que nous y trouvons (table 4).

Jeu	Sens	Lemmes
train	10117	4012
freq-dev	1581	465
freq-test	1339	448
rand-dev	540	472
rand-test	649	473

TABLE 2 – Nombre de sens lexicaux et lemmes dans chaque jeu de données.

On peut voir dans la table 2 que pour des nombres comparables de lemmes distincts comparables (environ 460) les jeux de données "freq" correspondent à environ trois fois plus de sens lexicaux que les jeux de données "rand".

Subset	rand-dev	freq-dev	rand-test	freq-test	train
tous les lemmes	540	1581	649	1339	10117
lemmes monosémiques	457	257	446	267	1771
lemmes polysémiques	83	1324	203	1072	8346
lemmes simples	456	1581	574	1339	10117
lemmes complexes	84	0	75	0	0

TABLE 3 – Nombre de sens lexicaux pour chaque jeu de données, en tout (1ère ligne), et selon que le lemme correspondant est mono- ou polysémique, et est un lemme simple ou composé.

La table 4 de répartition des supersens montre que la distribution des classes est très inégale dans les données annotées, avec certaines classes beaucoup plus représentées que d'autres. Le jeu "train" a été repris d'études précédentes, et nous avons décidé de l'utiliser tel quel. Les répartitions dans les données frequency et random sont très différentes de celle des données train.

La classe "person" couvre presque un tiers des données random. Après inspection, il s'avère qu'il s'agit principalement de sens de gentilés, habitants d'un lieu.

Ces différences pourraient provoquer des biais dans le classifieur et des écarts de performance de manière générale, avec plus de chance pour les classes les plus représentées dans les données d'entraînement d'amener le classifieur à avoir de bonnes performances sur elles. Cependant, il n'existe pas nécessairement une corrélation entre le nombre d'exemples d'une classe vus à l'entraînement et les performances sur cette classe. Notamment, certaines classes sont potentiellement plus simples à capturer et discriminer tandis que d'autres peuvent présenter des frontières plus floues avec d'autres classes et donc amener à plus de confusion pour le modèle, et même parmi les annotateurs eux-mêmes. Nous analyserons cet aspect dans la section 5.2.2.

Supersense	train	freq-dev	rand-dev	freq-test	rand-test
act	11.1	19.5	7.6	23.0	8.3
act*cognition	1.6	3.0	0.7	3.4	0.5
animal	3.9	0.8	5.7	0.8	4.0
artifact	12.4	13.9	9.6	16.1	19.4
artifact*cognition	1.0	2.6	0.4	2.8	2.6
attribute	6.7	9.4	3.0	7.9	2.9
body	3.1	2.1	2.0	1.3	1.2
cognition	9.2	11.6	6.9	11.4	6.0
communication	1.6	1.3	2.6	0.7	0.9
event	4.6	3.5	3.1	2.7	2.8
feeling	2.4	1.5	0.7	1.1	1.2
food	3.1	1.4	2.2	1.0	1.4
groupxperson	0.8	2.1	0.2	2.0	2.0
institution	3.7	1.6	0.6	2.1	1.1
object	4.1	3.8	5.4	2.9	3.9
person	9.5	7.5	34.4	6.6	27.6
phenomenon	1.3	1.0	1.1	0.7	0.8
plant	3.4	0.9	2.6	0.9	4.0
possession	2.4	1.3	1.1	1.6	0.5
quantity	2.9	1.6	3.1	2.3	1.7
relation	1.0	0.6	0.0	0.1	1.1
state	4.2	4.8	3.1	5.5	2.0
substance	4.1	3.0	3.3	1.6	3.1
time	1.8	1.1	0.4	1.4	1.1

TABLE 4 – Distribution des supersens dans les sens annotés du Wiktionnaire en pourcentage pour chaque jeu de données.

Supersense	rand-dev simple	rand-dev MWE	rand-test simple	rand-test MWE
act	7.0	10.7	8.4	8.0
act*cognition	0.4	2.4	0.5	0.0
animal	5.3	8.3	2.4	16.0
artifact	9.2	11.9	20.0	14.7
artifact*cognition	0.2	1.2	2.8	1.3
attribute	3.1	2.4	2.9	2.7
body	1.7	3.6	1.0	2.7
cognition	5.9	11.9	5.9	6.7
communication	2.4	3.6	1.0	0.0
event	3.1	3.6	2.1	8.0
feeling	0.9	0.0	1.2	1.3
food	2.0	3.6	1.2	2.7
groupxperson	0.2	0.0	1.9	2.7
institution	0.2	2.4	0.9	2.7
object	5.7	3.6	4.0	2.7
person	39.2	8.3	30.0	9.3
phenomenon	1.1	1.2	0.5	2.7
plant	2.2	4.8	3.5	8.0
possession	1.3	0.0	0.5	0.0
quantity	3.5	1.2	1.9	0.0
relation	0.0	0.0	1.2	0.0
state	2.0	9.5	2.1	1.3
substance	2.8	5.9	3.1	2.7
time	0.4	0.0	0.7	4.0

TABLE 5 – Distribution des supersens dans les sens annotés du Wiktionnaire des données RANDOM en pourcentage selon que le lemme soit simple ou complexe.

Pour entraîner le classifieur d'occurrences, nous utilisons les exemples des sens lexicaux annotés en supersens (un exemple hérite du supersens associé au sens illustré), et les occurrences issues du corpus FrSemCor : comme nos jeux train, random-dev, random-test, freq-dev, freq-test correspondent à des ensemble disjoints de lemmes, nous avons pu simplement affecter à ces différents jeux les occurrences du FrSemCor en fonction de leur lemme.

Jeu	Exemples/Occurences
train	21606
freq-dev	4564
freq-test	4048
rand-dev	558
rand-test	852

TABLE 6 – Nombre d'occurrences en contexte annotées en supersens, par jeu de données. Les occurrences sont soit des exemples lexicographiques associés aux sens lexicaux du Wiktionnaire soit des occurrences du corpus FrSemCor.

Supersense	train	freq-dev	rand-dev	freq-test	rand-test
act	10.5	19.5	10.0	22.2	14.7
act*cognition	2.0	3.0	0.2	3.3	0.9
animal	2.1	0.5	3.8	0.4	1.1
artifact	7.7	9.2	10.7	10.1	15.6
artifact*cognition	1.1	2.2	0.2	2.0	3.3
attribute	6.5	10.8	9.3	8.5	2.5
body	2.6	2.2	2.5	0.8	1.7
cognition	8.1	10.3	10.7	9.2	10.5
communication	1.0	1.3	2.1	0.3	0.6
event	4.2	3.3	4.5	6.2	3.6
feeling	2.8	2.5	3.2	1.0	4.1
food	1.9	1.1	3.4	0.4	1.4
groupxperson	0.9	2.1	0.2	2.4	2.9
institution	6.9	4.9	0.7	6.9	1.7
object	3.2	3.0	5.5	2.3	3.0
person	13.0	8.7	12.3	8.8	11.5
phenomenon	1.1	1.1	3.6	1.3	0.9
plant	2.7	0.4	1.6	0.8	1.5
possession	2.3	1.7	4.3	1.9	0.2
quantity	3.9	1.1	4.5	1.5	2.1
relation	0.7	0.6	0.0	0.4	1.5
state	4.5	4.4	3.0	6.7	3.3
substance	5.5	3.9	3.2	1.3	2.4
time	4.8	2.1	0.4	1.2	9.1

TABLE 7 – Distribution des supersens dans les exemples lexicographiques des sens du Wiktionnaire annotés et des occurrences en contexte des noms associés à chaque jeu de données.

5.1.2 Entrainement du classifieur sur modèle contextuel bidirectionnel

La procédure générale d'entraînement consiste à ajuster les paramètres de FlauBERT et les poids du Perceptron Multi Couche par un algorithme de type descente de gradient. Comme indiqué précedemment, la perte utilisée

est la log-vraisemblance négative, laquelle vaut l'opposé de la log-probabilité de la vraie classe. En ce qui concerne l'implémentation, nous utilisons la librairie pytorch pour les différents modules du classifieur et l'optimisation. L'optimiseur que nous avons utilisé est AdamW.

Nous surveillons la performance du classifieur à chaque époque d'entraînement sur les deux jeux de données de validation (freq-dev et rand-dev) et utilisons la stratégie d'arrêt prématuré en utilisant la moyenne de la perte sur ces deux jeux de données. Ceci permet d'éviter une spécialisation trop importante sur les données d'entraînement et une perte de généralisation. Plus précisément, lorsque qu'après un certain nombre d'époques, égal à la valeur de l'hyperparamètre "patience", la perte sur la moyenne des deux jeux ne baisse pas alors nous arrêtons l'entraînement. Nous avons utilisé une patience de *2* après plusieurs essais et observations du comportement de la perte pendant l'entraînement.

Les hyperparamètres pour les expériences sont les suivants :

— Le taux d'apprentissage : La valeur de pas dans l'algorithme d'optimisation de type descente de gradient utilisé pour entraîner le classifieur.

— La taille de la couche cachée : Le nombre de neurones intermédiaires entre l'entrée et la sortie qui y sont liés par des couches linéaires.

— La valeur de dropout : La probabilité pour un neuronne d'être désactivé dans l'état où le dropout est utilisé.

Les hyperparamètres ont d'abord fait l'objet d'une recherche approximative des meilleures intervalles de valeurs. Par la suite, nous avons appliqué la stratégie de recherche en grille, laquelle consiste à choisir quelques valeurs pour chaque hyperparamètre et à tester pour toutes les combinaisons possibles les performances du classifieur après entraînement avec ces valeurs sur les jeux de données de validation. Pour s'assurer de la robustesse des performances suivant les combinaisons, nous avons en général observé les résultats sur cinq entraînements distincts pour chaque configuration. Nous avons finalement sélectionné les valeurs des hyperparamètres pour obtenir les meilleurs classifieurs de définition et d'occurrence (Table 8)

Hyperparamètres
Taux d'apprentissage : 0.000005
Taille couche cachée : 768
Dropout : 0.3

TABLE 8 – Hyperparamètres pour l'entraînement des modèles avec finetuning de FlauBERT

Nous avons défini une évaluation servant de référence de base (une "baseline" en anglais) afin de pouvoir juger dans quelle mesure nos classifieurs sont vraiment performants. Etant donné le grand nombre de classes possibles, les évaluations références classiques telles que l'assignation de la classe la plus fréquente ou une classe aléatoire ne sont pas tellement pertinentes dans le sens où, de par leur nature couplée à celle de notre cadre de classification, elles seront très basses et la comparaison avec nos résultats de classifieurs entraînés serait peu indicative. Ainsi nous avons choisi de prendre comme référence de base l'évaluation d'un classifieur suivant la même architecture que le classifieur de définition, utilisant uniquement la définition sans le lemme en début de chaîne de caractères, et en figeant les paramètres de FlauBERT afin de ne pas les ajuster et de finalement n'apprendre que les paramètres du perceptron multicouche. En faisant de cette manière nous pourrons voir à quel point l'ajustement fin des paramètres de FlauBERT jouent un rôle dans les performances du classifieur. Les hyperparamètres sont les mêmes à l'exception du taux d'apprentissage qui est pour ce classifieur de 0.00005 soit dix fois plus important que pour les classifieurs avec ajustement des paramètres de FlauBERT.

5.1.3 Métriques d'évaluation

Pour rappel, nous effectuons dans ce travail de la classification supervisée. Ainsi, chaque exemple d'entraînement est composé d'une part des informations nécessaires au modèle pour effectuer des prédictions et d'autre part d'une classe référence qui a été attribuée par des annotateurs et qui représente la classe que nous devons considérer juste pour cet exemple. Dans notre cas, chaque sens (constitué de la définition, 0 à n exemples lexicographiques, un lemme, et 0 à n étiquettes de la définition) se voit attribué un supersens référence.

Pour évaluer les performances de nos classifieurs nous avons utilisé différentes métriques et cadres d'analyse pour tenter de comprendre les forces et faiblesses des modèles entraînés. Etant donné notre problème de classification mono-étiquette, dont certaines étiquettes sont complexes, nous utilisons les métriques :

— **Exactitude** : la proportion de sens dont la classe prédite correspond à la classe référence.

— **Exactitude partielle** : la proportion de sens dont la classe prédite et la classe référence partagent au moins une partie en commun, c'est à dire que deux supersens/hypersens complexes partageant un supersens/hypersens nous amènent à considérer la prédiction comme juste et de même un supersens/hypersens simple étant compris dans un supersens complexe nous amènent à considérer la prédiction comme juste.

Nous avons également utilisé la métrique de la **F-mesure** afin de mesurer les performances du classifieur par supersens/hypersens individuels. Il s'agit de la moyenne harmonique du rappel et de la précision.

La Précision est calculée comme suit :

$$\text{Précision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

où :

— TP est le nombre de vrais positifs (instances positives correctement prédites),

— FP est le nombre de faux positifs (instances négatives incorrectement prédites comme positives).

Le Rappel est calculé comme suit :

$$\text{Rappel} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

où :

— FN est le nombre de faux négatifs (instances positives incorrectement prédites comme négatives).

Le score F1 (F-mesure) est la moyenne harmonique de la Précision et du Rappel :

$$\text{F1 score} = 2 \cdot \frac{\text{Précision} \cdot \text{Rappel}}{\text{Précision} + \text{Rappel}}$$

5.2 Résultats des expériences avec modèle contextuel bidirectionnel

5.2.1 Résultats généraux pour la classification de sens

En observant les résultats d'exactitude, sur les jeux freq-dev et rand-dev, de tous les classifieurs que nous avons évalués (tables 9 et 10), nous pouvons faire plusieurs remarques.

Tout d'abord, nous pouvons noter que l'ajustement fin des paramètres de FlauBERT permet un gain considérable de performance comparé à l'utilisation de FlauBERT avec ses paramètres figés (exactitude sur les supersens de 78.1% contre 61.3%). Ainsi, malgré la qualité certaine des représentations contextuelles de texte avec le modèle de base de FlauBERT, il est largement préférable d'ajuster ses paramètres pour cette tâche de classification sémantique. En outre, de manière systématique, le classifieur utilisant la définition, le lemme et les potentiels exemples obtient les meilleurs résultats (dernière ligne).

L'exactitude obtenue pour les données RANDOM est toujours plus élevée que celle obtenue pour les données FREQUENCY en cas d'utilisation de la définition. Nous pouvons potentiellement imputer cela au fait que la classe *Person* est la classe très largement majoritaire des données RANDOM avec une proportion de 34.4% et il s'agit d'une classe très facile à classifier de manière générale, d'autant plus quand une bonne partie de ces données prend la forme générique de gentilés et qu'il s'agit aussi de la classe majoritaire dans les données d'entraînement qui comportent également des gentilés.

Ensuite, nous pouvons remarquer que l'apport du lemme dans la chaîne de caractères de la définition est clair avec un gain d'exactitude pour les supersens de plus de 3.5 points pour les données FREQUENCY (exactitude sur les supersens de 73.1% contre 76.7%) et de plus de 5 points pour les données RANDOM (exactitude sur les supersens de 78.1% contre 83.3%). En analy-

sant les erreurs commises par les classifieurs utilisant le lemme, nous avons pu remarquer qu'il peut parfois potentiellement nuire à la classification lorsque le lemme semble avoir un sens particulièrement prévalent associé à une certaine classe sémantique qui peut ne pas être assignée à tous ses sens dans le lexique. Cependant, dans la plupart des cas, il semblerait que le lemme permette de renforcer la projection de la définition et de sa représentation dense avec le modèle FlauBERT sur le spectre sémantique.

Le classifieur utilisant uniquement les exemples pour la classification de sens n'obtient pas des performances très impressionnantes (ligne ex, exactitude sur les supersens inférieure à 66%), notamment comparé aux classifieurs utilisant la définition et l'ajustement des paramètres de FlauBERT. Au-delà de la nature très différente d'une définition par rapport à une occurrence d'un nom en contexte, nous ne nous attendions pas à un écart aussi important entre les deux. En observant quelques exemples du Wiktionnaire extraits, nous supposons qu'il y en a un certain nombre pour lesquels le texte n'est pas de bonne qualité, notamment dans le cadre d'un modèle qui doit produire une représentation contextuelle d'un mot en son sein. Cette hypothèse est renforcée par les quelques test de classification des occurrences de noms dans le corpus FrSemCor dont l'évaluation était plutôt du même ordre que les classifieurs utilisant la définition.

	rand-dev		freq-dev	
	Supersens	Hypersens	Supersens	Hypersens
frozen bert baseline	61.3	72.8	47.5	57.6
def	78.1	86.5	73.1	78.9
def+lemme	83.3	90.6	76.7	82.2
ex	65.7	77.4	65.0	72.5
def+lemme & ex	**84.3**	**91.3**	**77.1**	**83.0**

TABLE 9 – Exactitude en %, sur les données de validation RANDOM et FREQUENCY, en utilisant la définition seule (**def**), le lemme concaténé à la définition **def+lemme** , les exemples lexicographiques seuls (**ex**), et la combinaison du classifieur de définition et classifieur d'occurrences (**def+lemme & ex**, cf. section 4.1.3).

	rand-dev		freq-dev	
	Supersens	Hypersens	Supersens	Hypersens
frozen bert baseline	62.2	73.7	51.9	62.2
def	79.4	87.8	77.0	83.0
def+lemme	84.1	91.3	80.2	85.9
ex	66.8	78.5	69.7	77.2
def+lemme & ex	**84.8**	**91.9**	**80.3**	**86.2**

TABLE 10 – Exactitude partielle en %, sur les données RANDOM et FREQUENCY, en utilisant la définition seule (**def**), le lemme concaténé à la définition **def+lemme** , les exemples lexicographiques seuls (**ex**), et la combinaison du classifieur de définition et classifieur d'occurrences (**def+lemme & ex**, cf. section 4.1.3).

5.2.2 Résultats détaillés du meilleur classifieur de sens

Nous pouvons remarquer en analysant les résultats détaillés du meilleur classifieur obtenu, à savoir celui utilisant la définition avec le lemme et les exemples (**def+lemme & ex**, cf. section 4.1.3), qu'il existe un écart important de performance en fonction des classes sémantiques (table 11). Au niveau des hypersens, nous pouvons voir que le F1-score pour les entités et les situations dynamiques est très élevé par rapport aux autres. Au niveau des supersens, les classes *Animal* et *Person* appartenant aux entités animées sont des classes particulièrement bien discriminées par notre classifieur.

Hypersens	Supersens	rand dev		freq dev	
Animate entity	Animal	97.7	90.9	96.6	100.0
	Person		98.4		96.2
Inanimate entity	Artifact	93.9	88.9	90.9	86.3
	Body		76.9		84.9
	Food		76.2		85.7
	Object		57.7		68.4
	Plant		75.0		96 .5
	Substance		75.0		81.4
Dynamic situation	Act	89.2	87.1	86.7	85.9
	Event		75.7		70.0
	Phenomenon		0.0		48.3
Stative situation	Attribute	78.1	83.9	79.7	70.4
	Feeling		85.7		64.0
	Relation				29.6
	State		53.8		62.2
Informational object	Cognition	77.5	66.7	69.9	65.8
	Communication		69.2		74.4
Quantification	Quantity	85.7	85.7	61.2	61.2
Institution	Institution	57.1	57.1	68.1	68.1
Possession	Possession	71.4	71.4	81.8	81.8
Time	Time	66.7	66.7	72.2	72.2
Dynamic situation*Informational object	Act*Congition	66.7	66.7	53.1	53.1
Inanimate entity*Informational object	Artifact*Cognition	100.0	100.0	73.7	73.7
QuantificationxAnimate entity	GroupxPerson	0.0	0.0	84.7	84.7

TABLE 11 – F-scores par supersens et hypersens en % pour les jeux de données rand-dev et freq-dev obtenus lors de l'évaluation du meilleur classifieur, celui combinant définitions avec lemme et exemples(**def+lemme & ex**, cf. section 4.1.3).

Nous pouvons voir ci-dessous la matrice de confusion obtenue pour les données de validation FREQUENCY freq-dev (figure 10). Deux phénomènes principaux auxquels nous nous attendions dans une certaine mesure peuvent être observés :

— les confusions importantes entre les classes complexes et les classes simples qui les composent. Nous pouvons notamment citer les confusions (*act*cognition*, *cognition*), (*act*cognition*, *act*) et (*artifact*cognition*, *cognition*). Nous observons une sorte de propagation de cette confusion entre les deux classes simples composant une classe complexe, n'ayant pourtant à l'origine aucun lien spécial, comme les confusions des paires (*act*, *cognition*) et (*artifact*, *cognition*) ;

— les confusions fréquentes entre supersens correspondant à des classes sémantiquement proches et regroupables sous un même hypersens. C'est le cas des quatre classes relevant de l'hypersens *Stative situation*. Par exemple les sens *Attribute* de la référence sont prédits 8 fois *Feeling* et 8 fois *State*.

Ceci tend à illustrer une des limitations potentielles de cette tâche de classification sémantique à grain épais. En effet, ces confusions observées sont difficiles à contourner, et même entre annotateurs humains il s'agit de classes qui peuvent amener à des désaccords importants.

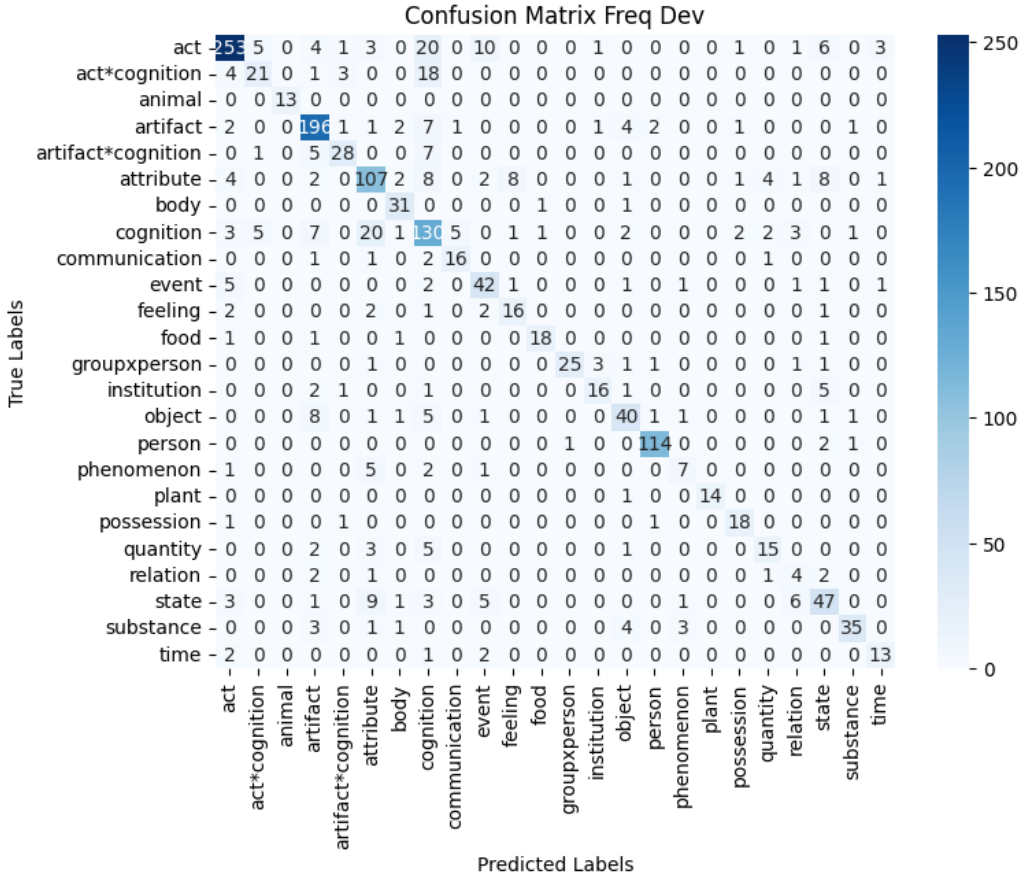

FIGURE 10 – Matrice de confusion pour le jeu de données freq-dev obtenue en évaluant le meilleur classifieur, celui combinant définitions avec lemme et exemples(**def+lemme & ex**, cf. section 4.1.3).

En observant les performances en différenciant les sens provenant de lemmes simples et ceux provenant de lemmes complexes (tables 12 et 13), c'est-à-dire des expressions polylexicales, nous pouvons remarquer que les performances sont bien meilleures sur les sens venant de lemmes simples. Le fait que cette différence existe aussi bien pour la classification joignant le lemme à la définition et pour celle utilisant la définition seule semble indiquer que cette baisse de performance est intrinsèque. En effet, nous aurions plutôt attendu que cette baisse soit visible pour la classification utilisant le lemme car d'une part le lemme est plus compliqué à interpréter en tant qu'expression complexe, et le classifieur ne s'entraîne pas à voir ce genre d'expression avant la définition car il n'y a aucun lemme générant ce genre de sens dans les données d'entraînement. Ainsi, cette plus faible performance sur les sens de lemmes complexes sans utilisation de lemme semble indiquer que son origine est potentiellement dans la qualité des définitions ou bien dans la distribution des classes sémantiques pour ces lemmes et sens. Nous pouvons aussi noter que le nombre de sens de lemmes complexes est beaucoup plus faible que ceux de lemmes simples et les performances sur ces sens sont plus sujettes à des variations suivant qu'un petit nombre de sens soient bien prédits ou non. Nous pouvons notamment noter que d'après les résultats par classe (table 11) et la répartition des supersens en fonction du type simple ou complexe du lemme associé au sens (table 5), la classe *Person* est beaucoup plus présente dans les sens de lemme simple et mène à des résultats excellents (F-mesure supérieure à 96% dans les données de validation), tandis que les classes *Cognition* et *State* sont davantages présentes dans les sens de lemmes complexes et mènent à des résultats moins bons (F-mesures inférieures respectivement à 67% et 63% dans les données de validation).

	rand-dev		freq-dev	
	Supersens	Hypersens	Supersens	Hypersens
def				
tous	78.1	86.5	73.1	78.9
lemme simple	79.6	87.7	73.1	78.9
mwe	70.2	79.8	-	-
def+lem & ex				
tous	84.3	91.3	77.1	83.0
lemme simple	85.5	92.3	77.1	83.0
mwe	77.4	85.7	-	-

TABLE 12 – Exactitude en %, sur les données de validation RANDOM et FREQUENCY, suivant que le lemme du sens soit simples ou complexes, en utilisant la définition seule (**def**) et la combinaison du classifieur de définition et classifieur d'occurrences (**def+lemme & ex**, cf. section 4.1.3).

	rand-dev		freq-dev	
	Supersens	Hypersens	Supersens	Hypersens
def				
tous	79.4	87.8	77.0	83.0
lemme simple	80.3	88.4	77.0	83.0
mwe	75.0	84.5	-	-
def+lem & ex				
tous	84.8	91.9	80.3	86.2
lemme simple	86.0	92.8	80.3	86.2
mwe	78.6	86.9	-	-

TABLE 13 – Exactitude partielle en %, sur les données de validation RANDOM et FREQUENCY, suivant que le lemme du sens soit simples ou complexes, en utilisant la définition seule (**def**) et la combinaison du classifieur de définition et classifieur d'occurrences (**def+lemme & ex**, cf. section 4.1.3).

Concernant les performances du classifieur sur les sous-ensembles de sens venant de lemmes polysémiques ou monosémiques (tables 14 et 15), nous observons que les exactitudes et exactitudes partielles sont plus élevées pour les sens venant de lemmes monosémiques. L'origine de cette différence entre sens de lemme monosémique et sens de lemme polysémique peut être de nature liée à la distribution des supersens dans les deux sous-ensembles ou bien inhérente à leurs caractéristiques propres.

	rand-dev		freq-dev	
	Supersens	Hypersens	Supersens	Hypersens
def+lem & ex				
tous	84.3	91.3	77.1	83.0
monosémie	85.6	91.9	82.5	87.5
polysémie	77.1	87.9	76.1	82.1

TABLE 14 – Exactitude en %, sur les données de validation RANDOM et FREQUENCY, suivant que le lemme du sens soit monosémique ou polysémique, le meilleur classifieur, celui combinant définitions avec lemme et exemples(**def+lemme & ex**, cf. section 4.1.3).

	rand-dev		freq-dev	
	Supersens	Hypersens	Supersens	Hypersens
def+lem & ex				
tous	84.8	91.9	80.3	86.2
monosémie	86.2	92.6	83.7	88.7
polysémie	77.1	87.9	79.7	85.7

TABLE 15 – Exactitude partielle en %, sur les données de validation RANDOM et FREQUENCY, suivant que le lemme du sens soit monosémique ou polysémique, le meilleur classifieur, celui combinant définitions avec lemme et exemples(**def+lemme & ex**, cf. section 4.1.3).

Voici enfin les évaluations sur les jeux de données de test pour les données RANDOM et FREQUENCY :

	rand-test		freq-test	
	Supersens	Hypersens	Supersens	Hypersens
Exactitude	84.0	88.1	79.0	83.9
Exactitude partielle	85.8	90.0	82.4	87.3

TABLE 16 – Exactitude et exactitude partielle pour les jeux de données rand-test et freq-test en évaluant le meilleur classifieur, celui combinant définitions avec lemme et exemples(**def**+**lemme & ex**, cf. section 4.1.3).

6 Analyse de la ressource produite

Dans cette partie nous allons analyser en profondeur la ressource lexicale produite à l'issue du travail de recherche réalisé. Dans un premier temps (section 6.1), nous analyserons le contenu de la ressource, à savoir les informations effectivement extraites, le processus d'enrichissement à partir du classifieur et quelques considérations concernant les informations sémantiques à partir de l'annotation automatique réalisée. Dans un second temps (section 6.2), nous présenterons des idées d'exploitation de cette ressource pour faire de la recherche en sémantique lexicale et en traitement automatique des langues, et démontrerons ainsi le potentiel d'une telle ressource et l'intérêt de la produire.

6.1 Contenu de la ressource

6.1.1 Production de la ressource

A ce stade de notre travail, nous avons alors à notre disposition l'extraction du Wiktionnaire d'un côté et d'un autre côté notre meilleur classifieur de sens lexical. Nous faisons alors la classification de tous les sens de la ressource avec notre classifieur pour obtenir toutes les prédictions de supersens.

Nous avons choisi de procéder selon les modalités suivante pour l'annotation sémantique de la ressource :

— Les sens dont la prédiction est un supersens simple reçoivent comme étiquettes sémantiques ce supersens et l'hypersens correspondant ;

— Les sens dont la prédiction est un supersens complexe reçoivent comme étiquettes le supersens complexe, les deux supersens simples composant le supersens complexe, l'hypersens complexe associé ainsi que les hypersens simples composant l'hypersens complexe ;

— Les sens dont la prédiction est le supersens *Relation* se voient assigner à la place l'étiquette sémantique *State*. Ce choix est dû aux très mauvaises performances du classifieur sur cette classe qui est également un supersens proche sémantiquement de la classe *State*.

6.1.2 Analyse de la ressource

La ressource lexicographique ainsi obtenue est un lexique avec une importante couverture lexicale des noms du français. Chacun des 228 989 noms est associé à au moins un sens lexical décrivant une de ses utilisations possibles, et cela donne lieu à 306 225 sens associés. Nous indiquons quelques informations statistiques liées à la ressource dans la table 17.

Information	**Valeur**
Nombre de sens	306 225
Nombre de lemmes	228 989
Ratio nb sens/nb lemmes	1.34
Ratio nb sens/nb entrées nominales	1.31
Lemmes avec homonymie	2%
Lemmes monosémiques	83%
Lemmes complexes (MWE)	20%
Sens sans exemples	50%
Sens gentilés	20%

TABLE 17 – Statistiques générales de la ressource lexicale extraite.

Une statistique qui peut sembler étonnante est la large domination du supersens *Person* (figure 11) et des hypersens d'entités (figure 12) dans les distributions des classes sémantiques dans l'annotation de la ressource. Cela s'explique notamment par la présence de 20% de sens gentilés (table 17), c'est à dire qui désignent les habitants d'un certain lieu.

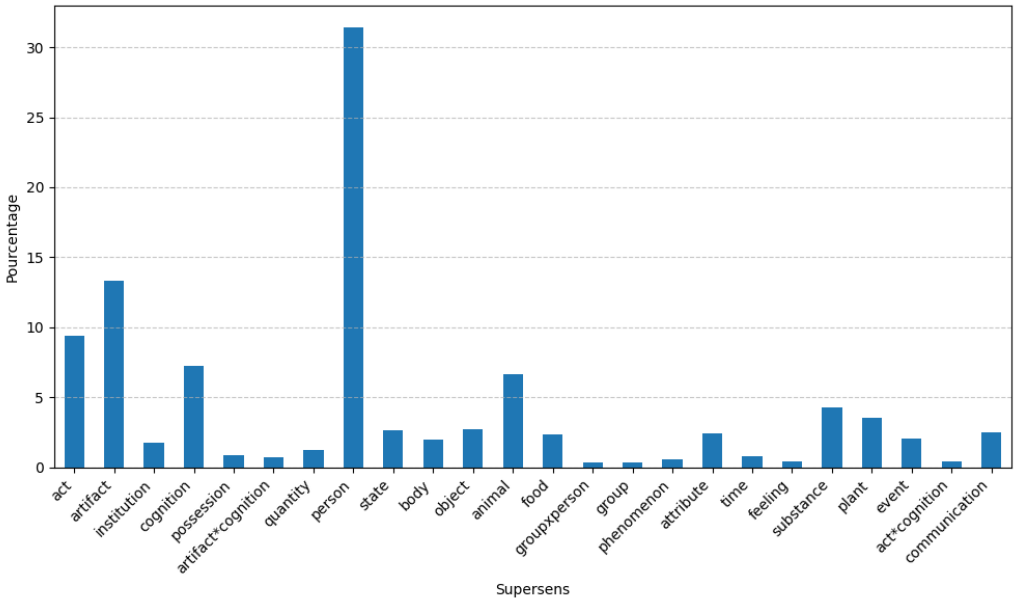

FIGURE 11 – Distribution des supersens dans la ressource en %

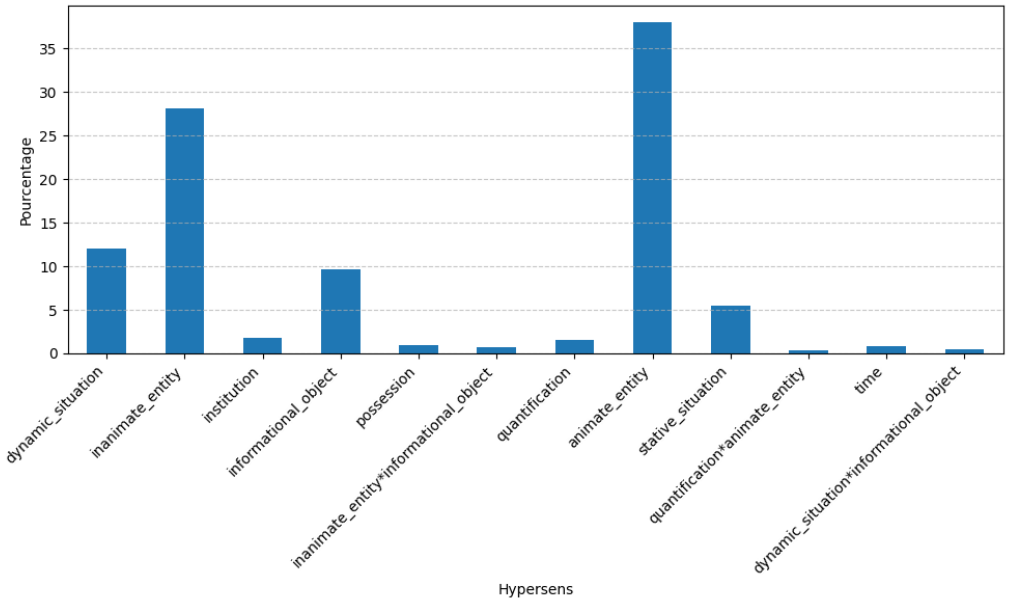

FIGURE 12 – Distribution des hypersens dans la ressource en %

Supersens	Général	Monosémie	Polysémie	Lemme simple	MWE
act	9.4	8.1	11.6	9.2	10.6
act*cognition	0.4	0.3	0.7	0.4	0.6
animal	6.6	8.5	3.5	5.1	14.0
artifact	13.3	10.0	18.6	12.8	15.5
artifact*cognition	0.7	0.6	0.9	0.6	1.5
attribute	2.5	1.9	3.3	2.6	1.9
body	2.0	2.0	1.9	1.6	3.9
cognition	7.2	6.3	8.6	6.4	10.9
communication	2.5	3.1	1.5	2.4	2.7
event	2.1	1.7	2.6	2.0	2.3
feeling	0.4	0.3	0.6	0.4	0.3
food	2.4	2.4	2.4	2.2	3.2
group	0.3	0.2	0.6	0.3	0.4
groupxperson	0.3	0.2	0.6	0.3	0.4
institution	1.8	1.5	2.2	1.6	2.4
object	2.7	2.7	2.7	2.5	3.6
person	31.4	34.7	26.0	36.6	7.0
phenomenon	0.6	0.4	0.9	0.6	0.8
plant	3.5	4.3	2.3	3.0	6.2
possession	0.9	0.8	1.0	0.7	1.6
quantity	1.3	1.2	1.3	1.2	1.5
state	2.6	2.7	2.5	2.7	2.4
substance	4.2	5.2	2.6	4.1	4.8
time	0.8	0.7	0.9	0.7	1.4

TABLE 18 – Distributions des supersens en % dans la ressource entière, dans le sous-ensemble de la ressource de sens issus de lemmes monosémiques puis polysémiques, dans le sous-ensemble de la ressource de sens issus de lemmes simples puis étant des expressions polylexicales

Hypersens	Général	Monosémie	Polysémie	Lemme simple	MWE
animate entity	38.0	43.2	29.5	41.6	21.1
dynamic situation	12.1	10.3	15.1	11.8	13.7
dynamic situation*informational object	0.4	0.3	0.7	0.4	0.6
inanimate entity	28.1	26.6	30.6	26.2	37.3
inanimate entity*informational object	0.7	0.6	0.9	0.6	1.5
informational object	9.7	9.4	10.1	8.8	13.6
institution	1.8	1.5	2.2	1.6	2.4
possession	0.9	0.8	1.0	0.7	1.6
quantification	1.6	1.4	1.9	1.5	1.9
quantificationxanimate entity	0.3	0.2	0.6	0.3	0.4
stative situation	5.5	4.9	6.4	5.7	4.5
time	0.8	0.7	0.9	0.7	1.4

TABLE 19 – Distributions des hypersens dans la ressource entière, dans le sous-ensemble de la ressource de sens issus de lemmes monosémiques puis polysémiques, dans le sous-ensemble de la ressource de sens issus de lemmes simples puis étant des expressions polylexicales

6.2 Exploitation de la ressource

La ressource produite va pouvoir être exploitée dans différents domaines. Elle va en premier lieu servir dans des études quantitatives en sémantique lexicale. À titre d'illustration, nous proposerons dans la section suivante une étude de la répartition des sens entre lexique simple et lexique morphologiquement construit.

La ressource pourra également être utilisée en TAL, pour améliorer les performances d'une tâche de catégorisation sémantique en contexte grâce à l'ajout d'informations issues du lexique. On espère notamment que ce genre d'information lexicale améliore le traitement des mots rares, la qualité de leur représentation issues de corpus ne permettant pas pour l'heure d'atteindre des résultats satisfaisant (Blevins et al., 2021).

Enfin, cette classification sémantique des sens nominaux du Wiktionnaire va permettre de poursuivre la valorisation de cette ressource. Un des pro-

chains objectifs est d'attribuer des informations de fréquences sur les sens. Suivant la méthode appliquée dans (Aloui et al., 2020), la ressource permettra d'entrainer un classifieur sur un corpus pseudo annoté en supersens grâce aux informations associées aux noms monosémiques uniquement (83%). Ce classifieur sera appliqué aux occurrences de noms polysémiques dans un très gros corpus et les prédictions seront utilisées pour obtenir des estimations de fréquences de sens.

6.3 Propriétés sémantiques des lexiques simple et construit

Dans son étude des liens entre classes sémantiques et catégories grammaticales, Croft (1991, 2022) formule un certain nombre de généralisations à propos des noms : les noms morphologiquement simples dénoteraient prototypiquement des objets, tandis que les noms dénotant des actions seraient quant à eux construits sur des verbes (e.g *dissolution* < *dissoudre*) et ceux dénotant des situations statives le seraient sur des adjectifs (e.g. *inquiétude* < *inquiet*)

Une première étude empirique a partiellement validé ces hypothèses de Croft (Tribout et al., 2014). Les auteurs ont annoté manuellement 3,489 noms morphologiquement simples à l'aide d'une classification sémantique à gros grain, constituée uniquement des 3 classes Object, Action, et State. Les résultats montrent que si une majorité des sens de ces noms simples relèvent bien de la classe Object, une portion non négligeable d'entre eux (25%) relèvent d'une autre classe. L'analyse a montré également que la classification tripartite n'est pas assez couvrante. Par exemple, les noms comme *mardi* ou *tonne* ne relèvent d'aucune de ces 3 classes, mais respectivement des classes Time et Quantification. Par ailleurs, si certains noms simples relèvent bien originellement d'une autre classe qu'Objet (par ex. *crime* qui dénote dans son sens premier une action), la polysémie semble jouer un rôle important dans la distribution de ces sens. Par exemple l'emploi de *bœuf* qui dénote une action (jam session) est dérivé sémantiquement de l'emploi Object (animal) du nom.

La ressource qui a été développée dans ce travail va nous permettre d'explorer plus largement la répartition des sens entre lexique simple et lexique construit.

6.4 Données

Pour mener cette étude, nous avons sélectionné un sous ensemble de noms pour lequels nous disposons d'informations morphologiques et d'information de fréquence en corpus. Ces dernières, qui nous permettent une représentation naturelle de l'ensemble étudié, ont été extraites de la ressource *Lexique 3* (New et al. 2004). Les informations morphologiques ont été extraites des 2 ressources *Demonette* (Namer et al 2024) et *Echantinom* (Bonami et Tribout 2021). Les noms figurant dans l'une ou l'autre de ces deux ressources se sont ainsi vus attribuer des informations sur leur construction morphologique (simplex, conversion, suffixation, etc), sur leur affixe, lorsqu'ils sont dérivés morphologiquement (ex. -age, -eur, re-), et sur la catégorie de leur base (verbale, adjectivale, nominale, etc).

Au total, le sous-ensemble issu de l'intersection entre les noms de notre ressource, ceux de Lexique 3 et ceux d'une des deux ressources morphologiques est constitué de 17 473 noms associés à 47 499 sens (taux de polysémie moyen : 2,7). Le tableau 20 montre la répartition des lemmes de cet ensemble selon leur construction morphologique.

	Lemmes	**Proportion**	**Exemple**
Suffix	9,032	51.7%	*cotisation*
Simplex	5,488	31.4%	*heure*
Conversion	2,476	14.2%	*siège*
Polylexical	271	1.6%	*hors-bord*
Nonconcat	115	0.7%	*micro*
Prefix	80	0.3%	*reflux*
Pre-suf	11	0.1%	*coreligionnaire*
Total	17,474		

TABLE 20 – Répartition des lemmes par type de construction morphologique

Pour la suite de l'étude, nous nous concentrons sur le groupe des noms simples et sur les deux sous-groupes de noms suffixés qui nous intéressent : ceux construits sur bases verbales et ceux construits sur base adjectivale.

6.4.1 Résultats

La figure 13 et la table 21 illustrent la répartion des sens (catégorisés ici en hypersens) pour les noms morphologiquement simples, les noms dérivés de verbes et les noms dérivés d'adjectifs.

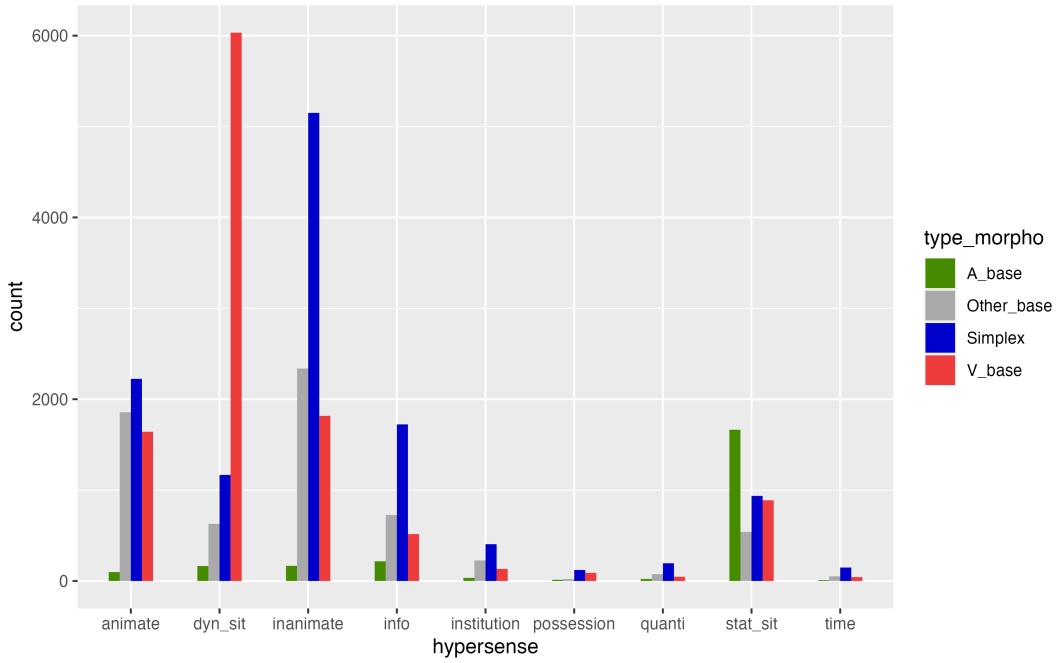

FIGURE 13 – Répartition des hypersens selon la construction morphologique des N

Concernant les noms simples, les résultats confirment, à plus large échelle, ceux de Tribout et al. (2014). Comme indiqué dans la 1ère colonne de la Table 21, la majorité des noms morphologiquement simples dénotent bien des entités concrètes (61.1% en regroupant les classes Animate_entity et Inanimate_entity) mais une partie substantielle d'entre eux dénotent des entités abstraites (21,5% si l'on additionne les effectifs de Informational_object, Institution, Possession, Quantity et Time). En outre, une partie non négligeable des sens des noms morphologiquement simples dénotent des situations, qu'elles soient dynamique ou stative (17,5%), ce qui confirme que, bien que validée empiriquement, l'hypothèse de Croft doit être quelque peu relativisée.

Considérons maintenant les sens nominaux dénotant des situations. Rap-

	Simplex	**Base_V**	**Base_A**
Animate_entity	18.4%	14.6%	4.1%
Dynamic_situation	9.7%	53.8%	6.9%
Inanimate_entity	42.7%	16.2%	7%
Informational_object	14.3%	4.6%	9.1%
Institution	3.4%	1.2%	1.4%
Possession	1.0%	0.8%	0.5%
Quantity	1.6%	0.4%	0.9%
Stative_situation	7.8%	7.9%	69.7%
Time	1.2%	0.4%	0.4%

TABLE 21 – Répartition des hypersens pour 3 sous-groupes

pelons que Croft prédit que les noms d'action sont dérivés de verbes et que les noms de propriété sont dérivés d'adjectifs. La table 22 ci-dessous présente ce qui ressort de nos données.

	Simplex	**Base_V**	**Base_A**	**Base_Other**
Dynamic_situation	14.6%	75.5%	2.1%	7.9%
Stative_situation	23.2%	22.0%	41.3%	13.4%

TABLE 22 – Origine morphologique des noms dénotant des sens situationnels (situation dynamique ou stative)

On voit que là encore la corrélation entre catégories sémantiques et catégories grammaticales est loin d'être parfaite. Les sens actionnels sont certes majoritairement dérivés de verbes mais ça ne semble pas être le cas pour environ 1/4 d'entre eux. Quant aux sens statifs, ils ne sont, considérés dans leur ensemble, que minoritairement dérivés d'adjectifs (41%).

7 Conclusion

Nous avons étudié dans ce mémoire la classification automatique de représentations contextualisées de noms du français, qu'il s'agisse de leur description ou de leur occurrence dans un texte, avec des classes sémantiques à grain épais. Nous avons développé et entraîné un classifieur sémantique qui permet de réaliser une telle classification avec des performances plus ou moins raisonnables et avons enrichi grâce à lui une ressource lexicale, le Wiktionnaire, qui décrit les sens d'un très large ensemble de noms du français. Cette ressource lexicale enrichie d'information sémantique permettra à la communauté francophone d'avoir de nouvelles perspectives pour des recherches en sémantique lexicale et en traitement automatique des langues.

Cependant, malgré les performances intéressantes obtenues, nous savons que les classifieurs testés dans cette étude pourraient être améliorés. Tout d'abord, les efforts de la communauté en intelligence artificielle actuellement portés sur la création de modèles génératifs basés sur l'architecture transformers pourraient permettre d'obtenir des modèles plus performants. De plus, ce travail de recherche s'est concentré principalement sur les modèles de type BERT et les autres architectures pourraient être exploitées davantage pour tenter d'obtenir de meilleurs résultats. Enfin, de nouvelles idées apparaissent dans la sphère de l'apprentissage machine régulièrement et certaines d'entre elles offrent des perspectives intéressantes pour de futur modèles. Nous avons par exemple fait quelques tests avec une tête de classification où le MLP était remplacé par une couche de type Kolmogorov-Arnold Network (KAN) (Liu et al., 2024), mais il s'agit là d'une idée assez récente et nous n'avons pas pu pousser assez loin l'idée sur nos classifieurs pour savoir ce qu'il en est pour la classification sémantique. Nous pouvons également citer certaines alternatives aux transformers qui pourraient se révéler pertinentes à l'avenir. Par exemple, l'architecture Mamba (Gu and Dao, 2024) qui utilise le concept d'espaces d'états sélectifs, ou bien une nouvelle approche pour les LSTM, le xLSTM (extended Long Short-Term Memory) (Beck et al., 2024), qui introduit des altérations à la structure de base pour régler ses limitations.

Références

Aloui, C., Ramisch, C., Nasr, A., and Barque, L. (2020). SLICE : Supersense-based lightweight interpretable contextual embeddings. In Scott, D., Bel, N., and Zong, C., editors, *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3357–3370, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Barque, L., Haas, P., Huyghe, R., Tribout, D., Candito, M., Crabbé, B., and Segonne, V. (2020). FrSemCor : Annotating a French corpus with supersenses. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 5912–5918, Marseille, France. European Language Resources Association.

Basile, P. (2013). Super-sense tagging using support vector machines and distributional features. In Magnini, B., Cutugno, F., Falcone, M., and Pianta, E., editors, *Evaluation of Natural Language and Speech Tools for Italian*, pages 176–185, Berlin, Heidelberg. Springer Berlin Heidelberg.

Beck, M., Pöppel, K., Spanring, M., Auer, A., Prudnikova, O., Kopp, M., Klambauer, G., Brandstetter, J., and Hochreiter, S. (2024). xlstm : Extended long short-term memory.

Blevins, T., Joshi, M., and Zettlemoyer, L. (2021). FEWS : Large-scale, low-shot word sense disambiguation with the dictionary. In Merlo, P., Tiedemann, J., and Tsarfaty, R., editors, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics : Main Volume*, pages 455–465, Online. Association for Computational Linguistics.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T. J., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language Models are Few-Shot Learners. *ArXiv*, abs/2005.14165.

Candito, M. and Seddah, D. (2012). Le corpus sequoia : annotation syntaxique et exploitation pour l'adaptation d'analyseur par pont lexical (the sequoia corpus : Syntactic annotation and use for a parser lexical domain adaptation method) [in French]. In Antoniadis, G., Blanchon, H., and Sérasset, G., editors, *Proceedings of the Joint Conference JEP-TALN-RECITAL 2012, volume 2 : TALN*, pages 321–334, Grenoble, France. ATALA/AFCP.

Ciaramita, M. and Altun, Y. (2006). Broad-coverage sense disambiguation and information extraction with a supersense sequence tagger. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 594–602.

Ciaramita, M. and Johnson, M. (2003). Supersense tagging of unknown nouns in WordNet. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 168–175.

Croft, W. (1991). *Syntactic Categories and Grammatical Relations : The Cognitive Organization of Information*. University Press of Chicago.

Croft, W. (2022). *Morphosyntax : constructions of the world's languages*. Cambridge University Press.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT : Pre-training of deep bidirectional transformers for language understanding. In Burstein, J., Doran, C., and Solorio, T., editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics : Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Flekova, L. and Gurevych, I. (2016). Supersense embeddings : A unified model for supersense interpretation, prediction, and utilization. In Erk, K. and Smith, N. A., editors, *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1 : Long Papers)*, pages 2029–2041, Berlin, Germany. Association for Computational Linguistics.

Gu, A. and Dao, T. (2024). Mamba : Linear-time sequence modeling with selective state spaces.

Le, H., Vial, L., Frej, J., Segonne, V., Coavoux, M., Lecouteux, B., Allauzen, A., Crabbé, B., Besacier, L., and Schwab, D. (2020). FlauBERT : Unsupervised language model pre-training for French. In Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., and Piperidis, S., editors, *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 2479–2490, Marseille, France. European Language Resources Association.

Levine, Y., Lenz, B., Dagan, O., Ram, O., Padnos, D., Sharir, O., Shalev-Shwartz, S., Shashua, A., and Shoham, Y. (2020). SenseBERT : Driving some sense into BERT. In Jurafsky, D., Chai, J., Schluter, N., and Tetreault, J., editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4656–4667, Online. Association for Computational Linguistics.

Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljačić, M., Hou, T. Y., and Tegmark, M. (2024). Kan : Kolmogorov-arnold networks.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013). Efficient estimation of word representations in vector space. In *Proceedings of the 1st International Conference on Learning Representations*.

Miller, G., Beckwith, R., Fellbaum, C., D., G., and Miller, K. (1990). Wordnet : An online lexical database. *International Journal of Lexicography*, (3) :235–244.

Polguère, A. (2014). Principles of lexical network systemic modeling (principes de modélisation systémique des réseaux lexicaux) [in French]. In Blache, P., Béchet, F., and Bigi, B., editors, *Proceedings of TALN 2014 (Volume 1 : Long Papers)*, pages 79–90, Marseille, France. Association pour le Traitement Automatique des Langues.

Sagot, B. and Fišer, D. (2008). Construction d'un wordnet libre du français à partir de ressources multilingues. In Béchet, F. and Bonastre, J.-F., editors, *Actes de la 15ème conférence sur le Traitement Automatique des Langues Naturelles. Articles longs*, pages 171–180, Avignon, France. ATALA.

Segonne, V., Candito, M., and Crabbé, B. (2019). Using Wiktionary as a resource for WSD : the case of French verbs. In Dobnik, S., Chatzikyriakidis, S., and Demberg, V., editors, *Proceedings of the 13th International*

Conference on Computational Semantics - Long Papers, pages 259–270, Gothenburg, Sweden. Association for Computational Linguistics.

Sérasset, G. (2014). DBnary : Wiktionary as a Lemon-Based Multilingual Lexical Resource in RDF. *Semantic Web Journal (special issue on Multilingual Linked Open Data)*. To appear.

Tribout, D., Barque, L., Haas, P., and Huyghe, R. (2014). De la simplicité en morphologie. In *SHS web of conferences*, volume 8, pages 1879–1890. EDP Sciences.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2023). Attention is all you need.

Vossen, P. (1998). *EuroWordNet : A Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers.